INSTITUTO TECNOLÓGICO
Y DE ESTUDIOS SUPERIORES DE OCCIDENTE

Reconocimiento de validez oficial de estudios de nivel superior según acuerdo secretarial 15018,
publicado en el Diario Oficial de la Federación el 29 de noviembre de 1976.

Departamento de Electrónica, Sistemas e Informática

DOCTORADO EN CIENCIAS DE LA INGENIERÍA

TÉCNICAS DE APRENDIZAJE AUTOMÁTICO Y ENFOQUES DE OPTIMIZACIÓN PARA VALIDACIÓN Y PRUEBAS ANALÓGICAS

Tesis que para obtener el grado de

DOCTOR EN CIENCIAS DE LA INGENIERÍA

presenta: Andrés Viveros Wacher

Director de tesis: Dr. José Ernesto Rayas Sánchez

Tlaquepaque, Jalisco. Diciembre de 2020

DOCTOR EN CIENCIAS DE LA INGENIERÍA (2020) ITESO, Tlaquepaque, Jal., México

TÍTULO: **Técnicas de aprendizaje automático y enfoques de optimización para validación y pruebas analógicas**

AUTOR: Andrés Viveros Wacher
Ingeniero Eléctrico Electrónico (Universidad Nacional Autónoma de México, México)
Maestro en Ingeniería de Sistemas Microelectrónicos (Universidad de Bristol, Reino Unido)

DIRECTOR DE TESIS: José Ernesto Rayas Sánchez
Profesor Numerario del Departamento de Electrónica, Sistemas e Informática, ITESO
Ingeniero en Electrónica (ITESO, México)
Maestro en Sistemas Electrónicos (ITESM Campus Monterrey, México)
Doctor en Ingeniería Eléctrica (Universidad McMaster, Canadá)
Senior, IEEE

NÚMERO DE PÁGINAS: xxxii, **141**

ITESO – The Jesuit University of Guadalajara

Department of Electronics, Systems, and Informatics

DOCTORAL PROGRAM IN ENGINEERING SCIENCES

MACHINE LEARNING TECHNIQUES AND OPTIMIZATION APPROACHES FOR ANALOG VALIDATION AND TESTING

Thesis to obtain the degree of

DOCTOR IN ENGINEERING SCIENCES

Presents: Andrés Viveros-Wacher

Thesis Director: Dr. José Ernesto Rayas-Sánchez

Tlaquepaque, Jalisco, México
December 2020

PhD in Engineering Sciences (2020) ITESO, Tlaquepaque, Jal., Mexico

TITLE:	**Machine Learning Techniques and Optimization Approaches for Analog Validation and Testing**
AUTHOR:	Andrés Viveros-Wacher Bachelor’s degree in electrical and electronics engineering (*Universidad Nacional Autónoma de México*, Mexico) Master’s degree in microelectronics systems engineering (University of Bristol, UK)
THESIS DIRECTOR:	José Ernesto Rayas-Sánchez Department of Electronics, Systems, and Informatics, ITESO-The Jesuit University of Guadalajara Bachelor’s degree in electronics engineering (ITESO, Mexico) Master’s degree in electrical engineering (ITESM, Mexico) Ph.D. degree in electrical engineering (McMaster University, Canada) Senior, IEEE
NUMBER OF PAGES:	xxxii, **141**

To my wife, Pamela, whose support and encouragement
have made this achievement possible

Resumen

La validación y pruebas analógicas post-silicio de enlaces de entrada/salida (HSIO, por sus siglas en inglés) en plataformas de computadora de alto rendimiento tienen retos continuamente crecientes por varios factores: la ley de Moore sigue avanzando con una constante miniaturización de los nodos tecnológicos, la complejidad de los productos continúa creciendo junto con funcionalidades más demandantes, y las velocidades de datos continúan aumentando con las nuevas generaciones de interfaces HSIO, entre otros. Por otro lado, existe la necesidad de mantener calendarios agresivos de lanzamiento de productos para mantener competitividad en el mercado. En este escenario, es crucial para las compañías encontrar soluciones innovadoras para acelerar los procesos de validación y prueba sin sacrificar la calidad de los resultados. Esta tesis doctoral propone una serie de metodologías de aprendizaje automático y de optimización para mejorar varios procesos de validación y pruebas analógicas, en su mayoría asociadas a enlaces HSIO en plataformas de cómputo modernas. Primero demuestra cómo los márgenes del diagrama de ojo del receptor son significativamente mejorados al usar un enfoque de optimización basado en diseño de experimentos. Posteriormente, muestra cómo la prueba de tolerancia a fluctuaciones en el tiempo es dramáticamente acelerada al ejecutar un algoritmo eficiente de optimización numérica. La presente tesis también describe cómo algoritmos de aprendizaje automático se usan para crear modelos sustitutos del sistema bajo prueba, para acelerar el proceso de sintonización de una plataforma física durante la validación eléctrica post-silicio usando optimización basada en sustitutos y mapeo espacial agresivo. Adicionalmente, la tesis elabora sobre la identificación automática de fallas analógicas, desarrollando modelos de redes neuronales de inyección de fallas y un algoritmo de detección basado en optimización explotando la extracción de parámetros restringida. Finalmente, esta tesis doctoral describe cómo se pueden entrenar modelos de redes neuronales profundas para clasificar la precisión de la extrapolación de la tasa de errores de bits (BER, por sus siglas en inglés) en mediciones de márgenes para estándares industriales de BER. Cada metodología propuesta es validada con casos de prueba, demostrando no solo la eficiencia de las técnicas propuestas sino también mejoras significativas a los procesos analógicos post-silicio en general. Algunas oportunidades de investigación futura y desarrollos con potencial promisorio asociados a las pruebas y validación analógica post-silicio también son identificados.

Summary

Analog post-silicon validation and testing of high-speed input/output (HSIO) links in high-performance computer platforms has ever-increasing challenges caused by several factors: Moore's law continues to advance with constant technology node miniaturization, product complexity keeps increasing along with more demanding functionalities, and data rates continue to escalate with new generations of HSIO interfaces, among others. On the other hand, there is a need to maintain aggressive product launch schedules in order to maintain market competitiveness. This scenario makes crucial for companies to find innovative solutions to accelerate validation and testing processes without sacrificing results quality. This doctoral dissertation proposes a set of machine learning and optimization methodologies aimed at improving several analog validation and testing industrial processes, most of them associated to HSIO links in modern computer platforms. It first demonstrates how receiver eye diagram margins are significantly improved by using an optimization approach based on design of experiments. It subsequently shows how the jitter tolerance test is dramatically accelerated by employing an efficient numerical optimization algorithm during execution. The present Ph.D. thesis also describes how machine learning algorithms are exploited to create surrogate models of the system under test to accelerate the physical platform tuning process during electrical post-silicon validation by using surrogate-based optimization and aggressive space mapping. Additionally, the proposed doctoral dissertation elaborates on automated analog fault identification, for which fault injection neural network models are developed by an optimization-based detection algorithm that exploits constrained parameter extraction. Finally, this Ph.D. thesis describes how deep neural network models can be properly trained to classify bit error rate (BER) extrapolation precision in margin measurements under specified BER industry standards. Each methodology proposed in this doctoral dissertation is properly validated by suitable test cases, demonstrating not only the efficiency of the proposed techniques but also the improvements to the overall analog post-silicon processes. Some future research opportunities and promising potential developments associated to analog post-silicon validation and testing are also envisioned.

Acknowledgements

The author wishes to express his sincere appreciation to Dr. José Ernesto Rayas-Sánchez, professor of the Department of Electronics, Systems, and Informatics at ITESO, and director of research in the Computer-Aided Engineering of Circuits and Systems (CAECAS) group at ITESO, for his encouragement, expert guidance, and keen supervision as doctoral dissertation director throughout the course of this work. The author offers his gratitude to Dr. Adam Norman from Intel Corporation, for his support during the development of this dissertation. He also thanks Dr. Paulo López-Meyer, Dr. Omar Longoria-Gándara, Dr. José Luis Pizano-Escalante, and Dr. Zabdiel Brito-Brito, members of his Ph.D. Thesis Examination Committee, for their interest, assessment, and suggestions.

Special thanks are due to Dr. Zabdiel Brito-Brito, from CAECAS research group at ITESO. It is the author's pleasure to acknowledge fruitful collaboration and stimulating discussions with his colleagues of CAECAS research group at ITESO – The Jesuit University of Guadalajara: Francisco Rangel-Patiño, José Luis Chávez-Hurtado, Rafael del-Rey-Acuña, Felipe de Jesús Leal-Romo, Jorge Dávalos-Guzmán, Rosa Jaqueline Sánchez-Mesa, Roberto Loera-Díaz and Edna Moreno-Mojica.

The author also thanks his peers and colleagues from Intel Corporation, for fruitful cooperation and helpful technical discussions: Ricardo Baca-Baylón, Edgar A. Vega-Ochoa, Miguel Dávalos-Santana, Jesús Gómez-Lopez, Alejandro Moscoso, Suriyaprakash Natarajan, Prashant Goteti, and many others.

The author gratefully acknowledges the financial assistance through a scholarship granted by the *Consejo Nacional de Ciencia y Tecnología* (CONACYT), Mexican Government, as well as the financial support provided by Intel Corporation.

Finally, special thanks are due to my family: my wife Pamela, my parents, and my sister, for their understanding, patience, and continuous loving support.

Contenido

Contents

List of Figures

List of Tables

List of Acronyms

3LP	Three Layer Perceptron
ANN	Artificial Neural Networks
ANOVA	Analysis of Variance
ASM	Aggressive Space Mapping
BB	Box Behnken
BDV	Bench Design Validation
BER	Bit Error Rate
CCD	Central Composite Design
CDR	Clock and Data Recovery
CRD	Completely Randomized Design
CTLE	Continuous Time Linear Equalizer
DCD	Duty Cycle Distortion
DFE	Decision Feedback Equalizer
DFT	Design For Test
DMI	Direct Media Interface
DNN	Deep Neural Network
DUT	Device Under Test
DV	Design Validation
DoE	Design of Experiments
EHM	Eye Height Margin
EQ	Equalization
EV	Electrical Validation
FIR	Finite Impulse Response
FV	Functional Validation
Gbps	Gigabit per Second
GRNN	Generalized Regression Neural Network
HSIO	High-Speed I/O
I/O	Input/ Output
IC	Integrated Circuit
ICA	Independent Component Analysis
ISI	Inter-Symbol Interference
IoT	Internet of Things
JTOL	Jitter Tolerance
LB	Lower Bound
LSSD	Level Sensitive Scan Design
NN	Neural Networks
OA	Orthogonal Arrays
PCA	Principal Component Analysis
PCH	Platfrom Control Hub
PCIe	Peripheral Component Interconnect Express
PE	Parameter Extraction
PHY	Physical Layer
PLS	Partial Least Squares

LIST OF ACRONYMS

PSM	Polynomial Surrogate Modeling
PVT	Process, Voltage and Temperature
QTA	Qualitative Trend Analysis
RSM	Response Surface Methodology
ReLu	Rectified Linear Unit
Rx	Receiver
SATA	Serial Advanced Technology Attachment
SBO	Surrogate Based Optimization
SIV	Signal Integrity Validation
SMV	System Marginality Validation
SoC	System on Chip
SVM	Support Vector Machine
TIE	Time Interval Error
TTM	Time to Market
Tx	Transmitter
UB	Upper Bound
USB	Universal Serial Bus
VGA	Variable Gain Amplifier
XAUI	Ten (X) Attachment Unit Interface

Introduction

It is projected that by 2025, there will be 21 billion connected devices [Tankovska-20]. 5G technologies are expected to act as a catalyst to increase not only the number of devices, but also the type of devices, ranging from simple consumer internet of things (IoT) sensors and actuators, to edge computing devices, communications and base stations, and autonomous automotive applications, on top of the more traditional personal computing devices (laptops, PCs, tablets, cellular phones, and wearables). All of these connected devices will generate and/or consume data that will be processed in data centers and other devices within the cloud computing ecosystem.

Given the diversity, as well as the complexity of these devices, the validation and testing performed on them has become crucial and a challenging problem [Mishra-17]. On top of this, it is known that up to 70% of the time and resources allocated for chip design is spent on validation [Kabisatpathy-05]. Therefore, it is extremely important for companies to find innovative methodologies to accelerate validation and enable a fast time to market (TTM) in order to maintain competitiveness.

As technology advances into new technology nodes, higher system complexity, as well as increased data rates, the "guaranteed by design" quality paradigm is no longer valid [Fan-11]. Extensive validation and testing need to be performed on a product prior to its market release to ensure not only correct functionality, but most importantly, customer satisfaction. Despite the advances in pre-silicon validation, system-level simulation of computing devices is time-prohibitive. In contrast, post-silicon validation is performed on reference platforms with real system components. Testing is also executed in manufacturing facilities on each unit at die, package, and system levels.

Post-silicon validation involves operating manufactured chips in actual application environments to ensure correct behaviors across the specified operating conditions [Park-09]. Validation engineers execute a test plan aimed at detecting problems. When a problem is detected, triage and debug are performed to localize and root-cause the issue, so that the problem can be fixed by patching, tuning, or as a last resort, re-spinning.

The so-called disciplines of post-silicon validation include functional validation, power and performance validation, electrical validation and security validation. Functional validation aims to find logic bugs by running either synthetic content or stressing the system with applications running in commercial operating systems. Power and performance validation looks to ensure that the product complies with the power consumption targets as well as performance specifications. Electrical validation (EV) focuses on the analog performance of the input-output (I/O) links, and the electrical interaction with the other components on the board. Finally, security validation aims to ensure that the product has no security vulnerabilities that could be exploited in the field. Post-silicon validation is performed on a limited number of silicon units, prior to launching the product.

At the manufacturing level, testing is done on every fabricated chip, by applying input patterns, or test vectors, to the circuit and comparing its output against desired responses [Bushnell-02]. There are different types of tests: characterization tests, which verify that the device meets specifications against AC and DC parameters; production tests, which are typically brief to reduce cost, while still achieving a high coverage of possible faults; and burn-in tests, in which chips are subjected to production tests in a high temperature and over-voltage power supply to screen for infant mortality failures [Abdul-11].

Among the sub-components of the computing devices, I/O interfaces are of particular interest for post-silicon validation and testing. The electronics industry is continuously driving for ever-increasing data rates to fulfill the demand of data throughput. PCIe [PCISIG-16], for example, has evolved from 2.5 giga-transfers per second (GT/s) and 5 GT/s data rates in its first two generations, to 32 GT/s in the recent 5th generation, and 64 GT/s in the upcoming 6th generation. This increase poses particular challenges in the analog domain to ensure error-free communication as well as correct interoperability [Moreira-10]. Even though these I/O links are actually digital (sending and receiving 1’s and 0’s), the behavior is essentially analog when communicating at a multi-gigabit rate. Additionally, while on the digital or logical aspect, validation and testing have clear pass/fail criteria due to the discreteness of the expected outputs in both time and values, the analog side has further complexities not only to determine whether the behavior is correct or not, but also to localize and isolate specific faults [Kabisatpathy-05].

As mentioned before, EV is tasked with validating the analog performance of I/O links. Within EV, there are also different disciplines: system marginality validation (SMV) aims to provide a release qualification through a statistical analysis of receiver margins at a system level; signal integrity validation (SIV) is dedicated to performing tests to verify that the product meets all publicly available electrical compliance specifications; and design validation (DV) tests and characterizes the analog performance of individual circuits.

There are specific activities within EV that are challenging and time-consuming. Tuning, which is a common task between all EV disciplines, is performed to find the optimal equalization parameters that yield improved analog performance and is known to be one of the most time-consuming processes [Rangel-Patiño-16]. Jitter tolerance tests in SIV are also extremely slow when testing at the bit error rate (BER) specified by the industry standards. Fault detection and isolation in analog circuitry is extremely complex in test scenarios. Additionally, while BER extrapolation is a common technique to accelerate SMV tests, it can incur incorrect estimations which could affect the final qualification decision.

This doctoral dissertation presents several machine learning techniques, as well as numerical optimization approaches, to tackle these challenges. Machine learning algorithms build statistical models based on datasets to automatically make predictions. The objective of machine learning is to find underlying patterns in the given data [Yigit-17]. On the other hand, optimization algorithms aim at helping the task of decision making, by selecting the "best" choice among a number of alternatives. The measure of goodness among the alternatives is described by an objective function [Chong-96]. We employ these two types of algorithms to improve current methods and processes in analog validation and test. This doctoral dissertation is organized as follows.

Chapter 1 presents two optimization approaches to accelerate analog validation, one aimed at exploiting design of experiments techniques to optimize an eye diagram, and the other aimed at reducing the execution time of the jitter tolerance test.

Chapter 2 presents optimization techniques used in the context of tuning, employing surrogate models built with machine learning algorithms.

Chapter 3 provides an overview of analog faults concepts, a review of fault diagnosis methods, and proposes a methodology to identify analog gross faults using artificial neural network (ANN) models and parameter extraction, an optimization-based technique.

Chapter 4 proposes the use of deep neural models to classify the precision of margins extrapolated to a specific BER.

In the General Conclusions, the most relevant remarks about this doctoral dissertation are summarized, discussing the results of the proposed machine learning and optimization techniques. It also provides some opportunities for future research.

Finally, Appendix A shows the reference list of the thirteen internal research reports developed during these doctoral studies, and Appendix B shows the list of papers published during this same period of time.

1. Optimization Techniques to Accelerate Analog Validation

The continuous increase in data rates for high-speed input/output (HSIO) links demands equalization techniques to cancel any type of undesired effect induced in a transmitter-to-receiver interconnect [Rangel-Patiño-15]. As equalization complexity increases, so does the number of variables, thus incrementing the number of possible combinations of these variables. Design, as well as pre- and post-silicon validation, must find the means to find optimal parameter values in the fastest manner.

A proper design of experiments (DoE) is required in order to find the optimal equalization parameters executing the least number of tests. Failing to do so, leads to a waste of time and resources and may in turn have an impact on the competitiveness of a product [Rangel-Patiño-14]. Efficient experimental strategies must then be employed to overcome these challenges and comply with the ever-decreasing time to market. In this chapter, fundamental DoE techniques are first briefly described. These techniques are then applied to find the optimal equalization parameter values that maximize the eye opening of a HSIO link.

This chapter also describes receiver compliance tests acceleration. Post-silicon validation considers testing hundreds of silicon samples in realistic application environments, with the goal to check for robustness of the design by performing measurements on both receiver (Rx) and transmitter (Tx) circuitry of the HSIO links. These measurements have to comply with electrical standards and ensure that the design can operate under worst stressing conditions [Rangel-Patiño-20]. One of the most common ways to measure the performance of a HSIO link is by measuring the bit error rate (BER) through the link [Hong-08]. The fewer the errors measured, the better the performance of the link. BER measurement is typically used to characterize the Rx jitter tolerance (JTOL) performance in order to determine compliance with the industry standard specifications, such as XAUI [10GEA-16], PCIe [PCISIG-16], USB [USBOrg-16a], and SATA [SATAOrg-16]. The goal of JTOL is to verify that the Rx under test is capable to operate at a BER under worst case signaling conditions. The JTOL is usually measured with a BER tester instrument by sweeping the injected periodic jitter (J_P) amplitude across a range of frequencies until bit errors are detected. The test is considered to be passed when the measured error-free J_P amplitude is above the amplitude threshold defined by the protocol specification for each frequency point.

JTOL tests are very time-consuming when running at specification BER. For example, assuming a 95% confidence level and a target BER of 10^{-12}, it is necessary to transmit 3×10^{12} bits for each combination of J_P amplitude and frequency [Agilent-05]. This roughly translates to 10, 8.3, and 6.3 minutes per testing point for USB3.0 [USBOrg-16a], SATA3 [SATAOrg-16], and PCIe3 [PCISIG-16], respectively. Therefore, a full JTOL test can take several hours. Even when JTOL tests are only executed on a few units, it takes a large amount of time to achieve appropriate process, voltage, and temperature (PVT) coverage for a qualification decision.

Some alternatives for JTOL test time reduction have been previously reported. In [USBOrg-16b], a higher BER mask is presented as means to approximate a pass/fail criteria at a BER of 10^{-12} by slightly increasing the injected noise profile and executing the test at a BER of 10^{-10}. Another approach frequently used is to characterize the tolerated amplitude degradation between 10^{-10} and 10^{-12} tests. However, for both cases it is still needed to run a test at the specified BER to guarantee that the Rx passes the compliance test, mainly due to the lack of correlation between the 10^{-10} and 10^{-12} results. In [Fan-09a], an extrapolation algorithm for JTOL is proposed. The goal of this extrapolation algorithm is to predict the jitter tolerance at low BER based on high BER region data with the objective to reduce the JTOL testing time. However, this algorithm failed when verified in a post-silicon validation compliance environment, as opposed to being used in high volume manufacturing testers. Under the high variation of measurements typically seen in system compliance tests, the linear regression of the Q factor [Bergano-93] has a poor fit, which translates to poorly predicted values of J_P at low BER that do not correlate with real measurements.

This chapter presents a new approach to accelerate the JTOL testing based on the golden section optimization algorithm. The proposed method exploits the fast convergence of the golden section search with a high BER. The lack of correlation between different BERs is solved by performing a downward linear search at the actual target BER until no errors are seen. Our proposed approach is validated by testing it on two different HSIO links in a realistic server platform: SATA and USB3, demonstrating that the JTOL testing can be accelerated by 92.7% with respect to the current industrial practice.

This chapter revisits our work in [Viveros-Wacher-16] and [Viveros-Wacher-18a].

1.1. Design of Experiments

In science, the acquisition of knowledge is carried out using a process known as the scientific method. This iterative process involves making conjectures from an initial idea, also known as a hypothesis, testing such idea, and making deductions or drawing conclusions based on the data retrieved from the tests [Popper-02]. In most cases, there is no way to know beforehand whether the hypothesis is correct. The predictions made from the hypothesis must be tested by performing experiments [McKillup-03]. In a more formal manner, an experiment is a test or series of tests that deliberately change the inputs of a system to observe the reasons of change in the outputs of the system [Montgomery-14].

In any experiment, the results depend on the manner in which the data is obtained. Therefore, a proper planning and execution of experiments, along with the corresponding data analysis is of utmost importance to draw significant and valid conclusions. In engineering, DoE is a series of tools that allow the improvement of existing processes and products and the development of new ones. Based on statistical analysis, a DoE lets an engineer improve the performance of a process, select the key design parameters that affect its throughput, reduce variability and guarantee the compliance of certain output requirements of the process or system. In other words, a proper DoE is capable of characterizing a process (determine the factors that produce the greatest effect on the output), optimizing a process (determine the best values of the input factors that produce the desired outcome) and guaranteeing the robustness of a new design (determine a region of operation of a new product or process where its functionality is optimal regardless of disturbing or noisy factors).

1.1.1 Factorial Design

In any given system or process with *k* input variables, or factors, which have an influence on some output variables, or responses, there are many possible combinations of experiments to consider. With a large number of variables, a limited amount of resources or a reduced time to experiment, it might be unfeasible to carry out experimental strategies such as “one factor at a time”, where a baseline for all factors is selected and then each factor is successively swept, keeping all other factors at their base levels. Furthermore, these types of strategies mask any possible interaction between factors. A more suitable approach is known as a factorial experiment, where all variables are varied altogether. In a factorial design with k factors and n predefined levels for each factor, the number of required observations or experimental runs is n^k. The most efficient and simpler factorial designs include only two levels for each factor, thus reducing the number of observations to only 2^k. The two levels are chosen depending on the type of factor; for quantitative factors, such as temperature or voltage, two numerical values are chosen, whereas with qualitative factors, two different states are chosen, for example activating and deactivating a certain parameter. By randomizing the order in which the observations are made, a factorial design can be analyzed as a completely randomized design (CRD) and the effects can be studied using analysis of variance (ANOVA) and linear regression techniques [Lind-15].

1.1.2 Fractional Factorial Design

When the number of factors increase, the execution of a full factorial design can quickly surpass the resources allocated for its realization. Furthermore, the degrees of freedom corresponding to the main effects and the two-factor interactions are far lower than the total degrees of freedom. Assuming that the interactions between three or more factors are relatively insignificant, it is possible to obtain the most significant factors (or two-factor interactions) with only a fraction of a full factorial design.

A fractional factorial design with two levels for each factor is expressed as 2^{k-p}, where k is the number of factors under study and p describes the size of the fraction, given by 2^{-p}. Fractional factorial designs exploit the confounding technique, where the information of certain effects is indistinguishable or confounded with other effects. In other words, when analyzing a particular factor in a fractional design, such factor is also the alias of another effect. The method to determine the alias structure is by using a defining relation, which allows to find the p generators required.

A key characteristic of fractional factorial designs is their resolution: the ability to distinguish between the main effects and other interactions. In general, a resolution of *III* corresponds to a design where the main effects are not an alias of other main effects, but they are an alias of second order interactions; a resolution of *IV* corresponds to a design where none of the main effects is an alias of another main effect or of a second order interaction; and a resolution of *V* relates to a design where second order interactions are alias of third order interactions.

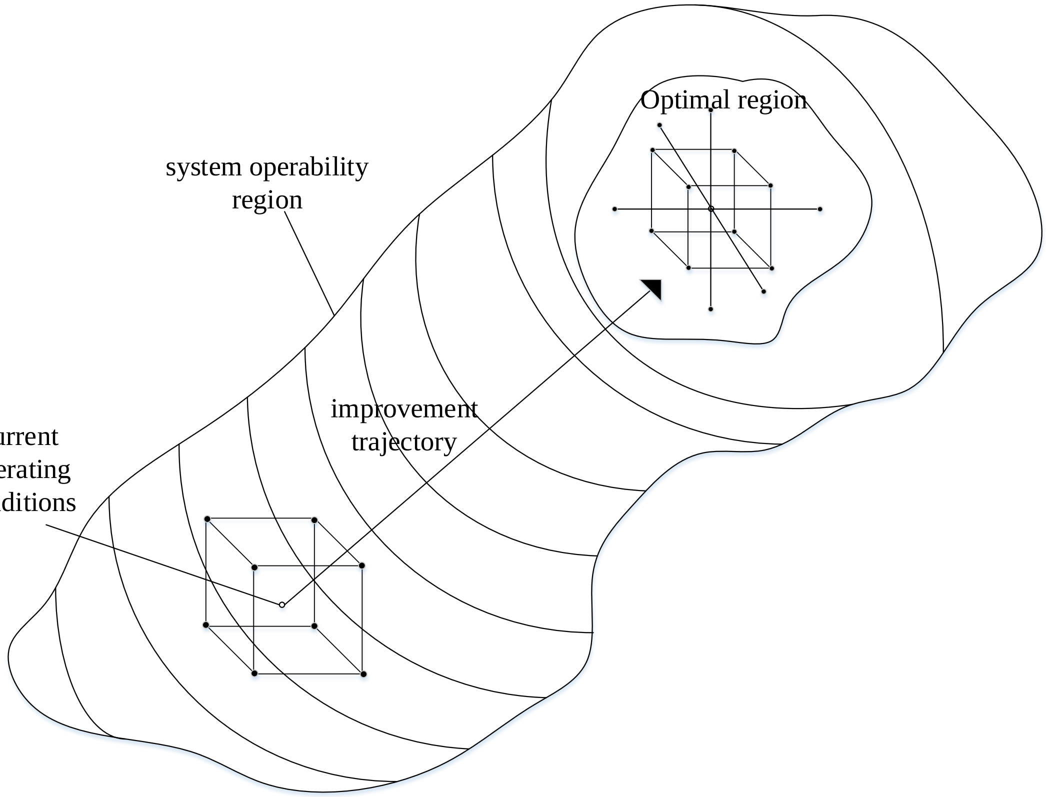


Fig. 1.1 Sequential nature of RSM. Figure taken from [Montgomery-14].

1.1.3 Response Surface Methodology and Central Composite Design

Response surface methodology (RSM) is a collection of techniques to model and analyze a response in terms of certain factors, where the main goal is to optimize such response. RSM is a sequential process, as depicted in Fig. 1.1. When the optimal point of operation is unknown, the first step in RSM is to generate a first order model, usually employing a factorial design. Once a first order model (in DoE terminology) is obtained, it can be used to determine the improvement trajectory, which points to the optimal region. In order to follow such trajectory, methods such as steepest ascent or steepest descent are economical and efficient approaches to find the optimal neighborhood. Finally, a more elaborate model, such as a central composite design (CCD), allows to find second order effects to determine the optimal point of the system under study.

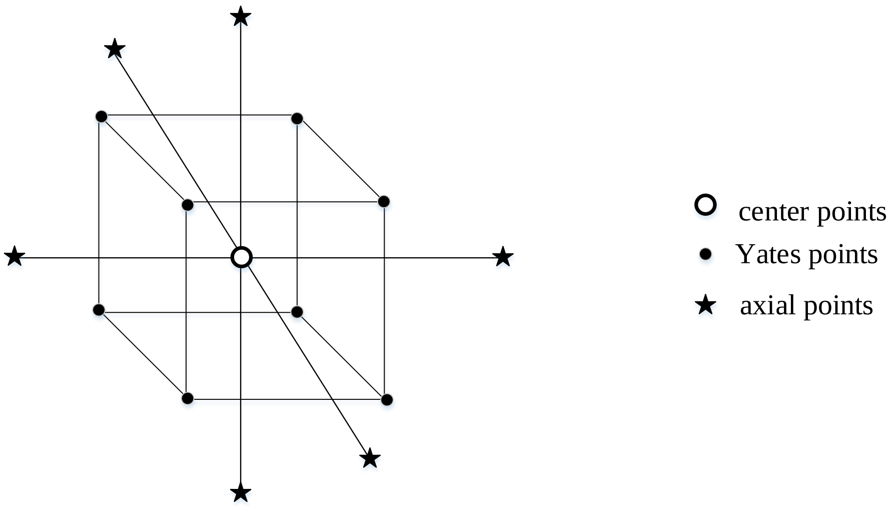


Fig. 1.2 Taxonomy of a central composite design (CCD).

1.1.4 Steepest Ascent Method

The steepest ascent method is a procedure to find the maximum response of a system. It consists of calculating the gradient of a function $\nabla f(x, y)$ starting from a given point $[x_0, y_0]$. This function can be determined by a first order model (in DoE terminology) of a fractional design. The direction of maximum slope is such that in each evaluation, the output of the system increases more rapidly. The size of the step to take during each evaluation is proportional to the regression coefficients of the first order model. It can be said that a maximum has been found when there is no increase on the response. A similar procedure, also known as steepest descent, can be performed to find the minimum response of a system. Once a maximum or a minimum has been found, a more precise optimum point can be found using a second order model, by employing a CCD.

1.1.5 Central Composite Design

CCD is the most popular class of designs to fit a second order model. It is composed of 3 main ingredients: 1) a 2^k fractional design (or alternatively, a fractional factorial design with resolution of *V*) with n_F runs, where n_F is the number of fractional runs, 2) $2k$ axial or star runs and 3) n_C runs at the center point, where $3 \leq n_C \leq 5$ for better error estimation. The resulting taxonomy of a CCD can be seen in Fig. 1.2. A key aspect to ensure a good prediction of the model is the

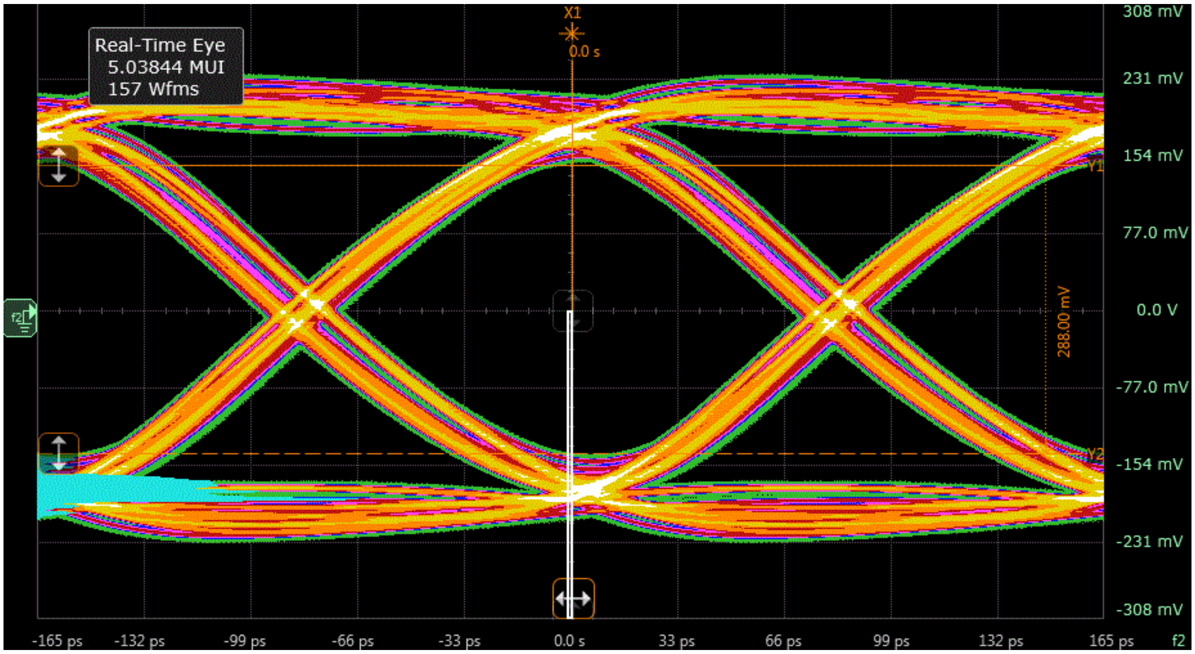


Fig. 1.3 Example of an eye diagram.

selection of the distance between the center and the axial points, known as α. To guarantee that the design is rotable (the variance of the predicted response is uniform across all points with the same distance to the center point), then $\alpha = (n_F)^{1/4}$. Furthermore, for a spherical CCD (all factorial and axial points are located in a sphere), then $\alpha = \sqrt{k}$.

1.2. Eye Diagrams in HSIO Links

A HSIO link eye diagram is the superposition of several bits into a single time-domain graph [Moreira-10]. Fig. 1.3 shows a typical eye diagram. The horizontal axis displays time, while the vertical axis shows the voltage amplitude of the signal under investigation. Important signaling parameters can be retrieved from an eye diagram, such as rise/fall time, total jitter, amplitude noise, minimum eye width and minimum eye height [Zhang-15]. Eye diagrams help to determine the correctness of the transmitter (Tx) signaling, as well as the ability of a receiver (Rx) to understand incoming data. When the sampling point (both in timing and voltage scales) of a receiver is positioned at the center of the eye, the probability of an error is low. Therefore, by increasing the eye width and eye height measurements of the eye, the probability of error decreases.

1.3. Equalization Techniques

Several undesired effects exist within a HSIO link: Tx jitter, channel-dependent sources such as channel loss, interconnect discontinuities and inter-symbol interference (ISI), EM interference and even noise sources within the Rx circuitry. The quality of an interconnection between a Tx and an Rx can be measured through an eye diagram. Correct reception depends on the ability of the Rx to decode incoming data regardless of the noise sources in the link. In order to guarantee proper link functionality, equalization techniques are employed to emphasize certain signaling components and suppress unwanted ones.

1.3.1 Transmitter Equalization

Equalization on the transmitter (Tx EQ) side is usually referred to as pre-emphasis/de-emphasis. It compensates loss by amplifying high frequency components (also known as pre-emphasis) and attenuating low frequency ones (de-emphasis). Tx EQ is usually carried out by implementing a finite impulse response (FIR) filter. Pre-emphasis is performed by increasing the voltage level of the first bit of a series of consecutive bits of equal value. On the other hand, de-emphasis decreases the voltage level of all but the first bit of a sequence of equal value bits. When the series of bits travels through a lossy channel, the forced distortions caused by the Tx EQ on the signals disappear, leaving the received signal shape closer to what was initially intended for transmission.

1.3.2 Receiver Equalization

Equalization at the receiver is a major research topic broadly documented in literature. Different EQ techniques have been proposed in digital and analog domains, in linear and non-linear manners and with feed-forward and feedback topologies [Fan-11]; at data rates as high as in HSIO links, several techniques can be combined to achieve best performance. The two most common circuital equalization techniques on the receiver side are the continuous time linear equalizer (CTLE) and the decision feedback equalizer (DFE). The main purpose of the CLTE is to counteract the channel loss and open the eye by amplifying signals near the Nyquist frequency.

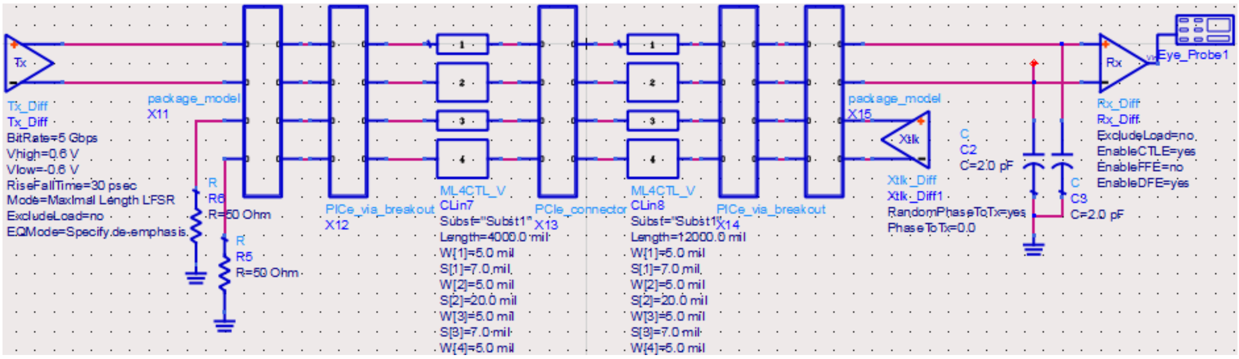


Fig. 1.4 HSIO system modeled in Keysight ADS. It is comprised of a Tx, an Rx, a crosstalk aggressor and a channel encompassing packages, vias, PCB traces and a connector.

Even though the CTLE provides a great means to better sample the incoming data by opening the eye, it is still susceptible to noise, given that it also amplifies unwanted signals at high frequency. A DFE is a non-linear EQ technique used to cancel the post-cursor ISI from the present bit by using previously received bits.

1.4. Rx Eye Area Maximization Methodology

The case under study considered in this report consists of a 5 Gbps HSIO link comprised of a Tx, a channel and an Rx. Fig. 1.4 shows the entire system under study. The channel is emulating a real interconnection including packages, vias, PCB traces, a connector and a crosstalk aggressor. The entire system is modeled and simulated using Keysight ADS[1]. Every part of the system introduces certain kinds of noise. In order for the system to work optimally, EQ techniques are used. On the Tx side, de-emphasis is applied, while on the Rx side, a CTLE with one zero and one pole and a 4-tap DFE are employed. Thus, in total there are seven EQ variables to be used for maximizing the Rx eye. The outputs of the system are the eye width, e_{w}, and eye height, e_{h}, measurements. The area of the eye is taken as a figure of merit for the optimization process and is simply calculated by

$$e_{\mathrm{A}} = e_{\mathrm{w}} e_{\mathrm{h}} \tag{1-1}$$

Noise factors are also included in the system; two Tx noise sources are used: random jitter with an amplitude of 7 mUI, and periodic jitter with 30 ps of amplitude at 20 MHz. A crosstalk

[1] Advanced Design Systems (ADS) ver. 2016.01, Keysight Technologies, 1400 Fountaingrove Parkway, Santa Rosa, CA.

TABLE 1.1. VARIABLES AND LEVELS DEFINITIONS FOR THE EQ FACTORS

Factor	Variable	+1	–1
Tx deemphasis	x_1	1 dB	3 dB
CTLE zero	x_2	–5 Grad/s	–7 Grad/s
CTLE pole	x_3	–7 Grad/s	–10 Grad/s
DFE Tap 1	x_4	0.0005	0.0009
DFE Tap 2	x_5	0.0015	0.0020
DFE Tap 3	x_6	–0.0015	–0.0020
DFE Tap 4	x_7	–0.0020	–0.0025

aggressor is included with a random jitter of 10 mUI of amplitude and a 50 ps periodic jitter at 20 MHz. Also, the PCB trace lengths are selected to emulate a long-channel topology commonly used in server applications.

The plan of action pursued in this work is comprised of three main steps: 1) perform a fractional factorial design to find the most significant variables; 2) perform an initial optimization using the steepest ascent method, and 3) use a CCD to improve the model and maximize the area of the eye diagram.

1.5. Results

Given that we have 7 EQ variables, a 2^{7-2}_{IV} fractional factorial design is selected because it provides a resolution of *IV*. By employing this design, only 32 runs are needed, whereas a full factorial requires 128 runs. The designs 2^{7-3} and 2^{7-4} require only 16 and 8 runs respectively, but their resolution is *III*, which introduces unwanted aliases. The design generators chosen are $x_6 = x_1 x_2 x_3 x_4$ and $x_7 = x_1 x_2 x_4 x_5$.

Table 1.1 presents the factor to variable mapping, along with the +1 and −1 level coding for each factor. The resulting design is shown in Table 1.2.

Fig. 1.5 depicts the Pareto chart resulting from the fractional factorial design, obtained using Minitab[2]. Even though it shows that all factors and most of the two-level interactions are

[2] Minitab version 17.3.1, Minitab, Inc, Pennsylvania, US, 2016.

TABLE 1.2. FRACTIONAL FACTORIAL DESIGN DEFINITION

StdOrder	RunOrder	CenterPt	Blocks	x_1	x_2	x_3	x_4	x_5	x_6	x_7
1	1	1	1	–1	–1	–1	–1	–1	1	1
2	2	1	1	1	–1	–1	–1	–1	–1	–1
3	3	1	1	–1	1	–1	–1	–1	–1	–1
4	4	1	1	1	1	–1	–1	–1	1	1
5	5	1	1	–1	–1	1	–1	–1	–1	1
6	6	1	1	1	–1	1	–1	–1	1	–1
7	7	1	1	–1	1	1	–1	–1	1	–1
8	8	1	1	1	1	1	–1	–1	–1	1
9	9	1	1	–1	–1	–1	1	–1	–1	–1
10	10	1	1	1	–1	–1	1	–1	1	1
11	11	1	1	–1	1	–1	1	–1	1	1
12	12	1	1	1	1	–1	1	–1	–1	–1
13	13	1	1	–1	–1	1	1	–1	1	–1
14	14	1	1	1	–1	1	1	–1	–1	1
15	15	1	1	–1	1	1	1	–1	–1	1
16	16	1	1	1	1	1	1	–1	1	–1
17	17	1	1	–1	–1	–1	–1	1	1	–1
18	18	1	1	1	–1	–1	–1	1	–1	1
19	19	1	1	–1	1	–1	–1	1	–1	1
20	20	1	1	1	1	–1	–1	1	1	–1
21	21	1	1	–1	–1	1	–1	1	–1	–1
22	22	1	1	1	–1	1	–1	1	1	1
23	23	1	1	–1	1	1	–1	1	1	1
24	24	1	1	1	1	1	–1	1	–1	–1
25	25	1	1	–1	–1	–1	1	1	–1	1
26	26	1	1	1	–1	–1	1	1	1	–1
27	27	1	1	–1	1	–1	1	1	1	–1
28	28	1	1	1	1	–1	1	1	–1	1
29	29	1	1	–1	–1	1	1	1	1	1
30	30	1	1	1	–1	1	1	1	–1	–1
31	31	1	1	–1	1	1	1	1	–1	–1

significant, it is clearly seen that A, B and C, along with the two-level interactions between them, are the most significant. Therefore, those three factors are chosen for the next step.

In order to better maximize the output, the other four factors were selected as follows: x_4 in (–) level, x_5 in (+) level, x_6 in (–) level and x_7 in (–) level. After these considerations, the resulting first-order RSM (in DoE terminology) is

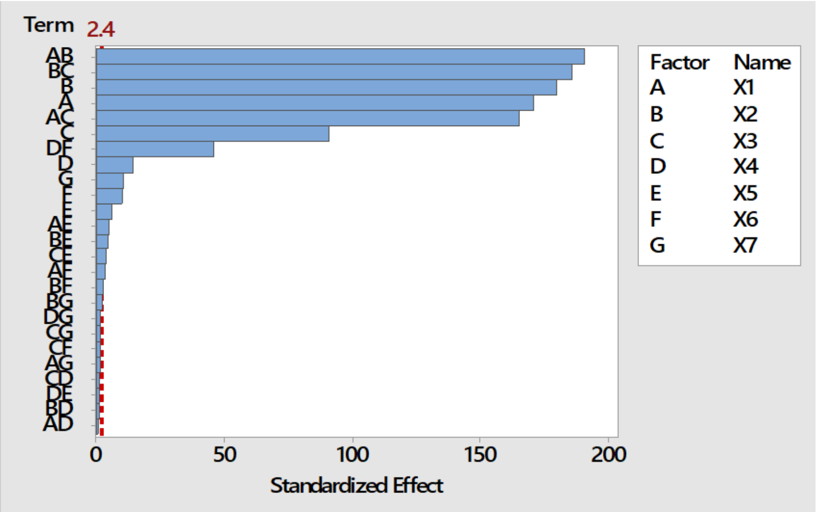


Fig. 1.5 Pareto chart from the fractional factorial design analysis.

$$e_{\mathrm{A}}(\boldsymbol{x}) = 10048 - 130.7x_1 + 135.6x_2 - 193.3x_3 + 1019x_1x_2 - 637.5x_1x_3 + 843.2x_2x_3 - 85.6x_1x_2x_3 \qquad (1\text{-}2)$$

Following the steepest ascent methodology, the gradient of ((1-2) was calculated, obtaining the relative effects of each variable, as

$$b_{x_1} = \frac{\partial \mathrm{e_A}}{\partial \mathrm{x_1}} = \text{-}165.2 \qquad (1\text{-}3)$$

$$b_{x_2} = \frac{\partial \mathrm{e_A}}{\partial \mathrm{x_2}} = 1912.2 \qquad (1\text{-}4)$$

$$b_{x_3} = \frac{\partial \mathrm{e_A}}{\partial \mathrm{x_3}} = \text{-}73.2 \qquad (1\text{-}5)$$

Given that b_{x2} presents the largest value, x_2 is selected to dictate the step size for the following experiments, while x_1 and x_3 are varied,

$$\Delta x_2 = 0.5 \qquad (1\text{-}6)$$

$$\Delta x_1 = \frac{\mathrm{b}_{\mathrm{x}_1}}{\mathrm{b}_{\mathrm{x}_2}} \Delta x_2 \qquad (1\text{-}7)$$

$$\Delta x_3 = \frac{\mathrm{b}_{\mathrm{x}_3}}{\mathrm{b}_{\mathrm{x}_2}} \Delta x_2 \qquad (1\text{-}8)$$

Results for the steepest ascent execution are shown in Fig. 1.6. As it can be seen, the values corresponding to the 5th experimental run provide the maximum eye area. These values are selected as the center point for the CCD.

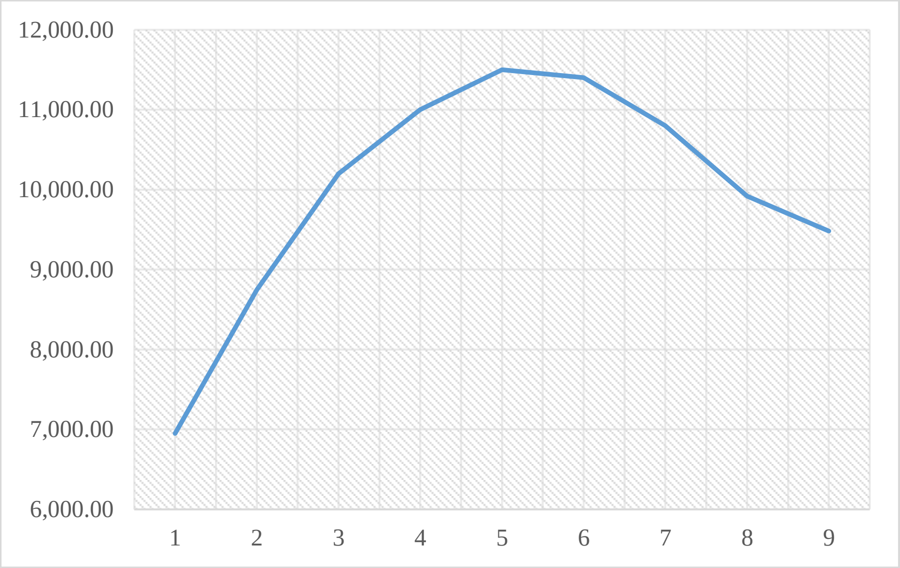


Fig. 1.6 Sequential results for the steepest ascent execution.

TABLE 1.3. CENTRAL COMPOSITE DESIGN DEFINITION

StdOrder	RunOrder	PtType	Blocks	x_1	x_2	x_3
18	1	0	1	0.00000	0.00000	0.00000
6	2	1	1	1.00000	–1.00000	1.00000
16	3	0	1	0.00000	0.00000	0.00000
14	4	–1	1	0.00000	0.00000	1.68179
10	5	–1	1	1.68179	0.00000	0.00000
2	6	1	1	1.00000	–1.00000	–1.00000
7	7	1	1	–1.00000	1.00000	1.00000
17	8	0	1	0.00000	0.00000	0.00000
20	9	0	1	0.00000	0.00000	0.00000
5	10	1	1	–1.00000	–1.00000	1.00000
12	11	–1	1	0.00000	1.68179	0.00000
19	12	0	1	0.00000	0.00000	0.00000
15	13	0	1	0.00000	0.00000	0.00000
4	14	1	1	1.00000	1.00000	–1.00000
11	15	–1	1	0.00000	–1.68179	0.00000
8	16	1	1	1.00000	1.00000	1.00000
13	17	–1	1	0.00000	0.00000	–1.68179
1	18	1	1	–1.00000	–1.00000	–1.00000
9	19	–1	1	–1.68179	0.00000	0.00000
3	20	1	1	–1.00000	1.00000	–1.00000

The CCD created with Minitab using factors x_1, x_2 and x_3 with α = 1.681793 is shown in Table 1.3. The resulting second model order for the system under study is

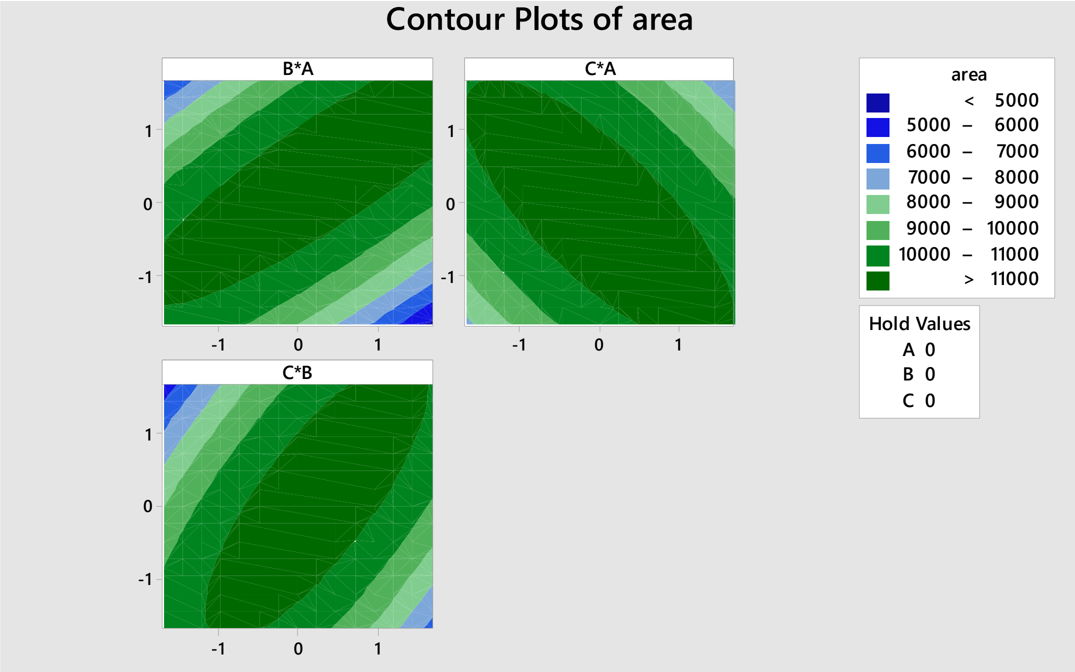


Fig. 1.7 Contour plots resulting from the CCD execution for the eye diagram area.

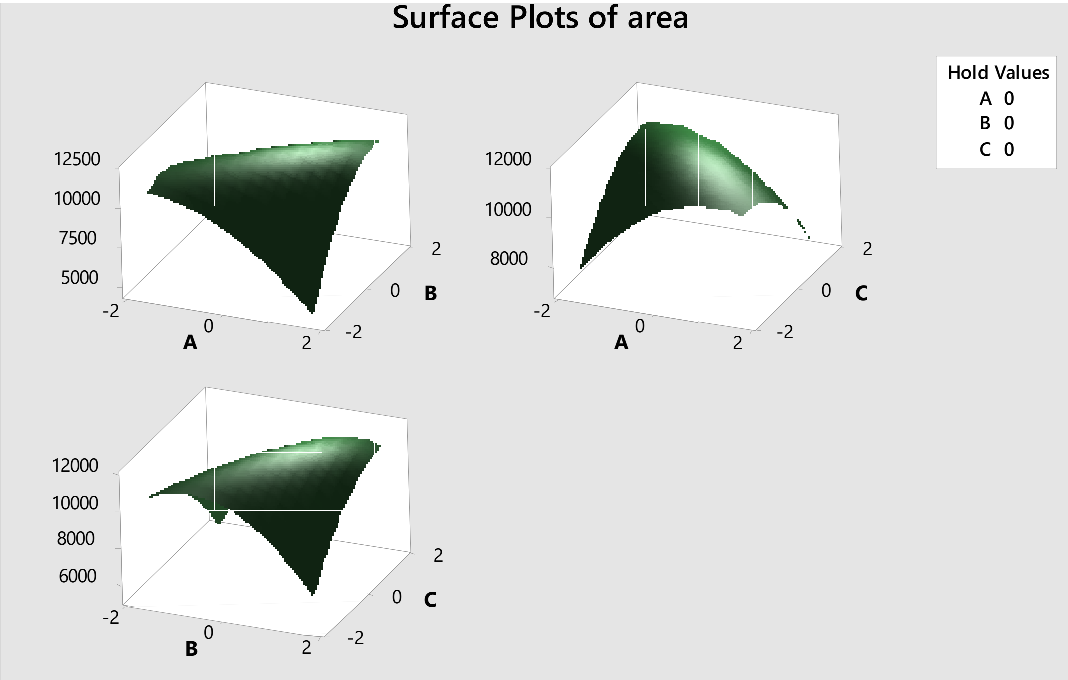
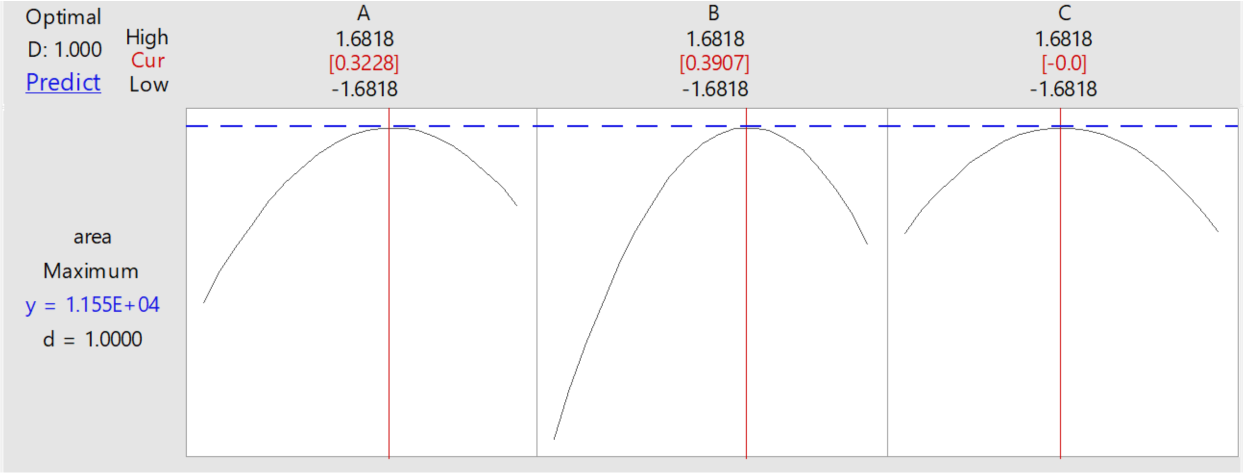


Fig. 1.8 Surface plots resulting from the CCD execution for the eye diagram area.

$$e_{\mathrm{A}}(\boldsymbol{x}) = 11511 - 104.5x_1 + 262x_2 - 118.8x_3 - 445.7x_1^2 - 383.6x_3^2 + 1018.8x_1x_2 \\ - 637.5x_1x_3 + 843.2x_2x_3 \tag{1-9}$$

Fig. 1.7 and Fig. 1.8 show the contour and surface plots of the model. Under careful study, the maximum point of the system can be inferred from these plots. However, by using the built-in

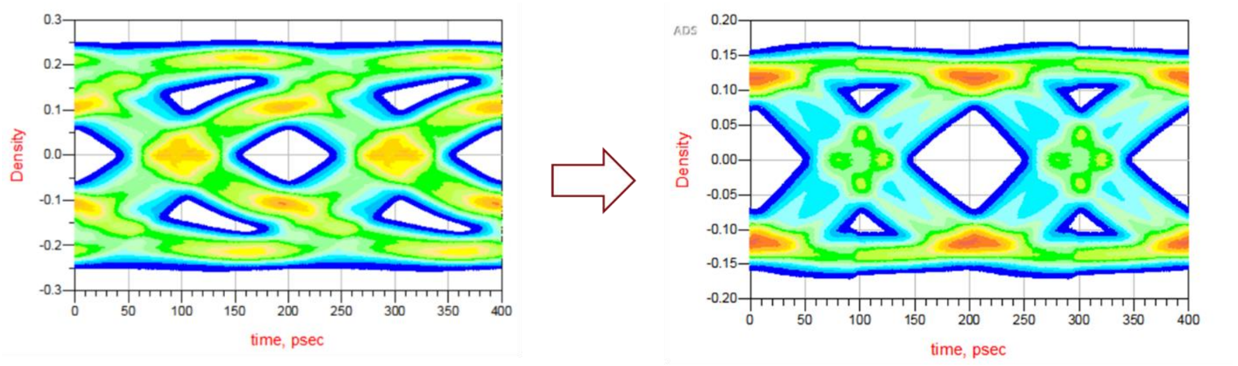


Optimal
D: 1.000
Predict

	A	B	C
High	1.6818	1.6818	1.6818
Cur	[0.3228]	[0.3907]	[-0.0]
Low	-1.6818	-1.6818	-1.6818

area
Maximum
y = 1.155E+04
d = 1.0000

Fig. 1.9 Minitab optimizer output showing the optimal values for factors A, B and C, as well as the predicted eye area.



Measurement	Value
Eye width	74.8ps
Eye height	56mV
Area	4188.8

Measurement	Value
Eye width	96.4ps
Eye height	119mV
Area	11471.6

Fig. 1.10 Simulation results for the initial EQ values (shown on the left side of the figure) and the optimal EQ values found (shown on the right side of the figure).

Minitab optimizer, a more accurate set of variable values is obtained, as well as a prediction of the area, as seen in Fig. 1.9. Based on this, the system was simulated using the optimal EQ values. Results of this simulation are depicted in Fig. 1.10, along with the initial eye for comparison purposes and the corresponding measurement values. The measured eye area with the optimal settings increased 173% with respect to the original EQ values. Furthermore, the total number of simulations to arrive at the optimal point were 61, which is barely 46.21% of the number of simulations required to run a full factorial design.

1.6. Post-Silicon Validation

Post silicon validation is the stage from the product development of modern computer platforms where silicon is available at the laboratory and validation is performed by several groups or disciplines, such as functional validation (FV), bench design validation (BDV), and electrical validation (EV), among so many others. All of these disciplines execute validation in parallel, each one focusing on different specifications, with the objective of qualifying a product over different operation conditions, process corners, and usage models [Gu-12].

Electrical validation focuses on the validation of several analog phenomena, such as the tuning of the so-called physical layer (PHY) [Rangel-Patiño-16], validation of the electrical parameters from the I/O links, power delivery, and clocks, as well as resilience to noise impairments such as crosstalk, inter symbol interference (ISI), etc.

HSIO interfaces have a Tx that sends a serial stream of bits with an embedded clock through a channel to the Rx. The Rx receives the incoming high-speed serial data, extracts the embedded clock, and determines a logical one or a logical zero for each bit received in the stream for further processing at the upper protocol layers.

Given that the data rate of HSIO interfaces is on the order of several gigabits per second (Gbps), the specifications associated to those interfaces regarding the timing budget is very stringent. This timing budget is reflected on several jitter specifications. On the Tx side, it is specified how much timing deviation the Tx can generate to consume the total jitter budget. Whereas on the Rx side, it specifies how much time deviation the Rx should tolerate before a false detection occurs. As noted in [Fan-09b], “The traditional guaranteed by design paradigm cannot be applied anymore”. Hence, the chip maker companies invest a lot of resources to do an exhaustive validation on the tight timing specifications to ensure their design and chip quality.

Many HSIO standards define the jitter performance at the BER of 10^{-12}, which requires a very large amount of time to get a statistical valid measurement. Testing a representative number of parts on different operation conditions can increase the time exponentially, which cannot be afforded for the reduced validation times, which are continuously aimed to be reduced to achieve a competitive TTM. Hence, the need of suitable optimization algorithms to reduce the Rx JTOL testing time without compromising the quality on the validation becomes highly relevant.

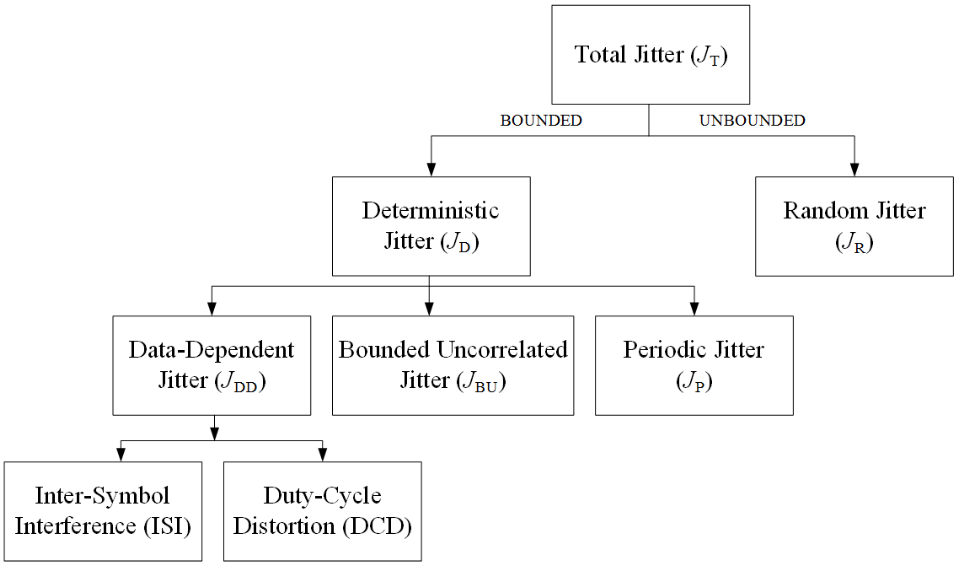

Total Jitter (J_T)
BOUNDED
UNBOUNDED
Deterministic Jitter (J_D)
Random Jitter (J_R)
Data-Dependent Jitter (J_{DD})
Bounded Uncorrelated Jitter (J_{BU})
Periodic Jitter (J_P)
Inter-Symbol Interference (ISI)
Duty-Cycle Distortion (DCD)

Fig. 1.11 The jitter components classification.

1.7. Jitter Tolerance Testing

Jitter is the variation in time of a periodical signal [Athavale-05]. In general, jitter sources are classified as random jitter (J_R) and deterministic jitter (J_D), which combined form the total jitter (J_T). The J_R is caused by stochastic unbounded sources or events and it can be characterized by a Gaussian distribution [Fan-11] and [Ham-05]. On the other hand, the J_D sources can be classified as deterministic events with bounded peak-to-peak values. The J_D is categorized into periodic jitter (J_P) or sinusoidal jitter, bounded uncorrelated jitter (J_{BU}), and data dependent jitter (J_{DD}) [Kuo-04]. The J_P is caused by the periodic variation of a signal from sources with repetitive noise, while the J_{BU} is typically associated to coupling from adjacent signal traces or randomly switching logic located on-chip. The variation of jitter that occurs in the same signal traces and depends on the transmitted pattern corresponds to J_{DD}; it is classified in two components: the duty-cycle distortion (DCD), which is the jitter produced by the inequality of the logic values of the high and low pulses widths, and the jitter caused by the bandwidth limitations and the signal traces losses triggered by the frequency, also known as inter-symbol interference (ISI). The jitter components classification is shown in Fig. 1.11 [Fan-11] and [Kuo-04].

HSIO links specifications require the measurement of jitter components, which can be performed by the use of different techniques, such as time interval error (TIE) measurement, jitter histograms, JTOL, and BER bathtub, among others. Describing the measurement and characterization of each jitter components is beyond of the scope of this report. However, to perform a JTOL test, each protocol specification defines a calibration procedure prior to the JTOL execution. This procedure defines the specific J_{R}, J_{BU}, DCD, and ISI values injected to the test pattern, which remain constant throughout the JTOL test, while J_{P} is varied in both amplitude and frequency. The JTOL response R_{J} can therefore be defined as

$$R_{\mathrm{J}} = u(J_{\mathrm{P}}, f) \qquad (1\text{-}10)$$

where J_{P} is the periodic jitter amplitude injected by the BER tester, and f is the frequency of the periodic jitter. The evaluation of u implies sending a certain amount of bits from the BER tester, receiving the data stream at the Rx of the device under test (DUT), looping back the data to the Tx of the DUT, and receiving it once again at the BER tester to check for bit errors. The equipment then computes the BER and returns a PASS if the measurement is above the target BER, or FAIL if there were more errors that those allowed to comply with the target BER. Therefore, u is a discrete function with continuous variables, and R_{J} is digital, since it can only have a PASS or a FAIL value. Given that there is usually a well-defined frontier between J_{P} values that yield a PASS and J_{P} values that yield a FAIL, u can be considered a unimodal function. Also, for a fixed value of f, the problem of finding the largest value of J_{P} that yields a PASS becomes a unidimensional optimization problem.

In the traditional way to run JTOL, at each frequency point, the value of J_{P} is initialized at the starting point, where we are guaranteed to have a PASS result from the BER test. Then, J_{P} is increased a certain amount, typically equivalent to the minimum value allowed by the BERT equipment for best accuracy. Then, a test is performed at the compliance BER. This is iteratively done until the BER test yields a FAIL. The result reported at each frequency point is the last J_{P} value that yields a PASS.

1.8. JTOL Optimization

The proposed algorithm to optimize the JTOL testing time is divided in two main stages: 1) execution of a linear search method based on the golden section [Chong-96] at a high BER, typically 10^{-11}, and 2) a downwards search at the compliance BER, i.e. 10^{-12}, starting from the value obtained from the previous step. This technique clearly takes advantage of the fast execution

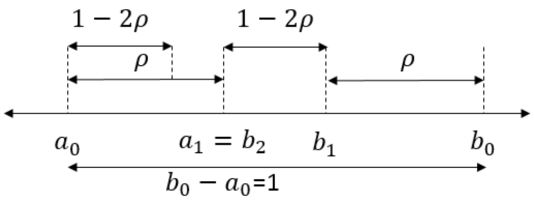


Fig. 1.12 Nature of the range reduction of the golden section algorithm.

```
$\alpha_j^a = \alpha_j^{lb} + \rho(\alpha_j^{ub} - \alpha_j^{lb})$
$\alpha_j^b = \alpha_j^{lb} + (1-\rho)(\alpha_j^{ub} - \alpha_j^{lb})$
evaluate $u(\alpha_j^a)$ and $u(\alpha_j^b)$
repeat until *StoppingCriteria*
  if $u(\alpha_j^a) = \text{PASS} \wedge u(\alpha_j^b) = \text{FAIL}$
    $\alpha_{j+1}^{ub} = \alpha_j^b \,;\, \alpha_{j+1}^{lb} = \alpha_j^{lb}$
    $\alpha_{j+1}^b = \alpha_j^a \,;\, \alpha_{j+1}^a = \alpha_{j+1}^{lb} + \rho(\alpha_{j+1}^{ub} - \alpha_{j+1}^{lb})$
    $u(\alpha_{j+1}^b) = u(\alpha_j^a)$
    evaluate $u(\alpha_{j+1}^a)$
  else if $u(\alpha_j^a) = \text{PASS} \wedge u(\alpha_j^b) = \text{PASS}$
    $\alpha_{j+1}^{lb} = \alpha_j^a \,;\, \alpha_{j+1}^{ub} = \alpha_j^{ub}$
    $\alpha_{j+1}^a = \alpha_j^b \,;\, \alpha_{j+1}^b = \alpha_{j+1}^{lb} + (1-\rho)(\alpha_{j+1}^{ub} - \alpha_{j+1}^{lb})$
    $u(\alpha_{j+1}^a) = u(\alpha_j^b)$
    evaluate $u(\alpha_{j+1}^b)$
  $j = j + 1$
$\alpha^* = (\alpha_j^{lb} + \alpha_j^{ub})/2$
```

Fig. 1.13 Pseudo code implementation of the golden section algorithm.

of the golden section search with a high BER, while overcoming the lack of correlation between different BERs by performing a downward linear search at the actual target BER until no errors are seen.

1.8.1 Golden Section Search

The golden section search algorithm is one of the most widely used unidimensional search methods. It aims to find the minimum value of a unimodal function f: $\Re \rightarrow \Re$. In order to do so, it is necessary to delimit the search points of the function to [a_0, b_0] as shown in Fig. 1.12, defined

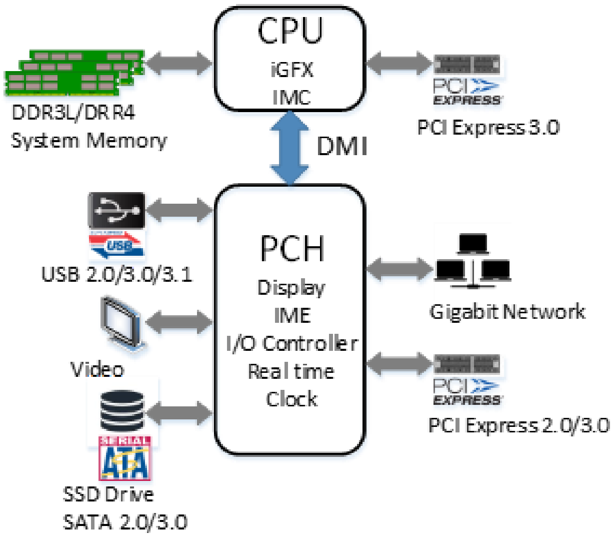


Fig. 1.14 Architecture of the system under test, including the PCH and CPU, as well as the USB, SATA and PCIe HSIO links. Figure taken from [Rangel-Patiño-20].

as the lower bound (*lb*) and upper bound (*ub*) search points. To reduce the range of uncertainty, the function is evaluated in intermediate points, a_1 and b_1, which are symmetrically selected by applying the golden section rule in such a way that a_1 is at a distance of ρ to a_0 and $(1-\rho)$ to b_0 while b_1 is at a distance of $(1-\rho)$ to a_0 and ρ to b_0, where $\rho = (3-\sqrt{5})/2$ is the so-called golden ratio. The range reduction is then accomplished by comparing the function evaluations of the intermediate points. If $u(a_1) < u(b_1)$, the minimum value must be in the new range of $[a_0, b_1]$; however, if $u(a_1) \geq u(b_1)$ then the minimum value lies in the range $[a_1, b_0]$. This process is iterated, as depicted in Fig. 1.13, until the following stopping criteria is met:

$$\left(\alpha^{ub} - \alpha^{lb}\right) \leq \varepsilon_{\text{step}} \qquad \text{(1-11)}$$

where $\varepsilon_{\text{step}}$ is defined as either the minimum J_P increment allowed by the BER tester or the known measurement to measurement variability. In JTOL, the evaluation of the function returns a PASS or FAIL response from the BER test at a certain J_P and f. This implementation of the golden section algorithm therefore differs from the classical one, in the sense that it is based on a discrete function response rather than a continuous one. Consequently, the decision of which search range to discard is based on the possibility to approach on every iteration the actual boundary between the PASS and FAIL responses, which is our optimal point. The value returned by the algorithm, α^*, is the average between the upper bound and lower bound points from the last iteration.

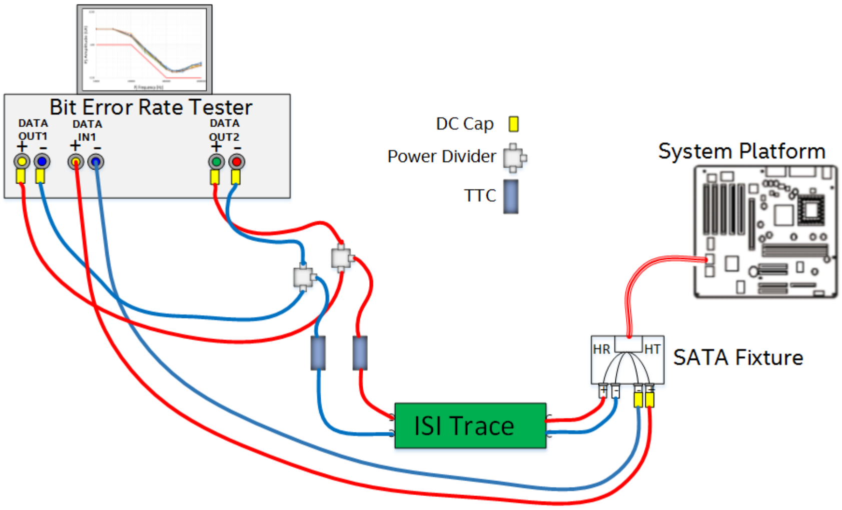


Fig. 1.15 SATA JTOL test setup.

1.8.2 Downwards Search

The second stage of the proposed algorithm performs a search starting from α^* from the previous stage, but now executing at the compliance BER. The search is performed in a downwards direction, meaning that the J_P is decremented in linear steps equivalent to ε_{step} (or a percentage of ε_{step} in accordance to precision used in the traditional method) until no errors are seen, or in other words, until the BER test passes. The range reduction achieved by the golden section search allows to decrease the number of evaluations in the downwards search. Typically only one to three evaluations are needed at the compliance BER, thus the overall test time is dramatically reduced.

1.9. Test Cases

The proposed methodology in this report was tested in two different HSIO links: SATA3 and USB3 super-speed Gen1. These links are part of an Intel platform controller hub (PCH) that works in conjunction with the CPU through the direct media interface (DMI) on a server platform, as shown in Fig. 1.14 [Rangel-Patiño-20].

1.9.1 Test Case 1: SATA3

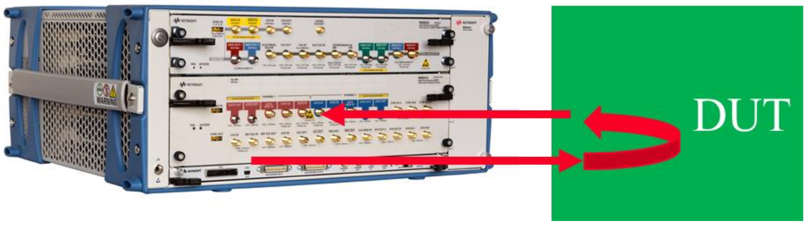


Fig. 1.16 JTOL setup showing BERT and loopback in DUT.

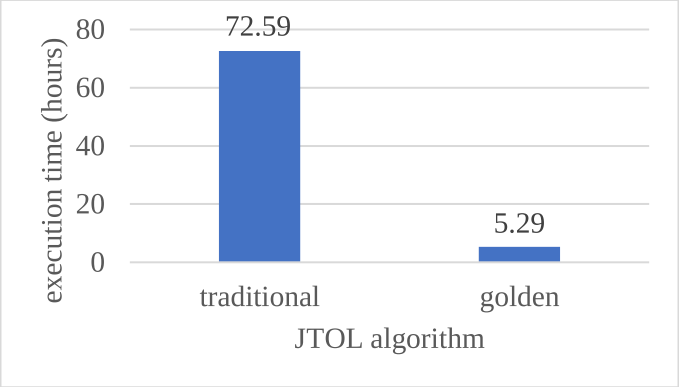


Fig. 1.17 SATA JTOL execution time comparison.

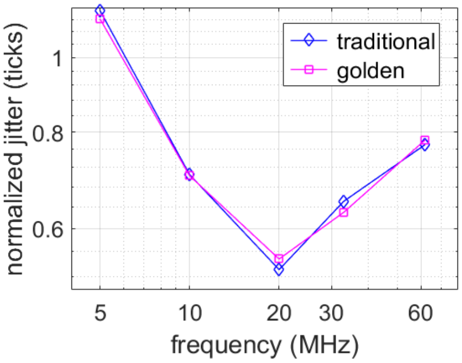


Fig. 1.18 SATA JTOL results for the traditional approach and the proposed golden section algorithm implementation.

The JTOL setup for SATA is comprised of a system platform which includes the DUT and a SATA connector, as well as a SATA3 Fixture, a SATA ISI Channel, two transition time converters, two power dividers and DC blocking capacitors, as shown in Fig. 1.15. During the Rx JTOL test, the BERT pattern generator sends a compliance test pattern with added jitter through the compliance channels connected with fixtures the to the Rx. Prior to running the test, the port should transition to loopback state. Once in loopback, the data received from the DUT is compared to the data generated and errors are counted by the BERT, as seen in Fig. 1.16. The JTOL execution following our proposal took 5.29 hours to complete 3 repetitions at five different frequency points, as compared to 72.59 hours that the traditional method requires (see Fig. 1.17).

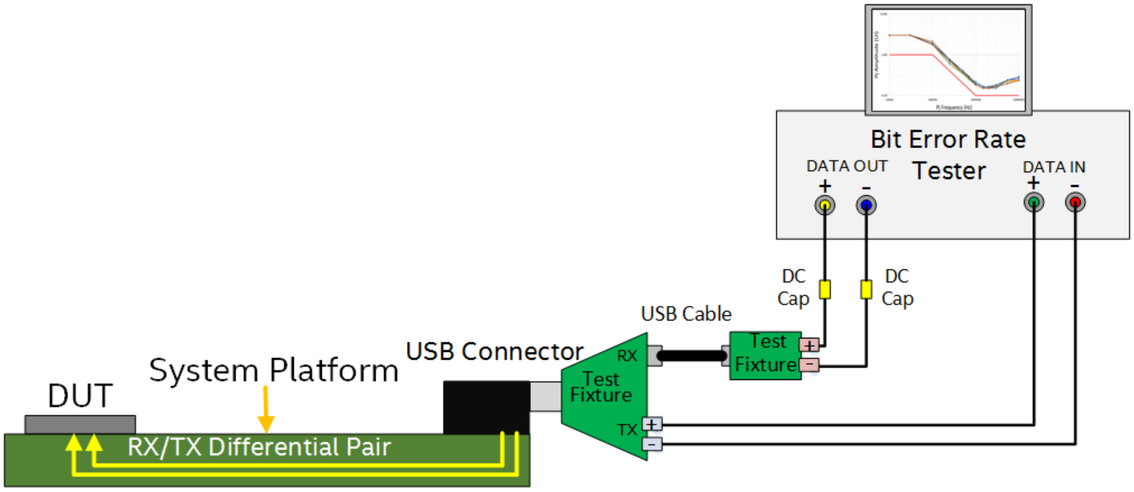


Fig. 1.19 USB3 JTOL test setup.

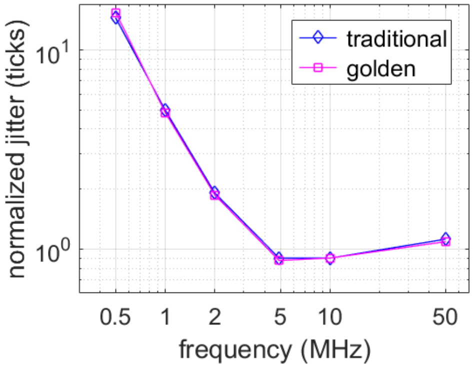


Fig. 1.20 USB3 JTOL results for the traditional approach and the proposed golden section algorithm implementation.

In other words, our proposal is 92.7% faster than the traditional approach to reach a comparable solution, as shown in Fig. 1.18.

1.9.2 Test Case 2: USB3

In the case of USB3, the JTOL setup includes the system platform which contains the DUT and a USB3 connector where USB3 fixtures are inserted, one BERT and DC blocking capacitors, as shown in Fig. 1.19. Validation time was significantly decreased using our JTOL algorithm as compared with the traditional methods, without sacrificing accuracy.

In this test case, the JTOL results for the traditional approach and those with the proposed golden section algorithm were practically the same, as shown in Fig. 1.20, however, the validation time was reduced by up to 96.09% with respect to the traditional approach. Current methods

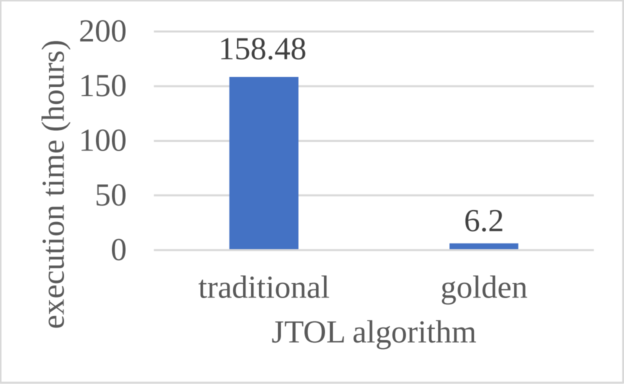


Fig. 1.21 USB JTOL execution time comparison.

require around 6 days for a complete execution, while the method proposed in this work can be completed in a few hours, as illustrated in Fig. 1.21.

1.10. Conclusions

DoE has proven to be a powerful set of statistical tools to efficiently find the optimal performance of a system. With the use of a fractional factorial design, the most significant EQ variables were identified with respect to the resulting Rx eye area measured. The steepest ascent method was used to find the optimal region of operation and by using the central composite design, a second order model of the system was found, which aided in finding the appropriate EQ values to obtain the maximum eye area under the specified conditions of operation.

Additionally, the golden section search algorithm has proven its effectiveness on reducing the jitter tolerance validation time as compared with the traditional method: it is around 95% faster without compromising the accuracy on the measurements. Even though the algorithm was tested on the SATA and USB3 standards as a proof of concept, the proposed algorithmic approach could easily be ported to be used on other standards, such as XAUI, and PCIe, among others. The incorporation of the golden section search algorithm to the post silicon JTOL tests allows, on one hand, a reduction of TTM by getting the evaluation of the silicon sooner, and on the other hand, an increase in the validation quality by achieving more unit coverage or PVT conditions at lower execution cost.

2. Surrogate based Modeling and Design Optimization of High-Performance Physical Platforms

The combined effects of increased product complexity, performance requirements, and time-to-market (TTM) commitments have added tremendous pressure on post-silicon validation [Keshava-10]. Within the computer server segment, there are conditions that further increase system complexities. These include increased I/O density, decreased power consumption, as well as non-flexible form factors [Lee-11]. The latter implies that channel designs remain unchanged, thus turning the problem towards analog circuitry optimization. Therefore, physical layer (PHY) tuning based on equalization techniques are used to cancel any undesired effect, such as transmitter jitter, attenuation, or inter-symbol interference, among others [Hodgkiss-83] and [Zhang-15]. Current industrial practices to perform PHY tuning essentially consist of an exhaustive enumeration method, turning them into the most time-consuming processes in post-silicon validation [Keshava-10], [Wang-15] and [Cheng-11]. To perform PHY tuning, the receiver (Rx) eye diagram margins [Viveros-Wacher-14] are optimized until compliance of the link specifications. Accurate direct simulations for PHY tuning in high-speed input/output (HSIO) links are computationally very expensive given the complexity of the system involved. On the other hand, surrogate models are scalable mathematical models that can be used as a parameterized approximation of a system response within a design space of interest [Yelten-12] and [Garistelov-12]. While an accurate surrogate model is desirable for direct surrogate-based optimization (SBO), it can be very expensive to derive it, since it typically requires massive lab measurements which are prohibitive under the current TTM schedules. However, by combining a good model vehicle with a suitable design of experiments (DoE) approach, more efficient surrogate modeling approaches can be developed.

In this chapter, coarse surrogate models of an HSIO link based on actual measurements of an industrial server post-silicon validation platform are developed. Several surrogate modeling techniques combined with different DoE approaches are compared to find the best coarse model, verifying the response of the resultant coarse models by comparing with actual measurements. We

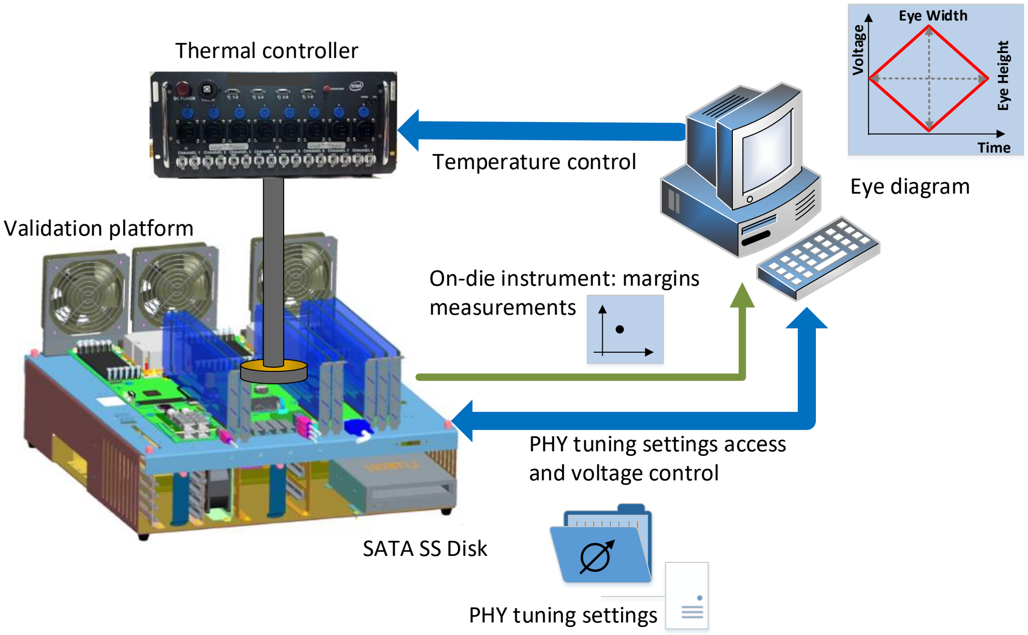


Fig. 2.1 Test setup: an Intel server post-silicon validation platform. Image taken from [Rangel-Patiño-17b].

next perform a surrogate-based optimization (SBO) with the best coarse models found to obtain the optimal PHY tuning Rx equalizer settings. We finally validate our approach by measuring the actual functional eye diagram on the real system using the optimal settings predicted by the coarse model.

Additionally, this chapter addresses a machine learning-based metamodeling technique to develop a coarse model. More specifically, a metamodeling approach is proposed, based on artificial neural networks (ANN), to efficiently simulate the silicon equalizer circuitry of the Rx. The model is generated using a frugal set of training data exploiting several design of experiments (DoE) approaches to reduce the number of test cases. We evaluate the neural model performance by comparing with actual measured responses on an industrial server validation platform.

While an accurate surrogate model is desirable for direct surrogate-based optimization (SBO), it can be computationally expensive to develop. By combining an adequate modeling technique with a suitable DoE approach, a coarse surrogate model can be efficiently developed with a very reduced set of data, as in [Rangel-Patiño-17c] and [Rangel-Patiño-19]. Once this coarse model is available, space mapping (SM) techniques can be exploited. In the present chapter, the Broyden-based input space mapping optimization algorithm, better known as aggressive SM (ASM) [Bandler-95] and [Rayas-Sánchez-16], is used for the first time in HSIO PHY tuning optimization. The proposed SM approach takes advantage of a coarse surrogate model developed

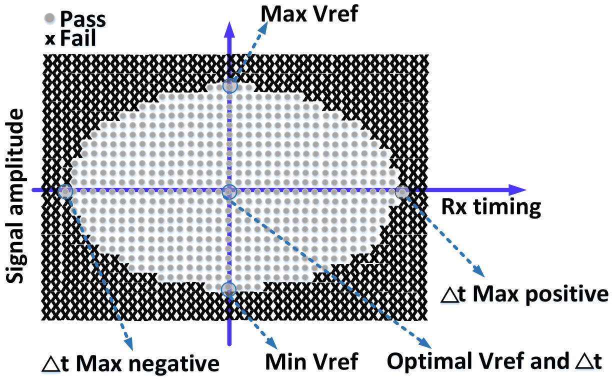


Fig. 2.2 Functional eye diagram based on system margin validation. Image taken from [Rangel-Patiño-17b].

following [Rangel-Patiño-17c]. In this case, the fine model is a measurement-based post-silicon validation industrial platform. Our approach is illustrated by optimizing the PHY tuning receiver (Rx) equalizer settings for a SATA Gen 3 channel topology, accelerating tuning from several days to a few hours.

This chapter revisits our work in [Rangel-Patiño-17b], [Rangel-Patiño-18], and [Rangel-Patiño-19].

2.1. System Description

The system under test is an Intel server post-silicon validation platform in an industrial environment, as shown in Fig. 2.1. The platform is comprised mainly of a CPU and a platform controller hub (PCH). Within the PCH, our methodology was tested on a SATA Gen3 HSIO link [SATAOrg-16]. The SATA Rx eye diagram is measured by a process known as system margin validation (SMV). The functional eye diagram measurements in SMV rely on on-die design for test (DFT) features that shrink the eye opening up to a point where the Rx detects errors or the system fails, as illustrated in Fig. 2.2.

2.2. Design of Experiments

A large amount of training and testing data is usually needed to ensure surrogate model accuracy. However, generating large amounts of data is very expensive in the post-silicon validation environment. DoE can be exploited to reduce the dimension of these data sets, ensuring adequate parameter coverage [Mack-07]. Here we use DoE to sample the complete design space in an efficient manner by selecting a relatively small number of base points. With *k* variables and 3 levels for each variable, a full factorial space search requires 3^k experimental runs. We employ three different DoE techniques to explore the desired solution space with a far less number of runs: Box-Behnken (BB), orthogonal arrays (OA), and Sobol. For each technique, we use five input variables that represent Rx PHY parameters, including CTLE (two), VGA (one), and CDR (two) settings, and then we retrieve the eye measurements from the system under test. The samples taken are later used as the training and testing data required for surrogate modeling.

2.2.1 Box Behnken (BB)

Response surface methodology (RSM) is a collection of techniques to find first and second order effects of *k* variables on the measured outputs. First order effects are easily obtained through two-level full or fractional designs, whereas second order effects are usually captured by spherical designs such as the central composite design [Viveros-Wacher-16]. that requires up to five levels for each variable (the center points, ±1 and $\pm\alpha$, where $\alpha = k^{1/2}$).

BB is a type of second order RSM design that combines factorial designs with balanced incomplete blocks designs [Wu-00]. This characteristic is particularly helpful for variables that are not able to take $k^{1/2}$ values, such as digitally controlled variables, as in our system under test. In this manner, we use only 3 levels for each variable, yielding a total number of 46 experiments. We denote this DoE as BB.

2.2.2 Orthogonal Arrays (OA)

OAs are experimental designs identified by $L_N(s^k)$, where *N* is the number of experimental runs, *s* is the number of states (or levels) for each variable and *k* is the number of variables [Chang-05]. Their most important feature is that for each variable, all possible levels appear equally often. OAs help to reduce the number of experiments while maintaining the ability to measure the effect of each variable on the output without the need to test all possible combinations.

When $s = 2$, the resulting OA allows to see linear effects. By increasing the value of *s*, nonlinear effects can be assessed. We use an $L_{27}(3^5)$ OA in our work in order to capture non-linear effects in the objective function by only running 27 experiments. We denote this DoE as OA27.

2.2.3 Sobol

The most commonly used stochastic sampling algorithm is Monte Carlo. Monte Carlo sampling tends to generate clusters of points, leading to unnecessary samples, as well as leaving gaps in the solution space. One approach to overcome these issues is to use quasi-Monte Carlo methods such as low-discrepancy sequences [Cheng-00], where discrepancy is the measure of non-uniformity of a sequence of points.

We select the Sobol [Sobol-67] low-discrepancy sequence as the third DoE option to sample the solution space, which improves the exploration as the number of samples increases, at the expense of increasing test time on the real system. Therefore, we use three different Sobol DoE, denoted as Sobol50, Sobol100 and Sobol150, with 50, 100 and 150 samples, respectively.

2.3. Surrogate Modeling and Optimization

Surrogate models provide fast approximations of the system response, making optimization and sensitivity studies possible [Queipo-05]. The major benefit of surrogate models is the ability to quickly obtain any number of additional function evaluations without resorting to more expensive numerical models. In this section, several surrogate modeling techniques are explored to construct an efficient surrogate model for PHY equalizer simulation.

2.3.1 Surrogate Model Formulation

The electrical margining system response $\boldsymbol{R}_\mathrm{f} \in \Re^2$, denoted as the fine model response, consists of the eye width $e_\mathrm{w} \in \Re$ and eye height $e_\mathrm{h} \in \Re$ of the measured functional eye diagram, and depends on the Rx PHY tuning setting values $\boldsymbol{x}$, the operating conditions $\boldsymbol{\psi}$ (voltage and temperature), and the devices $\boldsymbol{\delta}$ (silicon skew and end-point devices),

$$\boldsymbol{R}_{\mathrm{f}}(\boldsymbol{x},\boldsymbol{\psi},\boldsymbol{\delta}) = \begin{bmatrix} e_{\mathrm{w}}(\boldsymbol{x},\boldsymbol{\psi},\boldsymbol{\delta}) \\ e_{\mathrm{h}}(\boldsymbol{x},\boldsymbol{\psi},\boldsymbol{\delta}) \end{bmatrix} \tag{2-1}$$

The surrogate models are trained such that its response is as close as possible to the fine model response for all data in the training set,

$$\boldsymbol{R}_{\mathrm{s}}(\boldsymbol{x},\boldsymbol{\psi},\boldsymbol{\delta}) \approx \boldsymbol{R}_{\mathrm{f}}(\boldsymbol{x},\boldsymbol{\psi},\boldsymbol{\delta}) \tag{2-2}$$

where $\boldsymbol{R}_{\mathrm{s}} \in \Re^{m}$ is the response of the surrogate model.

The training procedure requires two sets of inputs $\boldsymbol{x}$ and targets $\boldsymbol{R}_{\mathrm{f}}$, one during the learning phase, where the model aims to approximate the actual measurements, and another one for the testing phase, to measure the generalization performance of the model.

2.3.2 Surrogate Modeling Techniques

We exploit five different surrogate modeling techniques: polynomial-based surrogate modeling (PSM), support vector machines (SVM), kriging, generalized regression neural networks (GRNN), and 3-layer perceptron neural networks (3LP ANN).

In PSM, the surrogate model is implemented by exploiting the multinomial theorem, which allows the algorithm to raise a polynomial to an arbitrary power. A polynomial function is used to represent the behavior of the response around a reference design. The order of the polynomial function is increased until generalization performance deteriorates.

The SVM technique solves a constrained quadratic optimization problem, finding a global optimum for the model parameters. The optimization problem is feasible due to the use of kernel functions, being the radial basis function the most employed kernel [Angiulli-07] and [Xia-06].

Kriging surrogate modeling is based on space filling experiments, aiming at covering the whole experimental area [VanBeers-05].

GRNN is a special type of ANN that does not require an iterative training procedure [Mahouti-14]. Moreover, the number of neurons in the hidden layers is equal to the number of learning samples [Specht-91]. As the number of samples becomes large, this technique exhibits a fast learning and convergence to the optimal regression surface [Panda-14].

The 3LP ANN is the most widely used feedforward network [Rayas-Sánchez-04]. The number of neurons in the hidden layer (h) depends on the required complexity of the ANN, and its final number is defined based on the ANN generalization performance. In this work, the 3LP ANN

TABLE 2.1. SURROGATE MODELS GENERALIZATION ERROR ε FOR EYE HEIGHT

model	BB	OAL27	Sobol50	Sobol100	Sobol150
PSM	2.77%	8.90%	2.68%	2.05%	0.42%
SVM	6.35%	6.70%	6.69%	6.79%	6.77%
Kriging	3.10%	7.01%	2.74%	1.89%	1.45%
GRNN	7.47%	9.27%	2.86%	2.15%	1.58%
3LPANN	3.33%	7.14%	2.49%	1.96%	1.15%

TABLE 2.2. SURROGATE MODELS GENERALIZATION ERROR ε FOR EYE WIDTH

model	BB	OAL27	Sobol50	Sobol100	Sobol150
PSM	1.66%	2.79%	1.37%	1.23%	0.11%
SVM	3.27%	4.32%	3.43%	3.48 %	3.49%
Kriging	2.71%	5.36%	1.23%	1.28%	0.55%
GRNN	3.82%	4.33%	1.14%	1.04%	0.53%
3LPANN	2.96%	2.59%	1.71%	1.27%	0.56%

is trained by using Bayesian regularization training [MacKay-92] available in the MATLAB Neural Network ToolBox . We start training the 3LP ANN with h = 1, and we keep increasing the complexity of the ANN (the number of hidden neurons h) until the generalization deteriorates [Rayas-Sánchez-06].

2.3.3 Direct Surrogate Model Optimization

We select the best surrogate model based on the generalization performance, from all DoE and modeling techniques combinations, and we apply a direct optimization algorithm to maximize the eye diagram area using the following objective function:

$$u(\boldsymbol{x}) = -\left[e_{\mathrm{w}}(\boldsymbol{x},\boldsymbol{\psi},\boldsymbol{\delta})\right]\left[e_{\mathrm{h}}(\boldsymbol{x},\boldsymbol{\psi},\boldsymbol{\delta})\right] \tag{2-3}$$

We aim at finding the optimal set of PHY tuning knobs, $\boldsymbol{x}^*$ by solving

$$\boldsymbol{x}^* = \arg\min_{\boldsymbol{x}} u(\boldsymbol{x}) \tag{2-4}$$

The optimization procedure uses the Nelder-Mead simplex-based method [Lagarias-98] to solve (2-4). Since $u(\boldsymbol{x})$ is evaluated from the surrogate model, solving (2-4) is computationally

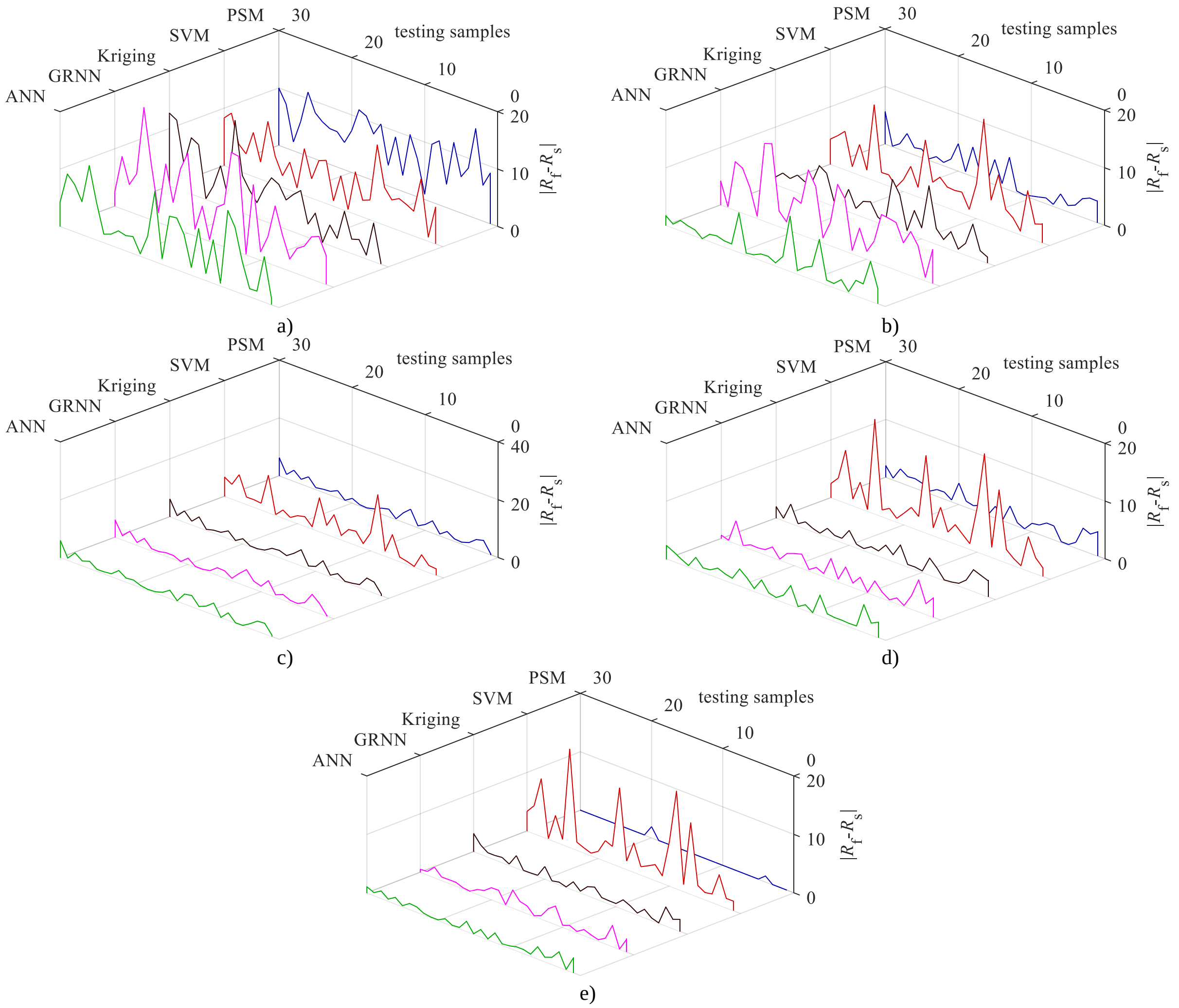


Fig. 2.3 Surrogate models absolute testing errors for eye height, using: a) OAL27, b) BB, c) Sobol50, d) Sobol100, and e) Sobol150. Image taken from [Rangel-Patiño-17b].

very efficient.

2.4. Results and Comparisons

The accuracy of the generated surrogate models is evaluated by comparing against actual SATA margins on the validation platform. The average relative error ε for eye height and eye width at testing base points ($\boldsymbol{x}_\mathrm{T}$) not seen during training is calculated as

$$\varepsilon = \frac{\left\|\boldsymbol{R}_\mathrm{f}(\boldsymbol{x}_\mathrm{T}) - \boldsymbol{R}_\mathrm{s}(\boldsymbol{x}_\mathrm{T})\right\|_2}{\left\|\boldsymbol{R}_\mathrm{f}(\boldsymbol{x}_\mathrm{T})\right\|_2} \tag{2-5}$$

The norms in (2-5) are calculated with 30 randomly distributed testing base points not seen during

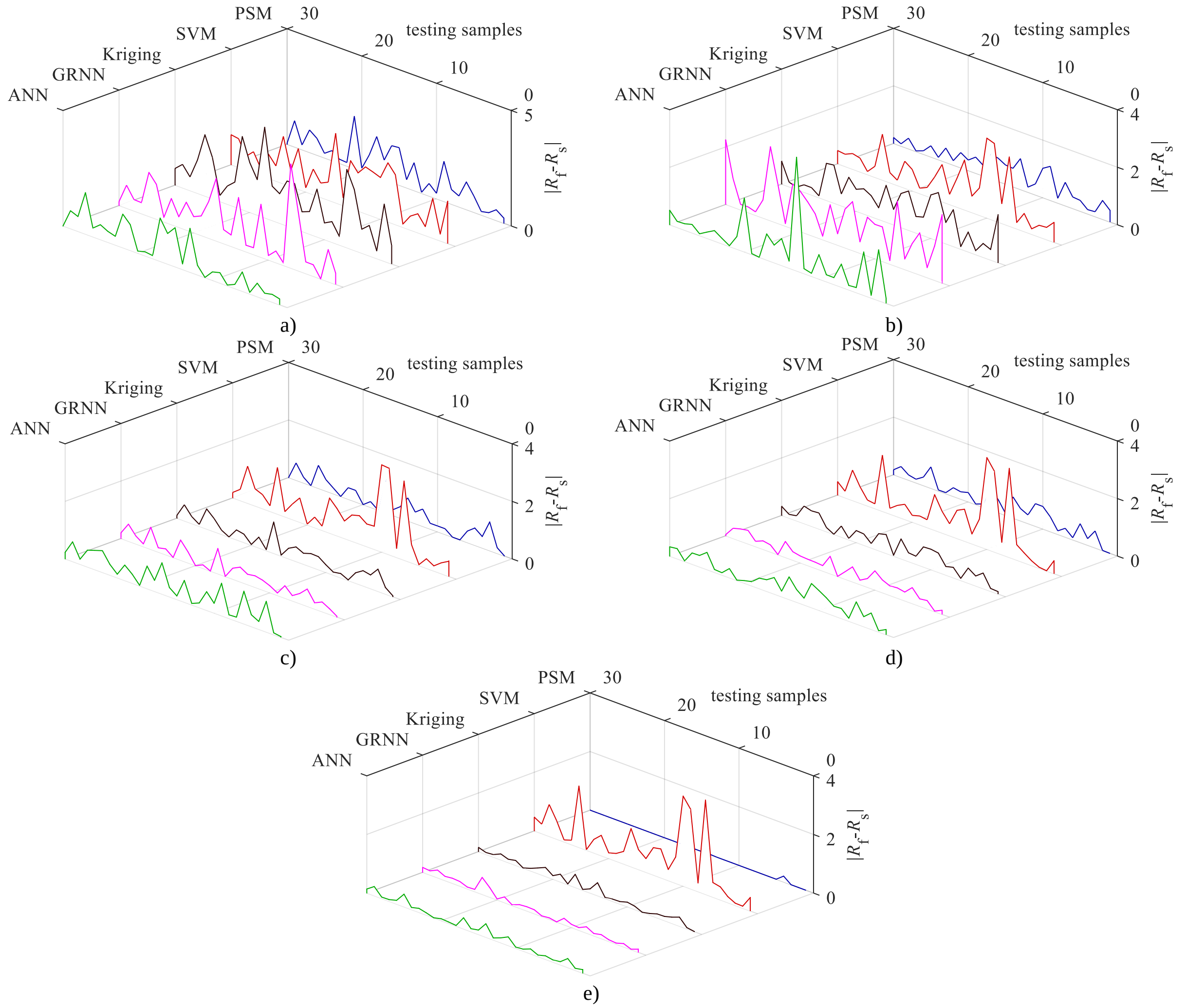


Fig. 2.4 Surrogate models absolute testing errors for eye width, using: a) OAL27, b) BB, c) Sobol50, d) Sobol100, and e) Sobol150. Image taken from [Rangel-Patiño-17b].

training.

Table 2.1 and Table 2.2 show a summary of the generalization performance, obtained from (2-5), for the eye height and eye width, comparing the five surrogate models using the five DoE: a) OAL27, b) BB, c) Sobol50, d) Sobol100, and e) Sobol150. It is seen from those tables that, overall, the PSM technique yields the lowest testing average relative errors for both eye measurements when using Sobol150, which is the DoE technique yielding best generalization performance.

Fig. 2.3 and Fig. 2.4 show the absolute error at the 30 testing samples for eye height and eye width, respectively, for the five surrogate models using the five DoE techniques. Both figures show that the accuracy of the models improve as the number of samples in the DoE technique

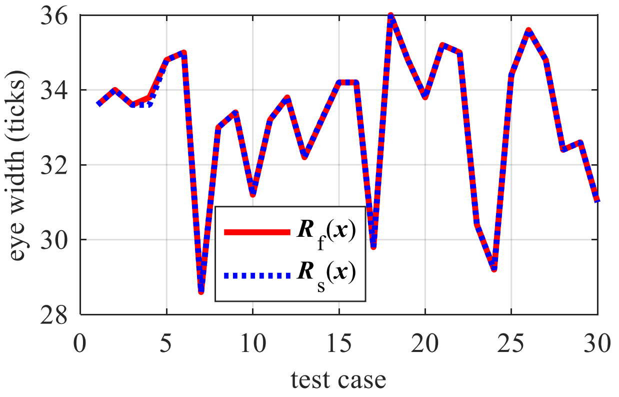


Fig. 2.5 Comparison between fine model responses and polynomial surrogate model responses at testing base points for the eye width. Image taken from [Rangel-Patiño-17b].

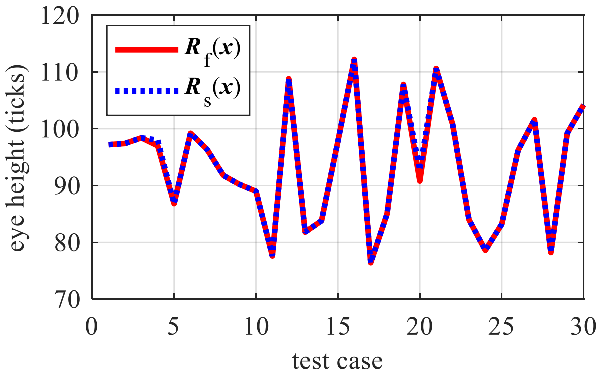


Fig. 2.6 Comparison between fine model responses and polynomial surrogate model responses at testing base points for the eye height. Image taken from [Rangel-Patiño-17b].

increase. Using OAL27, the best performance is obtained with SVM for eye height (Fig. 2.3a), and with 3LP-ANN for eye width (Fig. 2.4a). When using the BB DoE, the PSM shows the best accuracy for both eye height and width (Fig. 2.3b and Fig. 2.4b). Neural networks (both 3LP-ANN and GRNN models) exhibit the best performance when using Sobol50 DoE (Fig. 2.3c and Fig. 2.4c).

When the surrogate models are developed using Sobol100 DoE, the best performance is achieved by Kriging and GRNN models (Fig. 2.3d and Fig. 2.4d). Finally, it is observed that the PSM technique with Sobol150 DoE yields the best generalization performance (Fig. 2.3e and Fig. 2.4e), with the lowest average relative testing errors for all DoE-model combinations, as confirmed in Table 2.1 and Table 2.2. We therefore select PSM with Sobol150 as the best surrogate model found. The eye width and eye height responses from this model are compared against the actual system measurements on the 30 testing points in Fig. 2.6 and Fig. 2.6, where it is shown that the

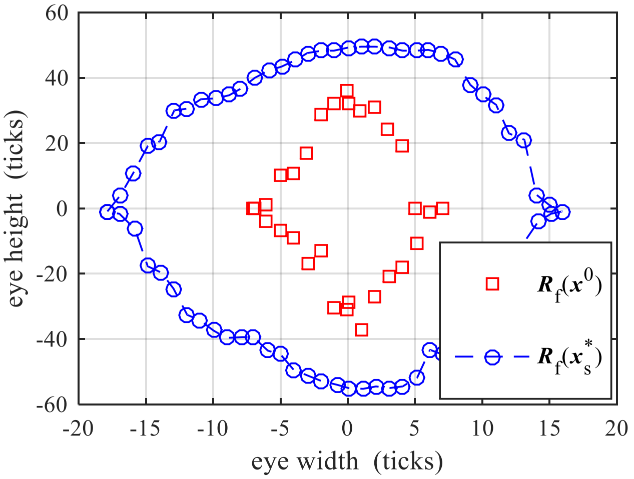


Fig. 2.7 Comparison between the system fine model responses before and after surrogate-based optimization (square and circle marks, respectively). Image taken from [Rangel-Patiño-17b].

surrogate model closely approximates the fine model responses.

The PSM model generated with the Sobol150 DoE is used to obtain the optimal Rx PHY tuning knobs by performing a SBO, as described in Section 2.3.3. The results, shown in Fig. 2.7, indicate an improvement of 400% on eye diagram area as compared to the initial PHY tuning settings, demonstrating the high effectiveness of our approach.

2.5. Machine Learning in Post-Silicon Validation

Machine learning algorithms, a branch of artificial intelligence, build statistical models from examples, which are then used to make predictions when faced with cases not seen before. On the other hand, the goal of HSIO post-silicon validation is to understand and validate from physical examples the correct operation of the design, identify bugs, and determine the best settings to avoid any failure. Machine learning aims at a similar goal: learning from examples and identifying the structure in a system [DeOrio-13]. In addition, the large volume of data generated from typical post-silicon testing suggests the application of machine learning techniques to predict post-silicon behavior.

There has been recent research on machine learning applications to some areas of post-silicon validation. In [Rahmani-17], authors propose a trace signal simulation-based selection technique that exploits machine learning to efficiently identify a small set of key traceable signals, reducing the simulation cost. An algorithm that applies anomaly detection techniques is proposed in [DeOrio-13] for post-silicon bug diagnosis. A Bayesian model fusion is described in [Wang-16] to efficiently reuse the data from pre-silicon and reduce the data collection for tuning during post-silicon. Machine learning is applied in [Pridhiviraj-15] to bug finding in post-silicon server power management. In [Deyati-14], several neural models are developed to learn post-silicon unknown module-level behavior and diagnose localized design bugs.

It is seen that all the previously cited machine learning approaches to post-silicon validation have been focused on developing efficient and reliable techniques for diagnosis, failure detection, or bug identification. An assessment of several surrogate modeling and DoE techniques to identify the best approach for a HSIO link model and simulation is realized in [Rangel-Patiño-17b]. From that assessment, polynomial-based surrogate modeling (PSM) combined with Sobol DoE with 150 samples was identified as the most accurate surrogate model [Rangel-Patiño-17b]. While an accurate model is desirable for direct optimization, it can be still expensive since it requires a significant amount of lab measurements to develop. Here, we propose a neural modeling approach to efficiently approximate the effects of a HSIO post-silicon receiver equalizer with a very reduced set of testing and training data. The resultant metamodel, obtained from the proposed inexpensive method, could later be used as a fast coarse model in a space mapping approach [Bandler-04] and [Rayas-Sánchez-16] to find the optimal equalizer settings that maximize the actual HSIO performance.

2.6. ANN-Based Receiver Metamodeling

Metamodels are scalable parameterized mathematical models that emulate the component behavior over a user-defined design space. These techniques allow developing an approximation of a system response within a design region of interest, following a “black-box” approach. The problem of modeling in post-silicon validation can be mapped to a mathematical problem of function estimation in presence of noisy data points. The most popular estimators are neural networks and Kernel estimation. In [Goulermas-07], authors demonstrate the functional estimation capability of an artificial neural network (ANN).

ANNs are particularly suitable to approximate high-dimensional and highly nonlinear relationships, in contrast to more conventional methods such as numerical curve-fitting, empirical or analytical modeling, or response surface approximations [Vicario-16]. ANNs have been used in many areas of applications, including RF and microwave circuits [Zhang-00], EM-based design optimization [Rayas-Sánchez-04], control process, telecommunications, biomedical, remote sensing, pattern recognition, and manufacturing, just to mention a few [Haykin-99]. Recently, ANNs have been used for HSIO simulations, but they were focused to model the nonlinear relationships between channel parameters and system performance to speed up system simulations, as in [Bistola-15] and [Liu-15]. In [Goay-17], authors proposed ANNs for eye diagram modeling based on simulations, and they use an adaptive sampling method for data collection process.

Once trained, ANN provides a fast way to perform a large number of I/O links and channel simulations that take into account the die-to-die process variations, board impedances, channel losses, add-in cards, end-point devices, and operating conditions [Beyene-07]. ANN modeling involves two inter-related process: a) neural network model development - that includes selection of representative training data, network topology, and training algorithms; and b) neural model validation - the neural network model is tested and validated according to its generalization performance in a given region of interest. A large amount of training data is usually needed to ensure model accuracy, and this could be very expensive in the post-silicon validation environment. An alternative to reduce the dimension of the learning set is to properly select the learning points by using DoE, to ensure adequate design space parameter coverage [Mack-07].

2.6.1 ANN Topology

Multilayer perceptrons are feedforward networks widely used as the preferred ANN topology [Rangel-Patiño-17a]. We use a 3-layer perceptron (3LP) to implement our neuromodel, with n inputs (equal to the number of Rx knobs), h hidden neurons, and m outputs (number of system responses of interest) [Rangel-Patiño-17a]. The required complexity of the ANN, determined by h, depends on the required generalization performance for a given set of training and testing data [Rayas-Sánchez-01].

2.6.2 ANN Modeling and Training

Let $\boldsymbol{R}_\text{f} \in \Re^m$ represent the actual electrical margining system response, denoted as a fine model response, which consists of the eye width $e_\text{w} \in \Re$ and eye height $e_\text{h} \in \Re$ of the measured

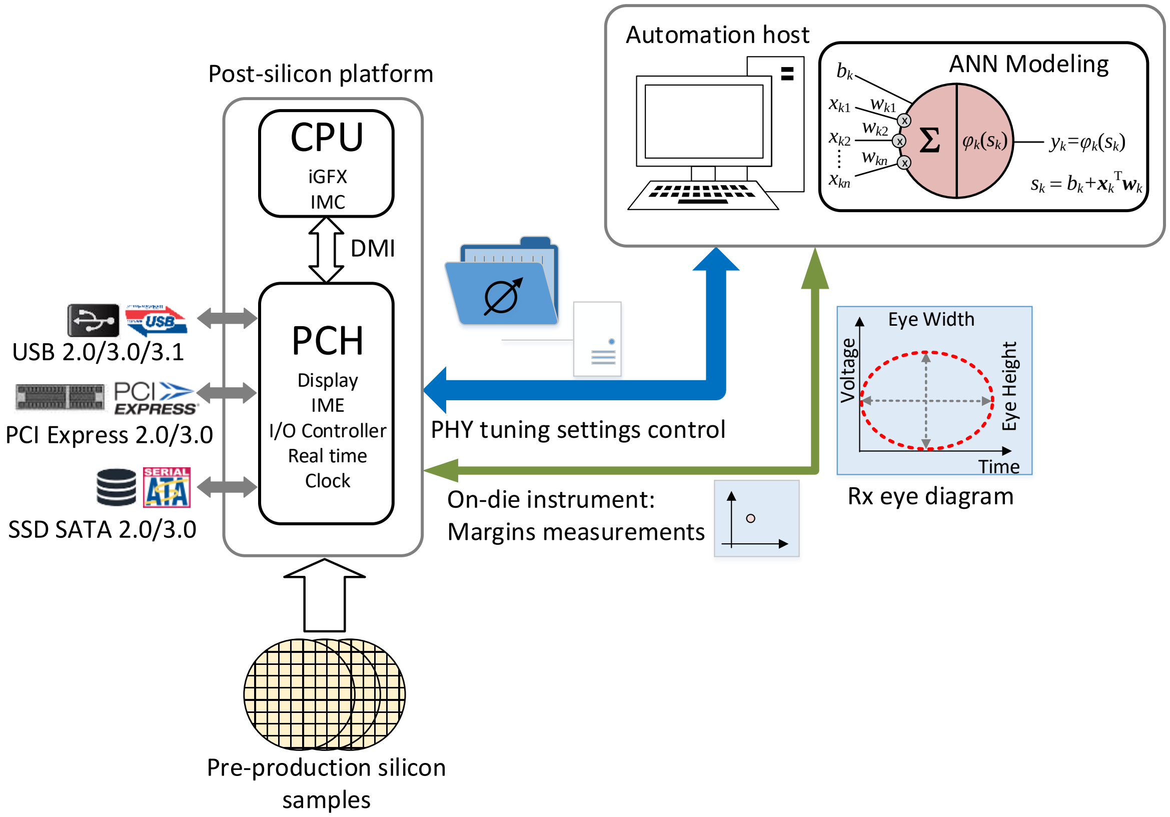


Fig. 2.8 HSIO server post-silicon hardware configuration for Rx metamodeling.

eye diagram,

$$\boldsymbol{R}_{\mathrm{f}}(\boldsymbol{x},\boldsymbol{\psi},\boldsymbol{\delta}) = \left[e_{\mathrm{w}}(\boldsymbol{x},\boldsymbol{\psi},\boldsymbol{\delta}) \quad e_{\mathrm{h}}(\boldsymbol{x},\boldsymbol{\psi},\boldsymbol{\delta})\right]^{\mathrm{T}} \tag{2-6}$$

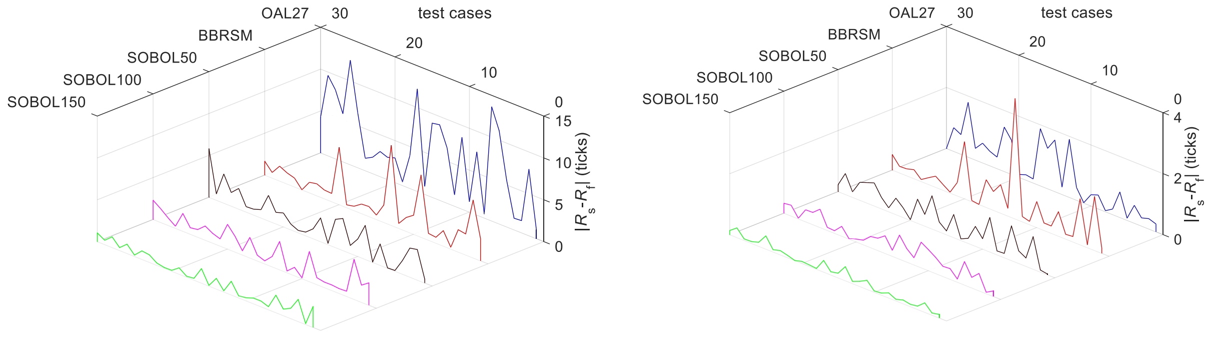
The electrical margining system response depends on the Rx knobs settings $\boldsymbol{x} \in \Re^n$, the operating conditions $\boldsymbol{\psi}$ (voltage and temperature), and the devices $\boldsymbol{\delta}$ connected to the system. The ANN is trained to find an optimal vector of weighting factors $\boldsymbol{w}$, such that the ANN response, denoted as $\boldsymbol{R}_s$, is as close as possible to the fine model response for all $\boldsymbol{x}$, $\boldsymbol{\psi}$, $\boldsymbol{\delta}$ in the region of interest,

$$\boldsymbol{R}_{\mathrm{s}}(\boldsymbol{x},\boldsymbol{\psi},\boldsymbol{\delta},\boldsymbol{w}) \approx \boldsymbol{R}_{\mathrm{f}}(\boldsymbol{x},\boldsymbol{\psi},\boldsymbol{\delta}) \tag{2-7}$$

The ANN main input-output relationship is denoted as

$$\boldsymbol{R}_{\mathrm{s}} = \boldsymbol{f}(\boldsymbol{x}) \tag{2-8}$$

We aim to develop a fast and accurate ANN model for $\boldsymbol{f}$ by training the ANN with a set of measured learning data. In [Rangel-Patiño-17a] an ANN modeling procedure was outlined, and an algorithm for training the ANN was developed. We use the same modeling procedure considering the learning data are pairs of ($\boldsymbol{x}_L$, $\boldsymbol{t}_L$), with L = 1, 2…, l, where $\boldsymbol{t}_L$ contains the desired outputs or



a) b)

Fig. 2.9 Comparison of SATA neural model generalization performance for different DoE techniques: a) eye height error; b) eye width error.

targets (obtained from measurements) for the ANN model at the $\boldsymbol{x}_L$ inputs, and l is the total number of learning samples. During training, we keep fixed the system at voltage/temperature (VT) nominal conditions and without changing the external device. Under these conditions, $\boldsymbol{\psi}$ and $\boldsymbol{\delta}$ remain constant. Therefore, the ANN model during training is treated as

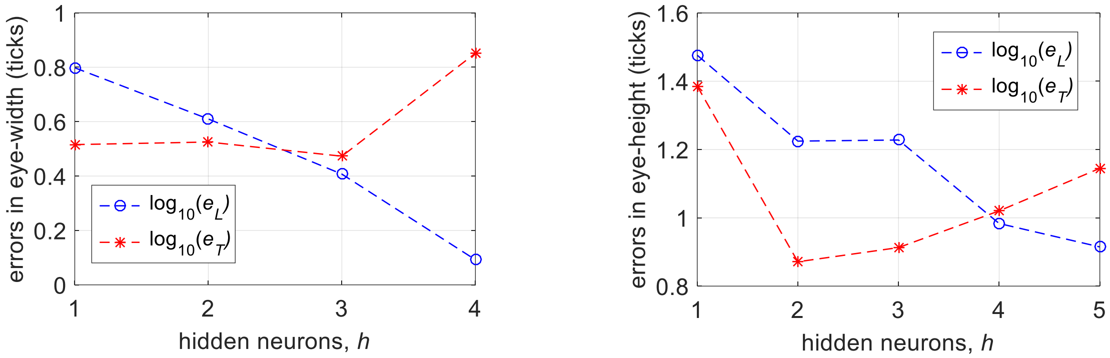
$$\boldsymbol{R}_{sL} = \boldsymbol{R}_s(\boldsymbol{x}_L, \boldsymbol{w}) \tag{2-9}$$

Following the procedure in [Rangel-Patiño-17a], the ANN model is developed.

2.7. Experimental System Configuration and DoE approaches

The system under test is a server post-silicon validation platform, comprised mainly of a CPU and a platform controller hub (PCH). The PCH is a family of Intel microchips which integrates a range of common I/O blocks required in many market segments, and these include USB, PCIe, SATA, SD/SDIO/MMC, and Gigabit Ethernet MAC, as well as general embedded interfaces such as SPI, I2C, UART, and GPIO. The PCH also provides control data paths with the Intel CPU through direct media interface (DMI), as shown in Fig. 2.8. This figure also shows the automation mechanism to read the Rx eye diagram parameters (eye width and eye height). Within the PCH, our methodology was tested on two different HSIO links: USB3 Super-speed Gen 1 and SATA3.

The measurement system is based in the system margin validation (SMV) process [Rangel-Patiño-16] and [Viveros-Wacher-14], which is a methodology to verify the signal integrity of a circuit board and assess how much margin is in the design relative to silicon characteristics and processes. The SMV methodology consists of measuring the Rx functional eye width and eye



a) b)

Fig. 2.10 Learning and testing errors during SATA neural training using Sobol50, for a) eye width and b) eye height.

height by using on-die design for test (DFT) features until the eye opening has been shrunk to a point where the Rx detects errors or the system fails [Rangel-Patiño-17b].

We employ three different DoE techniques to explore the desired solution space with a reduced number of test cases. For each test case, we use seven input variables that represent Rx knobs (n = 7), such as CTLE, VGA, and CDR settings, and then we retrieve the eye measurements from the system under test. The employed DoE techniques are: 1) Box Behnken (BB), which is a type of second order response surface methodology (RSM) that combines factorial designs with balanced incomplete blocks designs [Wu-00], using 62 experiments; 2) orthogonal arrays (OA) [Chang-05], using an L27(39) array in order to capture non-linear effects in the objective function by only running 27 experiments; and 3) Sobol [Sobol-67] low-discrepancy sequence to sample the solution space. Given the quasi-Monte Carlo sampling approach of Sobol, the solution space is better explored as the number of samples increases, at the expense of increasing test time on the real system. Therefore, we use three different Sobol DoE, denoted as Sobol50, Sobol100 and Sobol150, with 50, 100, and 150 samples, respectively.

System margining testing is very time consuming when running many test cases for PHY tuning. A single test case with 3 repetitions can take up 20 minutes, and then running a Sobol150 can take up 50 hours of testing for a single VT corner. The objective of comparing several DoEs is to find a suitable sampling strategy that provides adequate ANN model performance with the least amount of testing time.

2.8. Neural Modeling Results

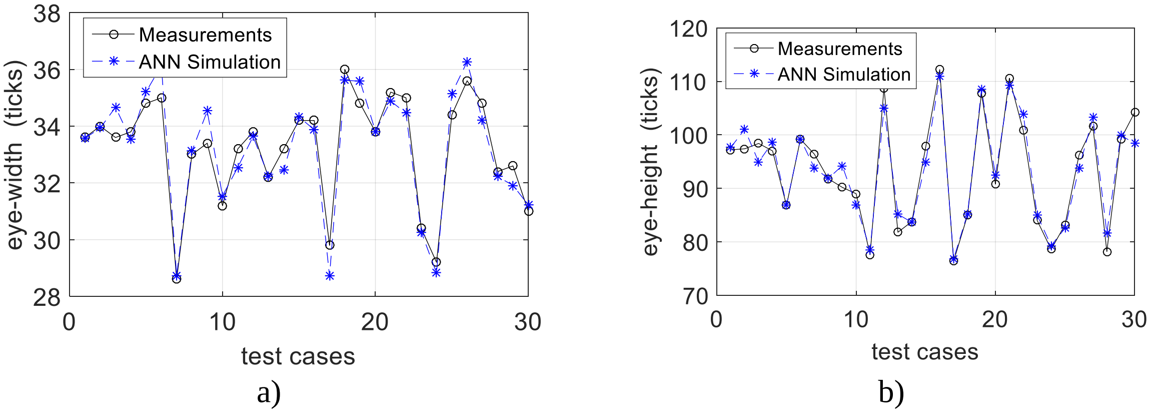


Fig. 2.11 Neural model generalization performance using Sobol50 for: a) SATA eye width; b) SATA eye height.

Fig. 2.9 shows the generalization error of the already trained neural model, comparing the different DoEs for SATA. It is seen that the best performance is achieved with Sobol150. The three Sobol cases provide the best generalization performance, as seen in Fig. 2.9. However, Sobol50 is able to achieve acceptable accuracy with only 50 samples.

Fig. 2.10 shows the learning and generalization performance of the neural training algorithm for SATA. The best performance is achieved with h = 3 for the eye width ANN, achieving a maximum relative learning error of 3.65% and 7.63% for the relative testing error. For the eye height ANN, best performance is achieved with h = 4, yielding 7.98% of learning error and 6.75% of testing error. Thus, the metamodels are able to reach above 90% of accuracy for these initial sampling points.

The already trained neural model response with h = 3 for e_w and h = 4 for eh from Sobol50 is compared in Fig. 2.11a and Fig. 2.11b, respectively, with the fine model (real measurements), by using 30 testing base points not used during training, in order to test the generalization performance. It is observed that the neural model effectively simulates the actual physical measurements with a total relative error of 1.7% for the ew response and 2.5% for the eh response. In other words, the ANN metamodel is able to predict margins with up to 95% of accuracy when using equalization values not used during the ANN training.

We obtained similar results for the case of USB3 Super-speed Gen 1. For the sake of brevity, we present only the final results in Fig. 2.12. It is seen that for USB, the resultant neural model also effectively simulates the fine model (physical platform), finding a total relative error of 6.7% for the ew response, as shown in Fig. 2.12a; and 5.7% relative error for the eh response, as shown in Fig. 2.12b. This metamodel performance was achieved using also a Sobol50 DoE.

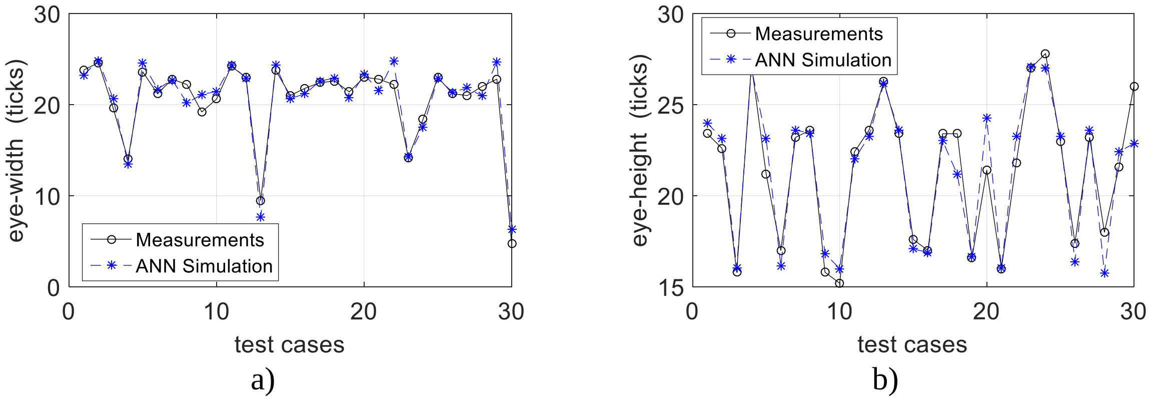


Fig. 2.12 Neural model generalization performance using Sobol50 for: a) USB eye width; and b) USB eye height.

As mentioned before, system margining testing is very time consuming when running many test cases for PHY tuning. A single test case with 3 repetitions can take up 20 minutes, and then running a Sobol150 can take up 50 hours of testing for a single VT corner. The objective of comparing several DoEs is to find the optimal sampling strategy that provides adequate ANN model performance with the least amount of testing time. The three Sobol cases provide the best error performance, as seen in Fig. 2.9. However, Sobol50 is able to achieve this with only 50 samples. Next, we improved the ANN metamodels by collecting data at three different VT corners: fast (high voltage, low temperature), slow (low voltage, high temperature), and high (high voltage, high temperature) using a Sobol 50 DoE at each corner, for ANN training purposes. A different VT corner – low (low voltage, low temperature) – and 50 testing base points not used during training were used to further test the generalization performance. Fig. 6 shows the comparison between the responses predicted by the neural model at low VT and the actual measured responses. The proposed coarse metamodel achieves a maximum error of 17.75% for eye width and 12.78% for eye height.

2.9. Broyden-Based Input Space Mapping

SM optimization methods belong to the general class of surrogate-based optimization algorithms [Booker-99]. They are specialized on the efficient optimization of computationally expensive models. The most widely used SM approach to efficient design optimization is the ASM or Broyden-based input space mapping algorithm [Rayas-Sánchez-16]. ASM efficiently finds an

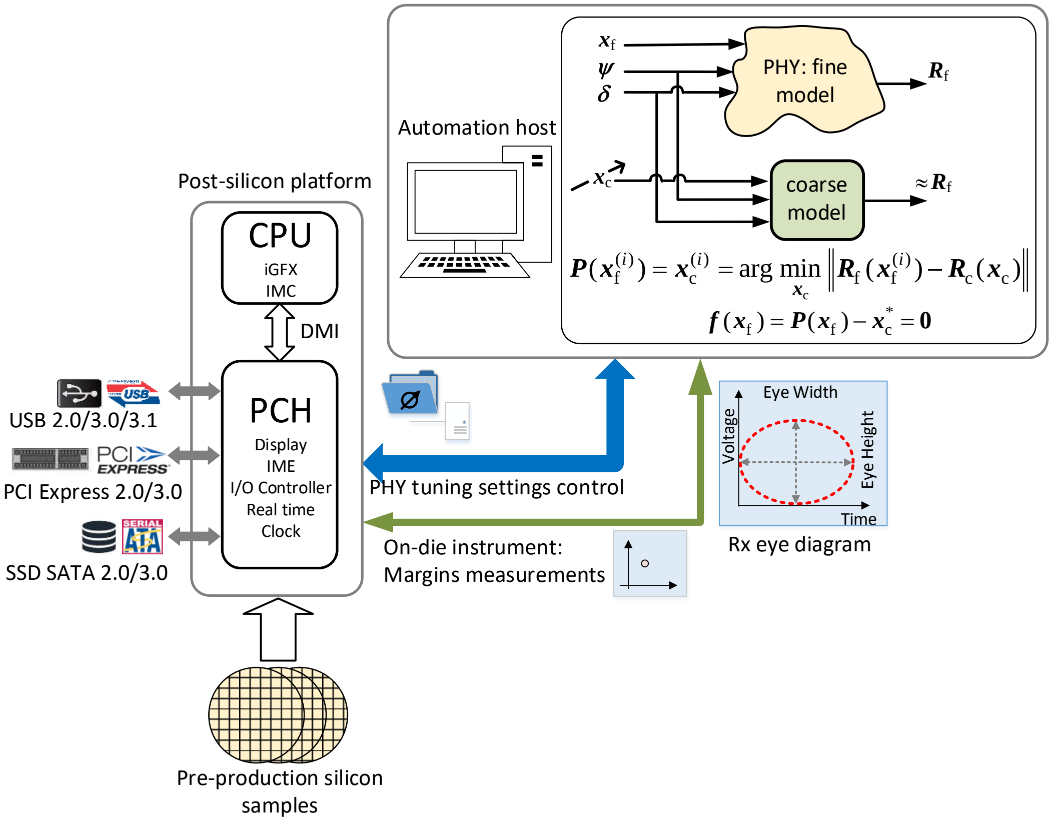


Fig. 2.13 Test setup: an Intel server post-silicon validation platform.

approximation of the optimal design of a computationally expensive model (fine model) by exploiting a fast but inaccurate surrogate representation (coarse model) [Rayas-Sánchez-16]. ASM aims at finding a solution that makes the fine-model response close enough to the desired response or target.

2.9.1 Fine Model

Our fine model is an Intel server post-silicon validation platform in an industrial environment, as shown in Fig. 2.13. The platform is comprised mainly of a CPU and a platform controller hub (PCH) [Rangel-Patiño-16]. Within the PCH, our methodology is applied to a HSIO link SATA Gen3. The SATA channel topology is comprised of the Tx driver, the Tx base board transmission lines, several via transitions, an I/O card connector, and 1 m SATA cable used to connect the base board to the device I/O card, as illustrated in Fig. 2.14. The measurement system is based on an Intel process called system margin validation (SMV) [Rangel-Patiño-16] and [Viveros-Wacher-14], which is a methodology to assess how much margin is in the design relative

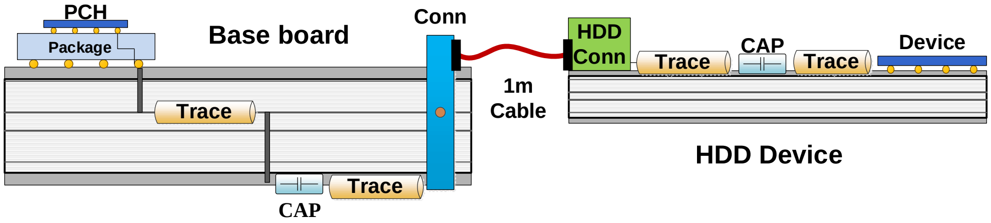


Fig. 2.14 SATA3 Rx channel topology. From [Rangel-Patiño-17b].

to silicon characteristics and processes that vary over time, including voltage, and temperature.

The fundamental process behind the SMV consists of systematically adjusting the corner conditions under which the validation platform operates, then measure the Rx functional eye opening by using on-die design for test (DFT) features until the eye opening has been shrunk to a point where the Rx detects errors or the system fails [Rangel-Patiño-17b].

Let $\boldsymbol{R}_\mathrm{f} \in \Re^m$ represent the actual (measured) electrical margining system response, denoted as a fine model response, which consists of the eye width $e_\mathrm{w} \in \Re$ and eye height $e_\mathrm{h} \in \Re$ of the measured eye diagram,

$$\boldsymbol{R}_\mathrm{f}(\boldsymbol{x},\boldsymbol{\psi},\boldsymbol{\delta}) = \left[e_\mathrm{w}(\boldsymbol{x},\boldsymbol{\psi},\boldsymbol{\delta}) \quad e_\mathrm{h}(\boldsymbol{x},\boldsymbol{\psi},\boldsymbol{\delta})\right]^\mathrm{T} \tag{2-10}$$

This electrical margining system response depends on the PHY tuning settings $\boldsymbol{x}$ (EQ coefficients), the operating conditions $\boldsymbol{\psi}$ (voltage and temperature), and the devices $\boldsymbol{\delta}$ (silicon skew and external devices). We use five input variables that represent the SATA Rx PHY tuning coefficients; these variables are settings used in three main Rx circuitry blocks (CTLE, VGA, and CDR). e_w and e_h are obtained from measured parameters,

$$e_\mathrm{w}(\boldsymbol{x},\boldsymbol{\psi},\boldsymbol{\delta}) = e_\mathrm{wr}(\boldsymbol{x},\boldsymbol{\psi},\boldsymbol{\delta}) + e_\mathrm{wl}(\boldsymbol{x},\boldsymbol{\psi},\boldsymbol{\delta}) \tag{2-11}$$

$$e_\mathrm{h}(\boldsymbol{x},\boldsymbol{\psi},\boldsymbol{\delta}) = e_\mathrm{hh}(\boldsymbol{x},\boldsymbol{\psi},\boldsymbol{\delta}) + e_\mathrm{hl}(\boldsymbol{x},\boldsymbol{\psi},\boldsymbol{\delta}) \tag{2-12}$$

where $e_\mathrm{wr} \in \Re$ and $e_\mathrm{wl} \in \Re$ are the eye width-right and eye width-left measured parameters, respectively, and $e_\mathrm{hh} \in \Re$ and $e_\mathrm{hl} \in \Re$ are the eye height-high and eye height-low parameters, respectively.

2.9.2 Coarse Model

Surrogate models can be constructed using data from high-reliability models or from measurements and provide fast approximations of the original system or component at new design points [Queipo-05]. In [Rangel-Patiño-17c], we analyze several surrogate models trained with different DoE techniques to find a good coarse model able to approximate a USB3.1 Gen1 HSIO link with a very reduced amount of measurements, selecting the best combination of surrogate modeling technique and DoE in terms of accuracy and development time. Here, we follow [Rangel-Patiño-17c] to develop a coarse surrogate model for a HSIO link SATA Gen3. By using the PHY tuning setting coefficients as inputs $\boldsymbol{x}$ and the corresponding eye height and width as outputs $\boldsymbol{R}_\mathrm{c}$, we select a Kriging surrogate modeling technique [Rangel-Patiño-20] with a Sobol [Rangel-Patiño-17c] DoE approach with only 50 samples.

2.9.3 Objective Function

We want to find the optimal set of PHY tuning settings $\boldsymbol{x}$ that maximize the functional eye diagram area. Therefore, our objective function is given by

$$u(\boldsymbol{x}) = -\left[e_\mathrm{w}(\boldsymbol{x},\boldsymbol{\psi},\boldsymbol{\delta})\right]\left[e_\mathrm{h}(\boldsymbol{x},\boldsymbol{\psi},\boldsymbol{\delta})\right] \tag{2-13}$$

During optimization, both $\boldsymbol{\psi}$ and $\boldsymbol{\delta}$ are kept fixed.

2.10. ASM Optimization

ASM starts by finding the optimal coarse model design $\boldsymbol{x}_\mathrm{c}^*$ from direct numerical optimization, that yields the optimal coarse model response, $\boldsymbol{R}_\mathrm{c}(\boldsymbol{x}_\mathrm{c}^*) = \boldsymbol{R}_\mathrm{c}^*$. ASM takes $\boldsymbol{R}_\mathrm{c}^*$ as the target response for the fine model, aiming to find a fine model design, $\boldsymbol{x}_\mathrm{f}^\mathrm{SM}$ (also known as the space-mapped solution) that makes the fine model response $\boldsymbol{R}_\mathrm{f}(\boldsymbol{x}_\mathrm{f}^\mathrm{SM})$ as close as possible to the target response $\boldsymbol{R}_\mathrm{c}^*$.

The central part of the ASM algorithm is the parameter extraction process [Rayas-Sánchez-16], which can be considered as a vector function $\boldsymbol{P}$ representing the mapping between both design parameter spaces, $\boldsymbol{x}_\mathrm{c}^{(i)} = \boldsymbol{P}(\boldsymbol{x}_\mathrm{f}^{(i)})$. If the current extracted parameters $\boldsymbol{x}_\mathrm{c}^{(i)}$ correspond approximately to $\boldsymbol{x}_\mathrm{c}^*$, then the current fine model response approximates the desired response, $\boldsymbol{R}_\mathrm{f}(\boldsymbol{x}_\mathrm{f}^{(i)}) \approx \boldsymbol{R}_\mathrm{c}^*$. To find $\boldsymbol{x}_\mathrm{f}^\mathrm{SM}$, the ASM algorithm solve a system of nonlinear equations defined as,

$$\boldsymbol{f}(\boldsymbol{x}_\mathrm{f}) = \boldsymbol{P}(\boldsymbol{x}_\mathrm{f}) - \boldsymbol{x}_\mathrm{c}^* \tag{2-14}$$

The parameter extraction process consists of finding, for the *i*-th fine model design $\boldsymbol{x}_\mathrm{f}^{(i)}$, the

```
Begin
    $x_c^* = \arg\min_{x_c} U(R_c(x_c, \psi))$
    $i = 0$, $x_f^{(i)} = x_c^*$, $B^{(i)} = I$, $\delta = 0.3$
    $f^{(i)} = P(x_f^{(i)}) - x_c^*$ using (6)
    repeat until StoppingCriteria
        solve $B^{(i)} h^{(i)} = -f^{(i)}$ for $h^{(i)}$
        $x_f^{(\text{test})} = x_f^{(i)} + h^{(i)}$
        while $x_f^{(\text{test})} \prec x_f^{\min} \vee x_f^{(\text{test})} \succ x_f^{\max}$
            $h^{(i)} = \delta h^{(i)}$
            $x_f^{(\text{test})} = x_f^{(i)} + h^{(i)}$
        end
        $x_f^{(i+1)} = x_f^{(\text{test})}$
        $f^{(i+1)} = P(x_f^{(i+1)}) - x_c^*$ using (6)
        $B^{(i+1)} = B^{(i)} + \frac{f^{(i)} h^{(i)T}}{h^{(i)T} h^{(i)}}$, $i = i+1$
end
```

Fig. 2.15 Pseudo-code for the Broyden-based input space mapping optimization. From [Rayas-Sánchez-11].

coarse model design $x_c^{(i)}$ whose corresponding response $R_c(x_c^{(i)})$ is as close as possible to $R_f(x_f^{(i)})$. This can be realized by solving

$$P(x_f^{(i)}) = x_c^{(i)} = \arg\min_{x_c} \left\| R_f(x_f^{(i)}) - R_c(x_c) \right\|_2^2 \qquad (2\text{-}15)$$

The system of equations $f(x_f)$ is directly solved by using Broyden's updating formula [Broyden-65]. Notice that solving $f(x_f)$ is equivalent to solving the mapping equation $P(x_f) = x_c^*$, which denotes that a solution to the system is found when the extracted parameters are equal to the optimal coarse model design, implying also that the fine model response is sufficiently close to the target response.

The next iterate in the algorithm is predicted by

$$x_f^{(i+1)} = x_f^{(i)} + h^{(i)} \qquad (2\text{-}16)$$

where $h^{(i)}$ solves the linear system defined as,

$$B^{(i)} h^{(i)} = -f(x_f^{(i)}) = -f^{(i)} \qquad (2\text{-}17)$$

where Broyden matrix B is an approximation of the Jacobian of f with respect to x_f at the current

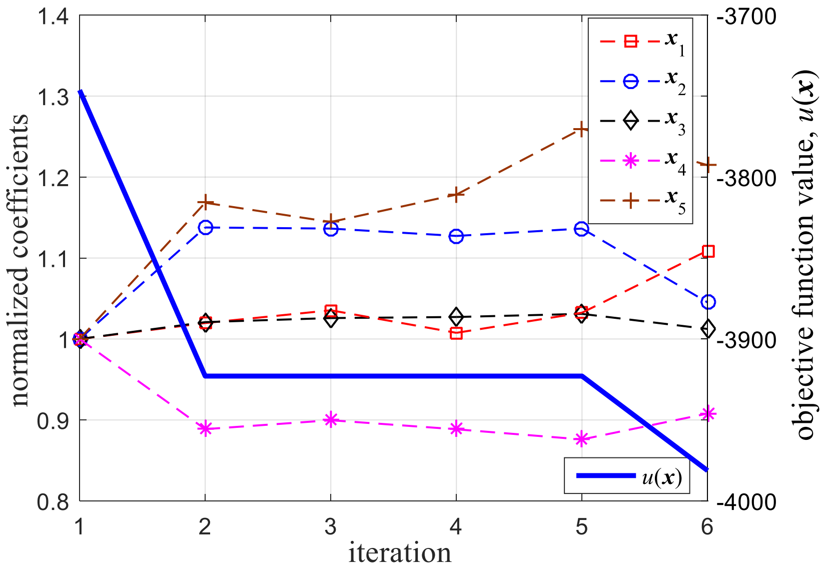


Fig. 2.16 Normalized coefficients and objective function values across SM optimization iterations.

iterate *i*. The matrix $\boldsymbol{B}$ is first initialized by the identity matrix and updated by using Broyden's formula,

$$\boldsymbol{B}^{(i+1)} = \boldsymbol{B}^{(i)} + \frac{\boldsymbol{f}^{(i)}\boldsymbol{h}^{(i)\mathrm{T}}}{\boldsymbol{h}^{(i)\mathrm{T}}\boldsymbol{h}^{(i)}} \tag{2-18}$$

The pseudo-code used to implement this algorithm is based on [Rayas-Sánchez-11] as shown in Fig. 2.15.

The stopping criteria considered in this work include four possibilities: when a root of the nonlinear system is found; when the relative change in the fine-model design parameters is small enough; when the maximum relative error in the fine-model response with respect to the target response is small enough; or when a maximum number of iterations is reached; as follows

$$\left\| \boldsymbol{f}(\boldsymbol{x}_\mathrm{f}^{(i)}) \right\|_\infty < \varepsilon_1 \vee \tag{2-19}$$

$$\left\| \boldsymbol{x}_\mathrm{f}^{(i+1)} - \boldsymbol{x}_\mathrm{f}^{(i)} \right\|_2 \le \varepsilon_2 (\varepsilon_2 + \left\| \boldsymbol{x}_\mathrm{f}^{(i)} \right\|_2) \vee \tag{2-20}$$

$$\left\| \boldsymbol{R}_\mathrm{f}(\boldsymbol{x}_\mathrm{f}^{(i)}) - \boldsymbol{R}_\mathrm{c}(\boldsymbol{x}_\mathrm{c}^{*}) \right\|_\infty \le \varepsilon_3 (\varepsilon_3 + \left\| \boldsymbol{R}_\mathrm{c}(\boldsymbol{x}_\mathrm{c}^{*}) \right\|_\infty) \vee \tag{2-21}$$

$$i > i_\mathrm{max} \tag{2-22}$$

where ε_1, ε_2, and ε_3 are arbitrary small positive scalars.

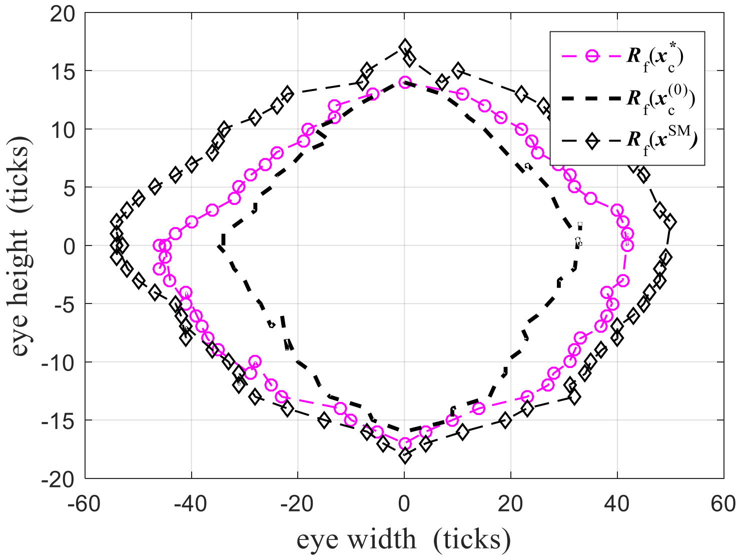


Fig. 2.17 Comparison between the system fine model responses at the initial Rx EQ coefficients, $\boldsymbol{x}_c^{(0)}$, at the optimal coarse model solution, $\boldsymbol{x}_c^*$, and at the space-mapped solution found, $\boldsymbol{x}^{SM}$.

2.11. Optimization Results

After applying the Broyden-based input space mapping algorithm [Rayas-Sánchez-16], we arrive to a space-mapped solution, $\boldsymbol{x}^{SM}$, in just 6 iterations (or fine model evaluations), as shown in Fig. 2.16. The set of Rx EQ coefficients contained in $\boldsymbol{x}^{SM}$ makes the measured SATA Rx inner eye height and width of the PCH as open as that one predicted by the optimized coarse surrogate model. The SM solution ($\boldsymbol{x}^{SM}$) found makes an improvement of 85% on the fine model eye diagram area as compared to that one with the initial settings ($\boldsymbol{x}_c^{(0)}$), and a 33% improvement as compared to that one with the optimal coarse model solution ($\boldsymbol{x}_c^*$), as shown in Fig. 2.17.

The efficiency of this approach is also demonstrated by a very significant time reduction in post-Si validation and PHY tuning Rx equalization. While the traditional industrial process requires days for a complete empirical optimization (based on engineering expertise), the method proposed here can be completed in a few hours. The technique can easily be applied to other interfaces such as USB and PCI express.

2.12. Conclusions

In this chapter, the analysis of several surrogate modeling methods were presented with different DoE techniques to approximate the response of a SATA HSIO link in a validation platform when subject to a variety of PHY tuning knobs combinations. All surrogate models were evaluated by comparing with actual measured responses. We selected the best combination of surrogate modeling technique and DoE in terms of accuracy and generalization performance and maximized the eye diagram area through SBO. The values obtained through the proposed SBO procedure were evaluated by measuring the real functional eye diagram of the physical system, showing a great improvement as compared with the initial margining system performance.

This chapter also presented a metamodeling technique based on artificial neural networks to efficiently simulate the effects of the receiver equalization circuitry in industrial HSIO links. The neural model is trained using different DoE approaches to identify the best system response sampling strategy that yields an acceptable neural model with a very reduced set of learning and testing samples. The resultant neural model approximates with sufficiently accuracy the eye diagram of a real post-silicon HSIO validation platform. The proposed machine learning approach can be exploited to develop extremely efficient vehicles to drive fast PHY tuning in HSIO links. Through this procedure, we found an efficient surrogate model that approximates the system with a reduced set of testing and training data, suitable for a future co-Kriging or space mapping optimization for PHY tuning.

Additionally, it was also demonstrated in this chapter how the Broyden-based input SM optimization algorithm, better known as aggressive space mapping (ASM), can efficiently optimize the PHY tuning receiver equalizer settings by using a low-cost low-precision surrogate as the coarse model, and a measurement-based post-silicon validation platform as the fine model. Our experimental results, based on a real industrial validation platform, demonstrated the efficiency of our method to deliver an optimal eye diagram, showing a substantial performance improvement while significantly reducing the typical time required for the PHY tuning process.

3. Analog Faults Diagnosis Exploiting Artificial Neural Networks

In the electronics business there is no such thing as a perfect design or perfect manufacturing processes. It is therefore fairly safe to assume that all products in industry will eventually fail. However, as Crosby states [Crosby-95]: “all non-conformances are caused. Anything that is caused can be prevented.” Prevention of failures comes from understanding all sources of failures as well as how, when, and why they appear.

The cost of finding a failure exponentially increases through the different stages of the lifecycle of a product. A defect found during the design development stage has an effect orders of magnitude lower than a defect found in the field. Proper fault diagnosis methodologies along with efficient testing strategies targeted to identifying all possible faults prior to production are key means for delivering a reliable and fault tolerant product to the market.

System on chip (SoC) technologies have boosted the importance of analog and mixed-signal circuitry in electronic devices. New technologies drive the demand for ever-increasing data rates, which prompts for more analog circuit complexity. While fault modelling and fault diagnostics in digital circuitry are widely used in industry, the same cannot be said for the analog counterpart. In order to deliver competitive and reliable products, substantial progress must be made in the analog fault diagnostics and testing fields. The present chapter outlines the definition of faults, distinguishing between digital and analog ones. Basic definitions of analog fault models to be used in diagnostics and testing are also reviewed.

In general, fault diagnosis is used to monitor, locate, and identify faults. Therefore, it includes three main tasks: fault detection, to check if and when a fault occurs; fault isolation, to determine the location of the fault, and fault identification, to determine the fault characteristics.

The simplest way to increase reliability and fault tolerance of a system is to use hardware redundancy, where the system is partially or completely duplicated in such a way that the different outputs can be compared to extract a diagnosis. However, hardware redundancy is expensive. For the past 40 years, alternatives such as analytical redundancy [Willsky-76] and fault tolerant control [Stengel-91] have been explored, where different methods have been created to provide fault diagnosis without the need to use hardware redundancy. There is a vast amount of applications of

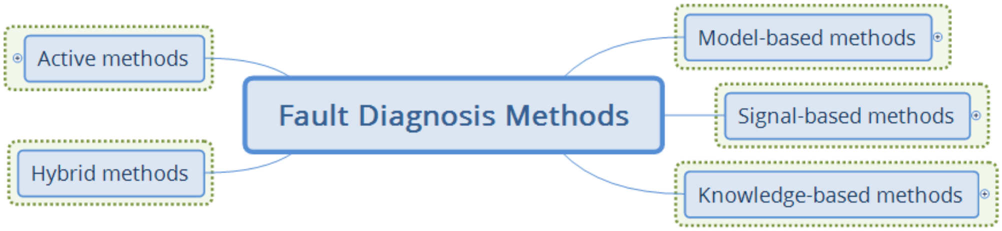


Fig. 3.1 Five main categories of fault diagnosis methods.

these methods, ranging from industrial applications, building systems, mining and machinery to electrical systems, motors and electronic circuits [Sottile-94], [Benbouzid-00], [Nandi-05], [Mortazavizadeh-14], [Katipamula-05a], [Katipamula-05b], [Widodo-07], [Feng-13], [Qin-14], [Campos-Delgado-08], [Song-13] and [Mirafzal-14].

This chapter is complemented by presenting an overview of diagnosis methods, not only for analog faults but for any fault that a system or process may present. The methods are categorized into five main branches [Gao-15a] and [Gao-15b]: a) model-based methods, b) signal-based methods, c) knowledge-based methods, d) hybrid methods, and e) active methods, as depicted in Fig. 3.1.

Additionally, this chapter presents an artificial neural network (ANN) modeling approach to efficiently emulate the injection of analog faults in RF circuits. The resulting meta-model is used for fault identification by applying an optimization-based process using a constrained parameter extraction formulation. The proposed methodology is illustrated in analog circuit examples with passive and active components. Then, a generalized neural modeling formulation to include auxiliary measurements in the circuit is proposed. This generalized formulation significantly increases the uniqueness of the faults identification process. The generalized methodology is illustrated by a faulty analog circuit: a reconfigurable bandpass microstrip filter.

This chapter revisits our work in [Viveros-Wacher-18b] and [Viveros-Wacher-19].

3.1. Definition of Faults

A fault is a malfunction in the system that affects its performance [Kabisatpathy-05]. There are many different types of classifications of faults. A fault can be a hardware defect, also known as physical flaws, or software errors [Koren-10]. In digital circuits, faults are classified as logic faults (perturbations leading to flawed logic) and timing faults (caused by delays within combinational logic that lead to an unexpected outcome) [Swingler-14]. In terms of duration, faults can present themselves as permanent, transient or intermittent: intermittent faults change between being active and inactive through time, transient faults disappear after some time and the circuit goes back to normal functionality, while permanent faults endure throughout the life of the circuit [Sorin-09]. Permanent faults are caused by three main sources: 1) physical wear-out due to mechanical stress [Dasgupta-91], thermal shocks [Blish-97], or electromigration phenomena [Barsky-04], among others; 2) defects caused by imperfect manufacturing processes or even in a perfectly manufactured chip, incorrect behavior can be caused by design defects; and 3) unendurable stress caused by improper operation.

The manner in which a fault is observed is known as a failure mode. There are three main categories of failure modes: 1) open and short circuits, 2) degraded performance, and 3) functional faults. In digital circuitry, open and short circuits lead to the so-called stuck-at faults, in which regardless of the input of a circuit, its output is stuck to a logical value, namely stuck-at-0 and stuck-at-1 [Yau-71]. Degradation faults are caused by variations of certain component parameters outside of their nominal range. Finally, in the presence of a functional fault, a circuit could still function, however its performance drops below the acceptable specified values.

3.2. The Importance of Fault Diagnosis

Integrated circuit (IC) complexity continues to increase exponentially as deep sub-micron technologies are pushed towards new horizons. This, in turn has increased the demand for efficient fault detection. One of the major cost factors in IC manufacturing is attributed to testing and fault diagnosis. Up to 70% of the total cost is related to test, according to [Koenemann-98]. Testing costs include the cost of test equipment, the cost of test development (including CAD tools and test programming) and the cost of design for test (DFT) circuitry development, among others [Bushnell-02]. In addition, cost brings other key elements to any company involved in the IC business: profitability, time to market (TTM) and beating the competition. Therefore, an optimal test strategy can provide companies a substantial competitive edge on the market. Fault diagnosis and testing techniques are valuable resources to check initial system installation and configuration and ensure correct system startup as well as to avoid masking and accumulation of errors during normal system operation.

3.3. Digital *vs* Analog Faults

Methodologies for testing digital circuits are well developed and even included in industry standards, such as Level Sensitive Scan Design (LSSD) [Eichelberger-77] and IEEE standard 1149.1 [IEEE-93]. CAD tools dedicated to generate tests and test circuitry for digital circuits have existed for several decades as well. The same cannot be said for analog circuits. The major reason for this huge difference is that test generation for digital circuits can easily be treated as a mathematical problem. This is due to the fact that in digital circuits, the difference between what does work and what does not work is crisp clear because of their intrinsic discrete signaling and timing characteristics. However, in analog circuits the question is 'how good' the circuit behaves.

Since the 1960's algorithms have been developed to create tests patterns for digital circuitry [Roth-66] and [Goel-81]. The creation of these patterns is based on calculating signal changes that might be introduced by faults. Such algorithms rest on logic rules that define input combinations to create different signatures between a faulty and a fault-free circuit. In the same manner, the algorithms calculate the way in which the faulty behavior can be propagated to the output pins. This concept is known as a fault dictionary.

In the case of combinational logic, the fault dictionary is comprised of test vectors, which define a set of inputs and the corresponding expected outputs (faulty and fault-free) [Strunz-16]. Test sequences, on the other hand, target sequential circuitry that requires not only a set of pre-defined inputs to target a fault, but a particular test vector preceded by a certain sequence of vectors.

Fault dictionaries have several limitations for their application in the analog domain. Firstly, because there are not only two choices for signal values, but in principle, an infinite number of possible values. In addition, the time characteristics of the signals is not discrete, thus time variation brings an extra dimension to the problem. A key characteristic in digital fault dictionaries is their ability to propagate the improper behavior caused by a fault to the primary outputs of the circuit under test; in the analog domain the main issue is that faulty behavior does not propagate in a single direction, it disseminates in all directions, thus even calculating the paths is far more complex than in a digital case. Furthermore, when a fault is present in a certain circuit node, it

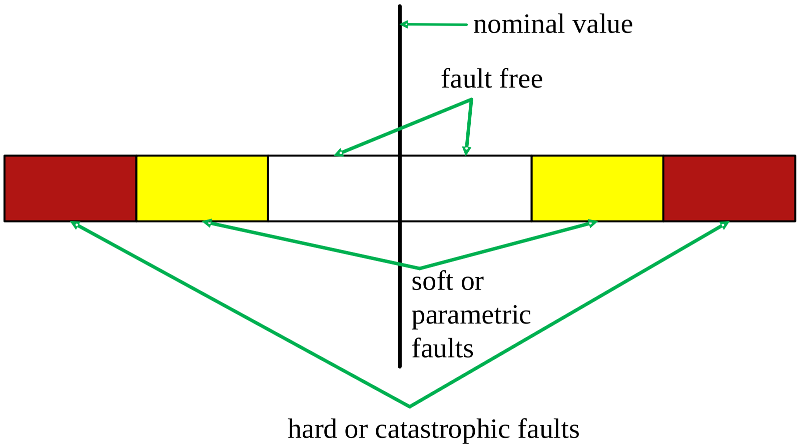


Fig. 3.2 Taxonomy of analog faults. The white area near the nominal performance parameter value shows the fault free zone. The yellow area shows soft faults and the red area shows hard faults.

does not correlate with the actual signal values of that particular node, requiring a large amount of extra calculations. In addition to the limitations already exposed, further complications arise from non-linearities, parasitic elements and energy-storing elements within the circuit, among others.

The limitations for analog fault dictionaries imply the need for an extremely large number of simulations. Moreover, the traceability issues after a fault is found render the complexity as nearly infinite [Liu-87]. Consequently, the brute force manner in which the fault dictionaries work for digital fault simulations is not applicable for the analog case.

3.4. Modeling Analog Faults

There are three main outcomes when testing an analog IC:

a) An acceptable performance, in which all specifications are within their acceptable ranges. It is said that the IC behaves correctly when facing this outcome.

b) An unacceptable performance degradation, where the circuit continues to operate, however some of its performance parameters fall outside of the adequate range; it is said that a “soft failure” is observed when the test encounters this sort of outcome.

c) A catastrophic failure, otherwise known as a “hard failure”, is seen when the circuit is completely inoperable.

Based on these kinds of outcomes, analog faults can generally be classified as catastrophic (also known as gross or hard) faults and parametric (or soft) faults. A catastrophic fault is caused

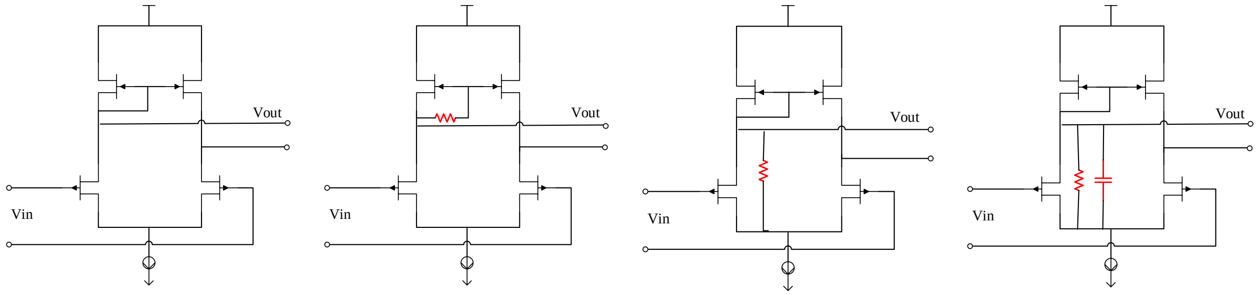


Fig. 3.3 A faultless analog circuit (left) and three models of gross faults introduced into the analog circuit.

by major structural deformities. Examples of these type of faults are open and short circuits. On the other hand, a parametric fault is caused by parameter variations outside of the tolerable range. Fig. 3.2 shows the taxonomy of analog faults based on these models. It shows a region of acceptable performance around the nominal value of a certain parameter. Then, beyond this range, parametric faults are seen and at the extremes of the parameter variations, gross faults are observed. Thus, whenever an extreme parametric fault is encountered, it can also be deemed as a hard fault.

The lack of efficient analog fault models is the major problem in analog fault diagnosis [Nagi-92]. Since both gross and parametric faults can have infinite varieties, there is an infinite number of possible analog faults, therefore, a subset has to be selected to be added into the fault list. The chosen models should be accurate enough to capture the major effects and at the same time simple enough to avoid the introduction of unnecessary components. By employing adequate models for each type of fault, a suitable diagnosis for analog ICs can be obtained.

3.4.1 Gross Fault Models

The two main types of gross faults are opens and shorts. An open fault can be modeled as a purely capacitive component inserted in series with the module under test, or as a high enough resistance at the incidence of the fault. Short faults are modeled as bridging faults using very small resistances in parallel with the component under test. Parasitic elements should be added into the fault models whenever their effect is large enough that it changes the expected outcome of the circuit under test, such as a capacitive coupling of an adjacent net. Fig. 3.3 shows examples of gross fault models introduced into an analog circuit.

3.4.2 Parametric Fault Models

As their name states, parametric faults are excursions of certain parameters outside the specified range. Examples of parameters for a CMOS transistor are its threshold voltage (V_{TH}) and channel length (L). In a typical manufacturing process, the specified range of these parameters is said to be ± 3σ, where σ is the standard deviation of the process distribution. Therefore, parametric fault models address alterations outside this range, either by shifting the mean value of the parameter or by increasing the standard deviation, effectively widening the parameter distribution. A parametric failure can occur due to a random or a systematic fault. A random parametric fault is modeled by setting the value of a parameter of each transistor (one at a time) within the circuit under test significantly further than the manufacturing range, e.g. ± 6σ. A systematic parametric fault model addresses parameter shifts of all transistors on a die in a correlated manner.

3.5. Model-Based Fault Diagnosis Methods

Model-based methods diagnose faults by monitoring the consistency between the measured outputs and the outputs predicted by the model. These models can be obtained by using physical principles or system identification techniques. Current model-based methods can be classified as deterministic methods, stochastic methods, discrete events methods, and methods for networked and distributed systems.

3.5.1 Deterministic Fault Diagnosis Methods

In deterministic systems, observers are used to detect faults through the use of residuals between the measured and the estimated outputs. A bank of residuals can be used to isolate faults using several residuals in such a way that a certain residual is sensitive to a particular fault and robust against other faults. Fault identification is performed with advanced observers (such as proportional and integral observers [Gao-08], proportional multiple integral observers [Gao-04], [Koenig-05] and [Gao-07], or sliding mode observers [Alwi-14], among others) by introducing additional system states for each fault and estimating the extended state vector. Fig. 3.4 shows the fault detection, isolation and identification of a system whose input is *u* and output is *y* and is prone

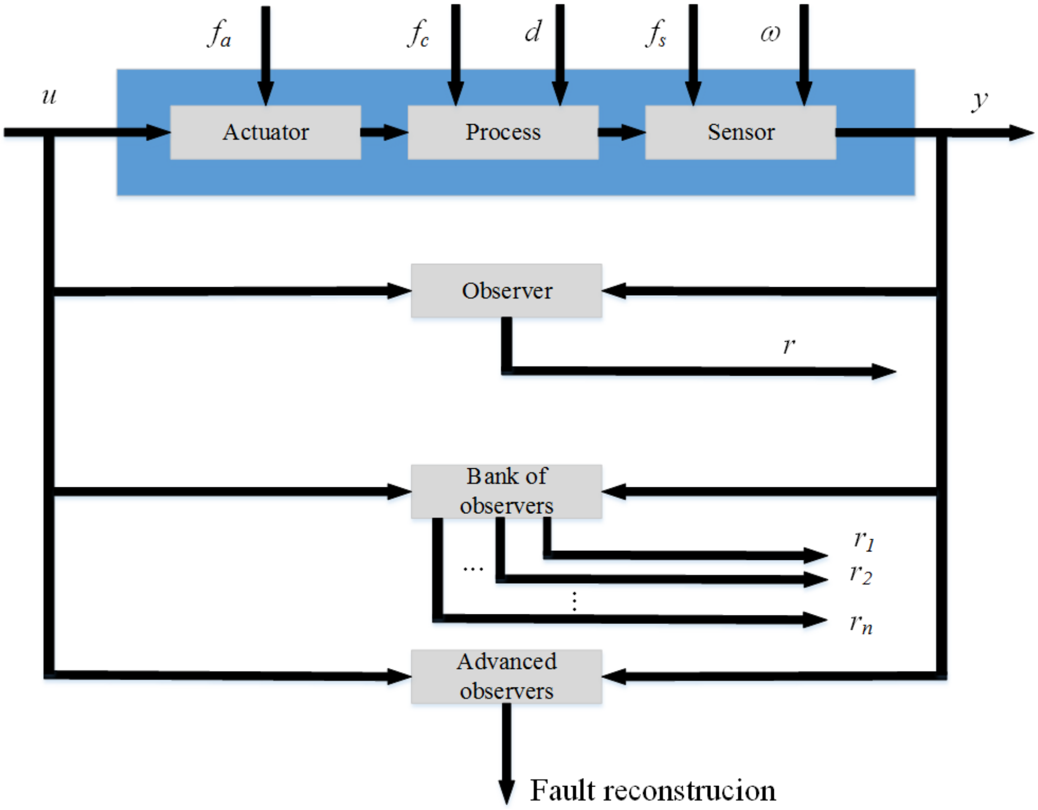


Fig. 3.4 Fault detection, isolation and identification techniques in model-based fault diagnosis methods. Figure taken from [Gao-15a].

to actuator faults (f_a), process faults (f_c) and sensor faults (f_s). The detection of a fault is feasible with the topmost observer through the residual r that must be robust against process disturbances (d) and measurement noise (ω). The bank of observers can isolate n faults using r_n residuals, while the advanced observers perform fault reconstruction.

3.5.2 Stochastic Fault Diagnosis Methods

Similarly to observers in deterministic systems, Kalman filters are used in stochastic systems [Hwang-10]. Kalman filters diagnose faults using statistical tests on whiteness, mean, and covariance of residuals. Banks of Kalman filters can be similarly used to isolate faults. Identification techniques are also used in parameter estimation methods, where certain system parameters that are prone to faults are compared to reference parameters.

3.5.3 Fault Diagnosis for Discrete Events and Hybrid Systems

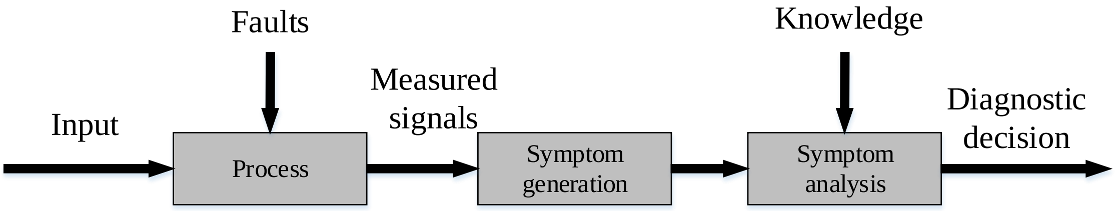


Fig. 3.5 Nature of the signal-based fault diagnosis methods. Figure taken from [Gao-15a].

Systems that change in a discrete manner rather than in a continuous one require special event-driven techniques for fault diagnosis, given that sequences of observable events need to be used to determine whether an unobservable event has occurred. For this purpose, there are two main techniques used: automata-based methods [Pencole-05] and Petri-net based methods [Cabasino-10].

There are also systems that combine continuous signals and discrete events that interact with each other. These hybrid systems can be represented with hybrid automata models [Zhao-05] and Bond-graph models [Arogeti-12].

3.5.4 Fault Diagnosis for Networked and Distributed Systems

Apart from modeling errors, process disturbances and measurement noises, networked systems are prone to problems such as communication delays, data dropout and monitoring loops. Fault diagnosis techniques for this type of systems need to be robust against all these problems. Certain types of observers and Kalman filters have been used for this type of application [He-13a] and [Rahme-13].

Distributed systems can be modeled as an interconnection of subsystems. Local estimators are used in each subsystem and a consensus strategy is used to ensure the whole estimation performance of the diagnostics network [Keliris-13], [Menon-14].

3.6. Signal-Based Fault Diagnosis Methods

Signal-based methods use actual signals from the system instead of input-output models for diagnosis. This type of methods can be used when faults manifest directly on a measurable signal that is analyzed and compared against a signal from a healthy system. Fig. 3.5 shows the

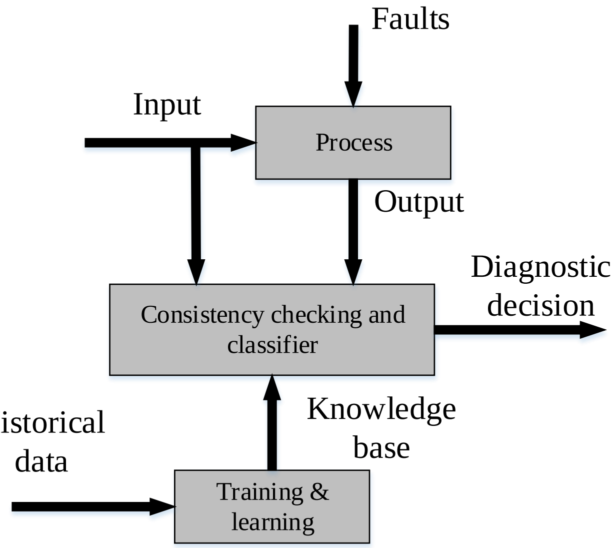


Fig. 3.6 Nature of the knowledge-based fault diagnosis methods. Figure taken from [Gao-15b].

nature of signal-based methods for fault diagnosis.

Signal-based methods are categorized by the type of signals measured: time-domain signal-based methods are used in continuous dynamical processes, where analysis can be performed based on RMS or peak magnitudes, phase, slope, mean or trends, among others; frequency-domain methods detect faults using spectrum analysis; finally, in cases where transient and steady states of the system need to be analyzed, time-frequency methods are used, extracting the required feature information of faults at any time of operation.

3.7. Knowledge-Based Methods

Knowledge-based methods require a large volume of historic data to be available, and based on this previous knowledge, artificial intelligence techniques are used for diagnosis, as shown in Fig. 3.6. The two main categories of knowledge-based methods are qualitative, and quantitative. In the former, expert-system-based methods and qualitative trend analysis (QTA) methods are found. In the latter, statistical analysis methods, non-statistical analysis methods and joint data driven methods are found.

3.7.1 Expert-system-based Methods

Expert-system-based fault diagnosis consists of generating a set of rules to detect the presence of a fault by using knowledge from past experience of human experts. In this manner, these methods are easily implemented and are able to provide diagnosis even under uncertainty. However, these methods tend to be system-specific and are therefore hard to generalize and expand.

3.7.2 Qualitative Trend Analysis

QTA is a technique that identifies trends on a process and associates them to fault trends stored in a database. This method requires either knowledge on how to identify a true change in trend vs. noise or other transient states that should not be considered as faults [Venkatasubramanian-03].

Signed directed graphs (SDG) are another qualitative fault analysis method that helps to show the causality between variables as well as to search for fault propagation paths [Xu-16]. Recent studies have also combined QTA and SDG to compensate for the disadvantages of each approach [Maurya-07] and [Gao-10].

3.7.3 Statistical-analysis-based Fault Diagnosis Methods

The main quantitative methods based on statistical analysis are principal component analysis (PCA), partial least squares (PLS), independent component analysis (ICA), and support vector machines (SVM). All of these require a large amount of training data to be able to carry out a statistical analysis.

PCA helps to reduce the dimension of the dataset into smaller features by determining the principal components of the dataset under study [Putra-16]. PLS also helps to reduce the dimensionality; this method models the relation between variables and responses by computing vectors that maximize correlation while preserving variance of the two data sets [Prates-16]. The goal of ICA, on the other hand, is to recover statistically independent components of a non-Gaussian random vector from certain observed linear mixtures of its elements [Sela-16]. Finally, SVM is a machine learning technique that relies on statistical learning theory; it can achieve high generalization even with low samples and in some cases it has proven to yield better fault diagnosis than PLS [Yin-14].

3.7.4 Non-statistical-analysis-based Fault Diagnosis Methods

Neural networks (NN) have become the most used non-statistical method for fault diagnosis [Chunlai-16], [Madani-99], [Ma-99], [Yang-08], [Talebi-07] and [He-00]. There is a large variety of NN topologies that can be used for this purpose: radial basis, recurrent dynamic, self-organizing, backpropagation and generalized regression, among others. Based on the learning strategy, NN are classified as supervised learning-based NN, in which both normal and faulty behaviors are used, and non-supervised learning-based NN, where only the normal system behavior is used during training.

Another non-statistical knowledge-based method is fuzzy logic, which partitions the feature space into fuzzy sets and uses fuzzy rules for diagnosis, approximating the way human reasoning works [Zidani-08].

3.7.5 Joint Data-driven Fault Diagnosis Methods

It is often desired to jointly use statistical and non-statistical methods to improve the diagnosis capabilities. Previous work includes the combination of PCA and NN [Ozgonenel-11], a fuzzy SVM and a self-organizing NN [Wang-12], and a Bayesian network and a recurrent NN [Cho-10].

3.8. Hybrid Fault Diagnosis Methods

Even though model-based, signal-based, and knowledge-based methods have been proven to be highly effective for fault diagnosis, each of them have some disadvantages and constraints. Model-based methods rely on the availability and accuracy of the model of the system. So, for processes where it is unfeasible or extremely difficult to derive an input-output relationship model, signal-based and knowledge-based methods are more suitable. Signal-based methods on the other hand, hardly take into account input disturbances and unbalanced conditions of the system, while knowledge-based methods depend on a high amount of prior data and are computationally expensive. Therefore, hybrid methods have appeared as a means to leverage the strength of two or more fault diagnosis methods, while decreasing the disadvantages of each individual method. Examples of hybrid methods include frequency signal-based models with statistical data-driven models [He-13b], a combination of a signal-based method, a PCA and a probabilistic NN [Seshadrinath-14], an observer-enhanced SVM [Sheibat-Othman-14], and a signal-based method combined with a knowledge-based method [Liu-17].

3.9. Active Fault Diagnosis

All the previous methods are non-invasive on the system to be diagnosed. To improve real-time detection and suppression of faults, a method known as active fault diagnosis can be used, where in specific testing intervals, the system is injected with specially designed input signals to quickly identify faults. However, as this method is invasive, the performance of the system is changed. Active methods are classified as stochastic, deterministic, and hybrid stochastic-deterministic.

3.10. Introduction to Neural Modeling of Analog Gross Faults

The growing utilization of analog and mixed signal integrated circuits (IC) has increased the demand not only of fault tolerant techniques but also of fault detection and isolation [Gao-15a]. While fault diagnosis techniques for digital circuits are mature and well established, those for analog circuits are still under development, facing significant technical challenges. This is mainly due to three key features [Bandler-85], [Kabisatpathy-05] and [Liu-91] of analog circuits not present in pure digital ones: a) there are not only two possible signal values, but in principle an infinite number of possible values; b) the timing characteristics of signals are not discrete, but continuous; and 3) the failure mode does not necessarily propagate to the output pins of the circuit.

Analog faults can be classified as catastrophic (or gross) faults and parametric (or soft) faults [Bhatta-13]. Gross faults are typically caused by structural deformities, such as open and short circuits, while parametric faults are generally caused by variations of component parameter values outside of their tolerance range. Prior work has used these two types of basic fault models and pursued a fault injection methodology to capture the circuit behavior under faulty conditions

[Bhatta-13] and [Yelten-13].

Among machine learning techniques for knowledge-based fault diagnosis, those that exploit artificial neural networks (ANN) have become the most extensively used approaches for fault diagnosis of many types of systems, including analog circuits [Chunlai-16], [Madani-99], [Ma-99], [Yang-08], [Talebi-07] and [He-00]. Machine learning-based techniques in current literature are mostly centered in trying to generate a fault dictionary to detect a predetermined k-number of faults [Haini-07] and [Ying-00], extracting features from circuital measurements, and modeling a binary-encoded set of failure modes [Yuan-06], [Zhang-16] and [Xue-11]. Most of this prior work focuses on the utilization of neural networks as classifiers, to distinguish between faulty and non-faulty responses [Chakrabarty-98], [Grzechca-02], [Yang-00] and [Rajan-98]. Other works have used the wavelet transform in pre-processing methods to improve not only the detection but also the isolation of faults [Guoming-15] and [Li-10], namely, the localization of the specific faulty circuit component. However, they require large and complex neural networks as well as significant pre-processing procedures to achieve the correct identification.

Optimization algorithms have also been extensively used in conjunction with neural networks for fault diagnosis, such as genetic algorithms [Li-12], [Liang-03] and [Li-09], particle swarm optimization [Tang-09] and [Ming-09], simulated annealing [Grzechca-11], and even hybrid methods [Haijun-12] and other novel algorithms [Binu-19]. However, most of this prior work applies those algorithms to reduce the training time, to improve the accuracy of the model, or to efficiently select the input features to model, while still employing fault dictionaries as means of identifying faults within the circuit.

This work proposes using a simple artificial neural network (classical 3-layer perceptron) to model the effects of injecting gross faults to the circuit under diagnosis. This neural model is used to learn the relationship between a faulty circuit set of responses (ANN outputs) and the origin of the failure (ANN inputs). Once the ANN is trained, it is used for fault identification and isolation through a simple yet efficient optimization process based on a constrained parameter extraction formulation, reproducing the faulty circuit responses by extracting by optimization the inputs of the already trained ANN model. We also present a generalized formulation for our fault identification neural model, by incorporating auxiliary (internal) responses of the circuit under diagnosis, in order to improve the uniqueness of the predicted fault identification.

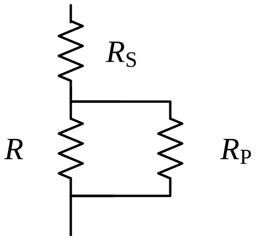


Fig. 3.7 Two possible faults in a resistor R: an open modeled with a resistance in series (R_S) or a short modeled with a resistance in parallel (R_P).

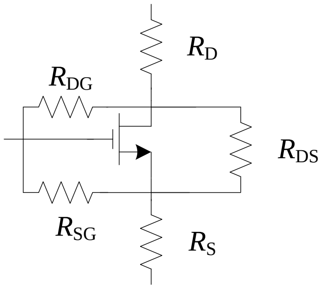


Fig. 3.8 Five possible faults in a transistor, modeled with opens on the drain (R_D) and source (R_S) terminals, and shorts between each pair of terminals (R_{DS}, R_{DG} and R_{SG}). Taken from [Viveros-Wacher-18b].

3.11. Analog Fault Models

Analog fault models aim at exposing the circuit under diagnosis to: a) a catastrophic failure, where the circuit cannot operate; b) a performance degradation, where the circuit still works but the performance is lower than its specification; and c) an acceptable performance, despite having the faults. The classification of fault models can be composed of gross fault models, which emulate open and short circuits within the main circuit topology, and parametric fault models, which emulate a variation in a circuit component outside of its nominal tolerance range.

In this work, we focus on employing gross fault models. Opens are modeled by using a high enough value of a serial resistance, while shorts are modeled by using a small enough value of a parallel resistance. Fig. 3.7 shows how fault models are employed within a resistor. A similar approach is followed when fault models are injected on any given circuit component with two terminals, such as capacitors, inductors or diodes. Fig. 3.8 shows how gross faults are injected in a transistor. An open fault is injected on each terminal (excepting the gate for the case of a CMOS transistor), while a short is injected between each pair of terminals. Given that the faults are analog,

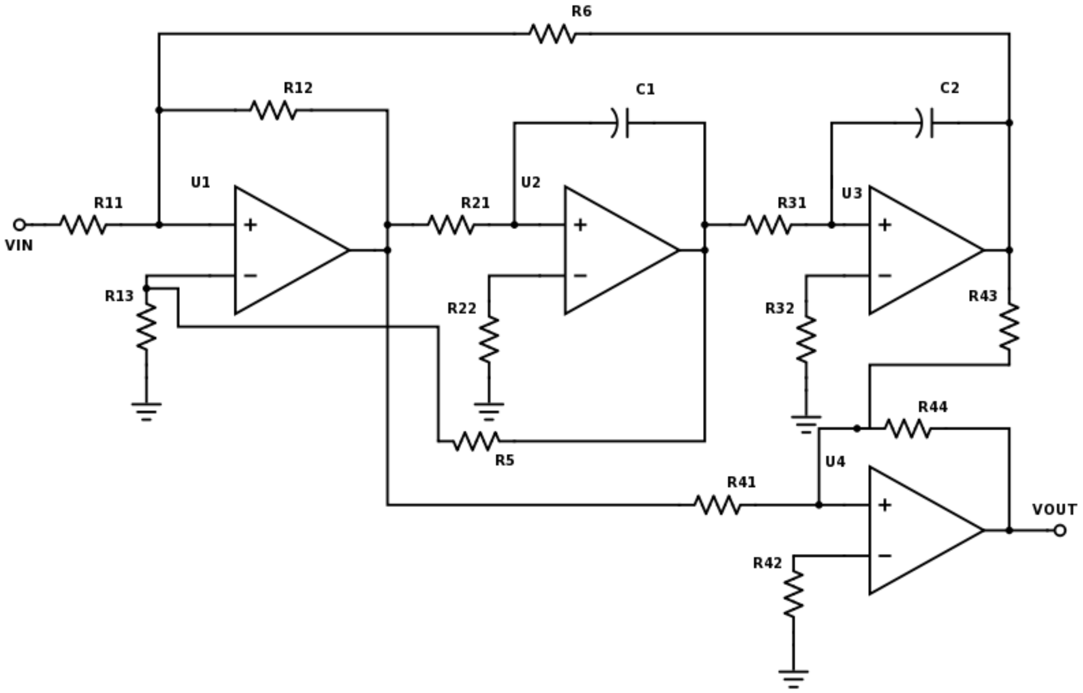


Fig. 3.9 Original state variable band rejection filter.

their values could take in theory an infinite number of possible values. However, a nominal value is chosen so that we guarantee that the desired effect is generated. In this work, the nominal values for the faults are in the order of MΩ for opens and mΩ for shorts.

We inject faults on each component of interest in a parametrized manner, in such a way that each fault can be individually activated and have a specific resistive value. When faults are not active, the value used for opens is in the order of mΩ and for shorts is in the order of MΩ. In this way, we guarantee that under no-fault conditions, the fault-injected circuit behaves as the original circuit.

3.12. Analog Faults Neural Modeling

3.12.1Neural Model Formulation

We define the vector of ANN inputs, $\boldsymbol{x}$, as follows: x_1 represents the location of the fault, or in other words, the component where the fault is injected during simulation; x_2 represents the possible fault in each component (1-2 for two-terminal components corresponding to R_S and R_P, or 1-5 for CMOS transistors corresponding to R_D, R_S, R_{DS}, R_{DG} and R_{SG}); and x_3 represents the

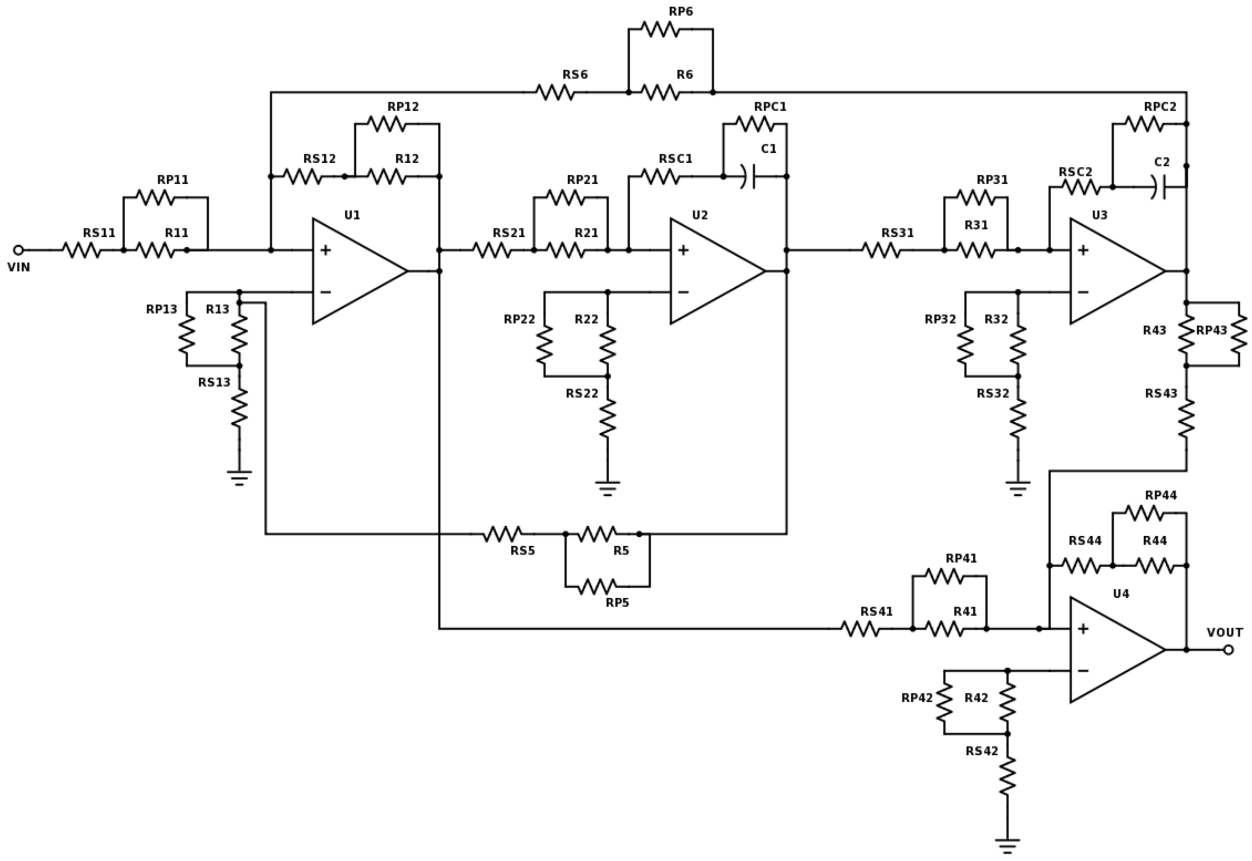


Fig. 3.10 State variable band rejection filter with gross faults injected on each passive component.

amount of deviation from the nominal fault value. In this work, we employ a reduced range from –5% to +5% for x_3, which is a reasonable manufacturing tolerance. As an initial approach, we aim to neuro-model the behavior of the circuit when injecting a single fault at a time. The output for the ANN model actually represents the deviation of the circuit responses from a no-failure condition.

3.12.2ANN Characteristics and Training

We select a 3-layer perceptron for the topology of our ANN. The ANN is implemented and trained using the Matlab[3] neural network toolbox. We select the Bayesian regularization algorithm for training, and use 1,000 base points generated using the Sobol pseudo-random sequence to sample the selected solution space as uniformly as possible [Sobol-67]. Out of the total number of

[3] MATLAB, Version 8.6.0, The MathWorks, Inc., 3 Apple Hill Drive, Natick MA 01760-2098, 2015.

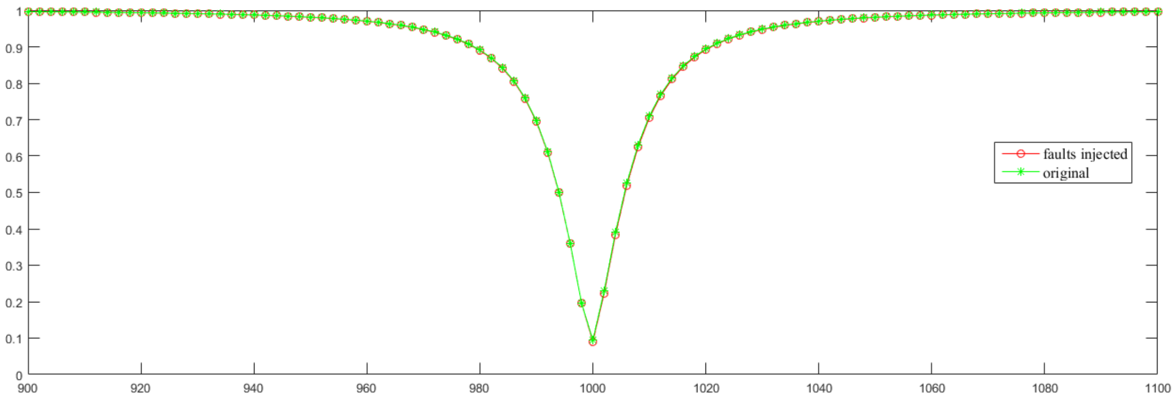


Fig. 3.11 Output comparison between the original circuit and the fault-injected circuit, when all faults are inactive.

base points, 70% are selected for learning and 30% are selected for testing. The algorithm used for training increases the number of neurons on the hidden layer, *h*, until the generalization performance deteriorates, or until the learning and testing errors are below 1%,

$$\left(e_{\mathrm{t\_old}} < e_{\mathrm{t\_new}} \wedge e_{\mathrm{t\_new}} > e_{\mathrm{l\_new}}\right) \vee \left(e_{\mathrm{t\_new}} < 1\% \wedge e_{\mathrm{l\_new}} < 1\%\right) \tag{3-1}$$

where $e_{\mathrm{t\_old}}$ is the testing error at the previous iteration and $e_{\mathrm{t\_new}}$ and $e_{\mathrm{l\_new}}$ are the testing and learning errors, respectively, at the current iteration. Each error is calculated as the Frobenius norm of the difference between the ANN output and the circuit output. Once the ANN is trained, we test it using 100 extra base points not used during training. The output from the ANN model is compared against actual deviations of the circuit simulated responses to calculate the model maximum relative error.

3.12.3Fault Identification by Parameter Extraction

Parameter extraction (PE) is an optimization problem that aims at minimizing the difference between a target response and the system response being optimized [Rayas-Sánchez-16]. In our work, we aim at finding the input values of the ANN model that minimize the difference between the objective function value of a faulty circuit, treated as the target, and the ANN output. The optimization procedure is executed by solving

$$\boldsymbol{z}^{*} = \arg \min_{\boldsymbol{z}} \left\| R(\boldsymbol{z}) - R^{t} \right\|_{1} \tag{3-2}$$

where $R(\boldsymbol{z})$ is the ANN model output and R^{t} is the target output. In our case, (3-2) is solved by using the Nelder-Mead method. In order to keep $\boldsymbol{x}$, the ANN inputs, within feasible values during

TABLE 3.1. NOMINAL VALUES FOR THE TEST CIRCUIT COMPONENTS

Component	Value
R11	19.9 kΩ
R12	20 kΩ
R13	20 kΩ
R21	12.7 kΩ
R22	20 kΩ
R31	59.1 kΩ
R32	20 kΩ
R41	10 kΩ
R42	3.3 kΩ
R43	10 kΩ
R44	10 kΩ
R5	5.1 MΩ
R6	20 kΩ
C1	5.79 nF
C2	5.79 nF
RS11, RS12, RS13, RS21, RS22, RS31, RS32, RS41, RS42, RS43, RS44, RS5, RS6, RSC1, RSC2	10 nΩ
RP11, RP12, RP13, RP21, RP22, RP31, RP32, RP41, RP42, RP43, RP44, RP5, RP6, RPC1, RPC2	1000 MΩ

the optimization iterations, we use box constraints defined as

$$\boldsymbol{x} = \begin{bmatrix} x_1^{\text{lb}} + \left(x_1^{\text{ub}} - x_1^{\text{lb}}\right)\left(\sin z_1\right)^2 \\ x_2^{\text{lb}} + \left(x_2^{\text{ub}} - x_2^{\text{lb}}\right)\left(\sin z_2\right)^2 \\ \sin z_3 \end{bmatrix} \tag{3-3}$$

where x_1^{lb} and x_1^{ub} are the selected lower and upper values, respectively, corresponding to the minimum and maximum values of x_1 when the ANN was trained. Similarly, x_2^{lb} and x_2^{ub} correspond to the upper and lower values for x_2. Notice that the optimal values of x_1 and x_2 should be both integer numbers; however, we solve (3-2) by letting the optimization process to run on continuous values for z_1 and z_2, rounding to the nearest integer the optimal final values found for x_1 and x_2.

Given the high number of possible local minima, to improve the PE performance we use a statistical PE algorithm, where the starting point of the optimization procedure is slightly perturbed

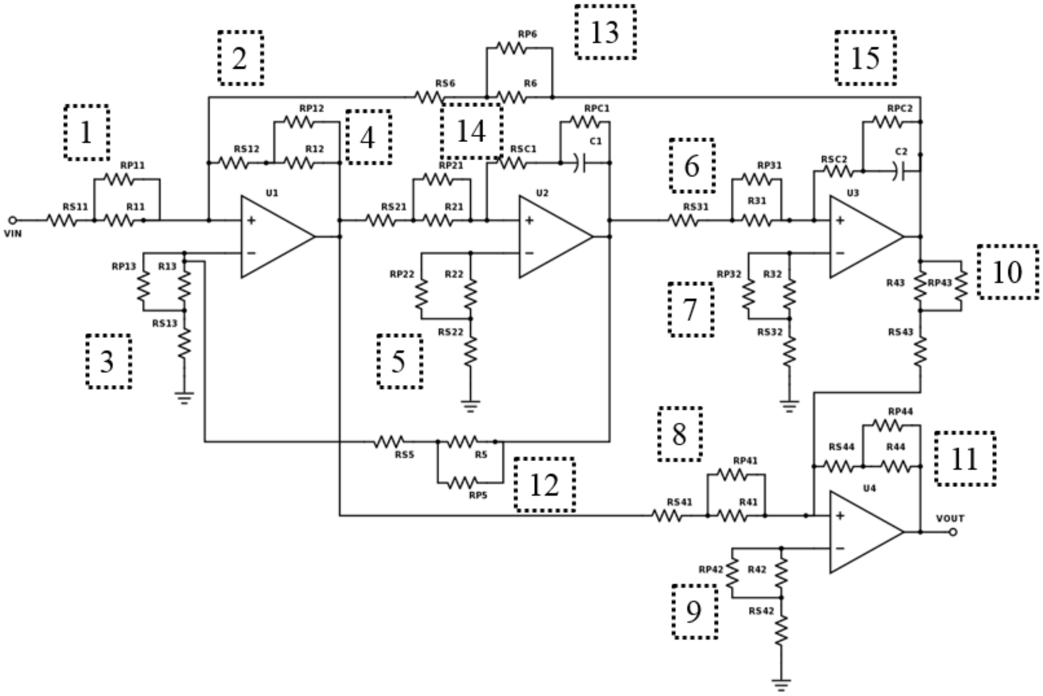


Fig. 3.12 Fault-injected circuit showing the 15 possible fault locations (shown in dotted rectangles).

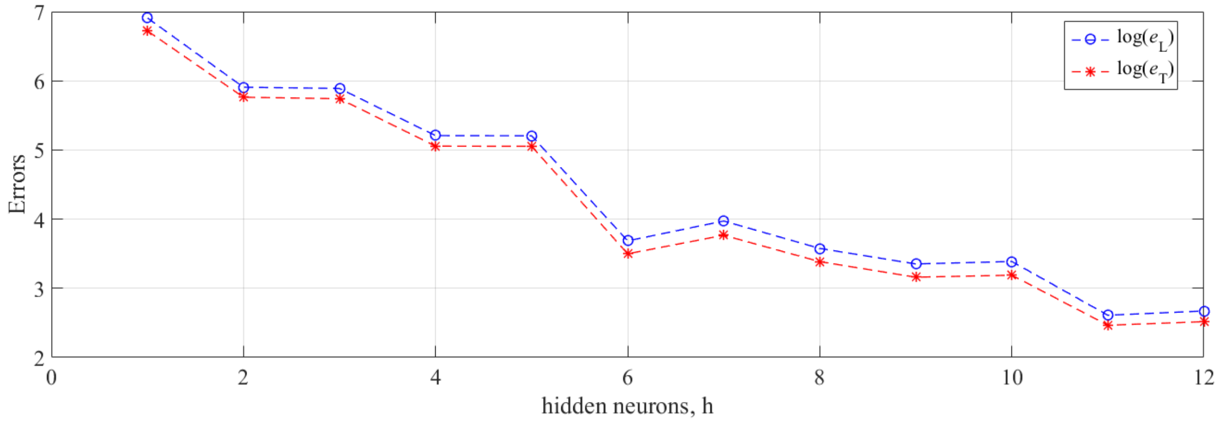


Fig. 3.13 ANN performance while increasing h, the number of neurons in the hidden layer.

each time the normalized difference between the optimal ANN response and the target response is larger than a desired value, ε_{PE}. In our case, the value selected is $\varepsilon_{PE} = 1\times10^{-5}$.

3.13. Example 1: State Variable Band Rejection Filter

3.13.1Circuit Description

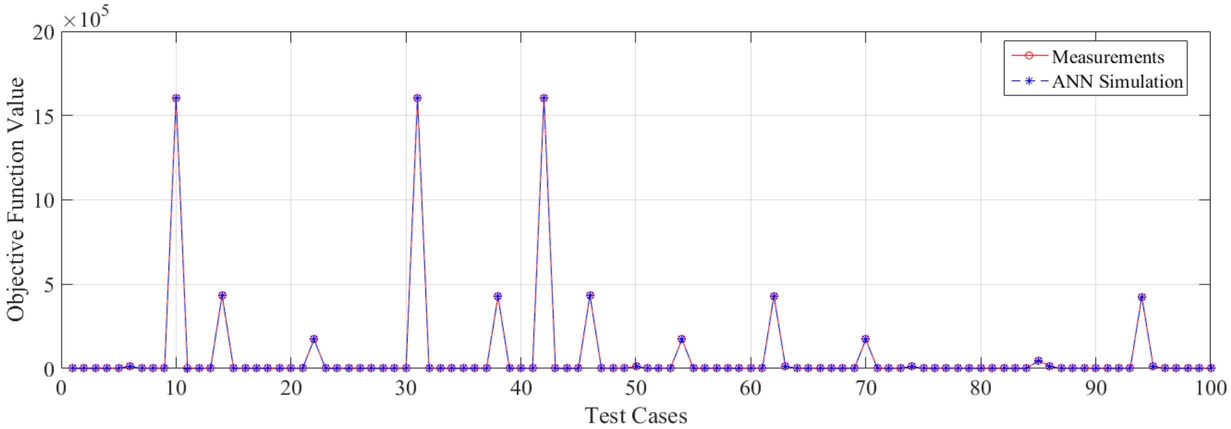


Fig. 3.14 ANN performance comparison with actual circuit simulation measurements at 100 cases not seen during training.

TABLE 3.2. VALUES FOR THE INJECTED FAULTS AND THE RESULTING IDENTIFIED FAULTS

Fault Case	Variable	Original fault	Predicted fault
Fault 1	x_1	12	12
	x_2	open	open
	x_3	3.01%	3.61%
Fault 2	x_1	3	3
	x_2	open	open
	x_3	–0.20%	0.59%

The circuit selected for fault injection is a classical state variable band rejection filter [Stanley-89] depicted in and simulated in SPICE. Under nominal conditions (without faults), the circuit is tuned at 1 MHz, as shown in Fig. 3.9.

For argument sake, we injected an open and a short fault to each passive component in the circuit in a parametrized manner, in such a way that each fault can individually be activated and have a specific resistive value. The resulting circuit is shown in Fig. 3.10. When faults are not active, the value used for opens is 10 nΩ and for shorts is 10 MΩ. In this way, we guarantee that under no-fault conditions, the fault-injected circuit behaves exactly the same as the original circuit, as shown in Fig. 3.11. The nominal values for each circuit component is found in Table 3.1.

3.13.2 Faults Neural Model and Training

We define the vector of inputs, $\boldsymbol{x}$, as follows: x_1 represents the location of the fault, from

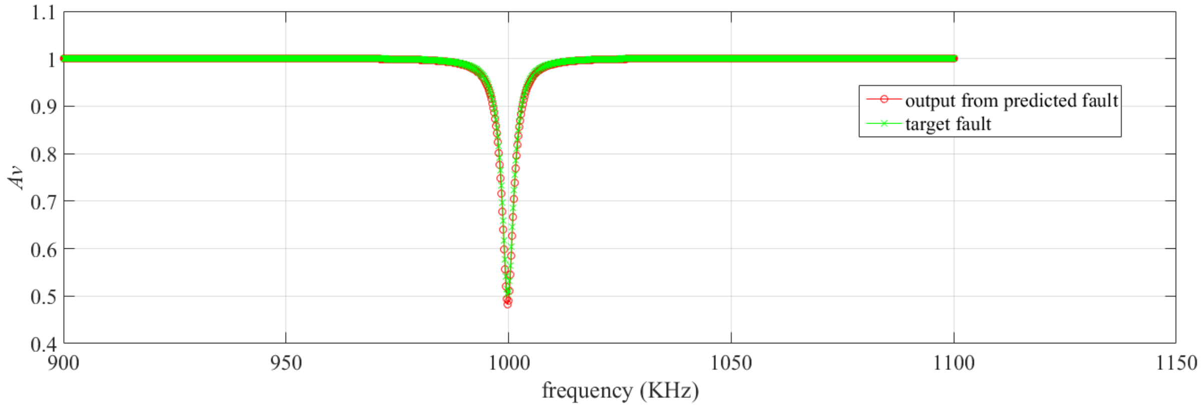


Fig. 3.15 Comparison between the circuit output at the first predicted fault and that one at the actual fault injected (used as target during parameter extraction).

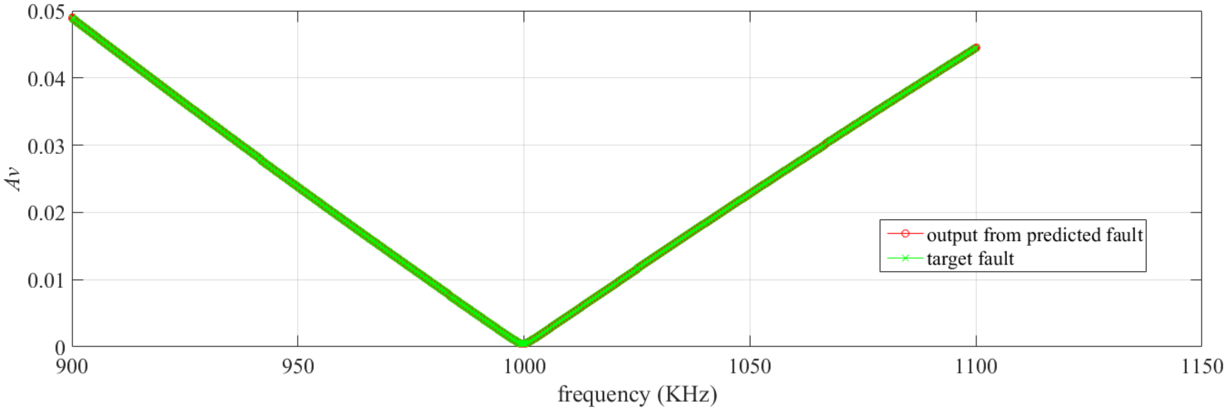


Fig. 3.16 Comparison between the circuit output at the second predicted fault and that one at the actual fault injected (used as target during parameter extraction).

one out of 15 passive components, as shown in Fig. 3.12; x_2 represents whether the selected fault is an open or a short; and x_3 represents the amount of deviation from the nominal fault value. The output selected for the ANN model is *y*, defined as

$$y = \sum_{i}^{n} \left| Av_i - Av_i^{\text{nf}} \right| \tag{3-4}$$

where *Av* is the voltage gain at the circuit output, *n* is the number of sampled frequency points, and Av^{nf} is the gain of the circuit when no faults are injected. In other words, the modeled output represents the deviation of the circuit voltage gain from a no-failure condition. The ANN performance while increasing *h* is seen in Fig. 3.13. The final value of *h* is 12.

Once the ANN is trained, we test it using 100 extra base points not used during training. The output from the ANN model is compared against actual circuit simulations. The ANN can closely predict the circuit faulty response with around 0.0103% of error, as seen in Fig. 3.14.

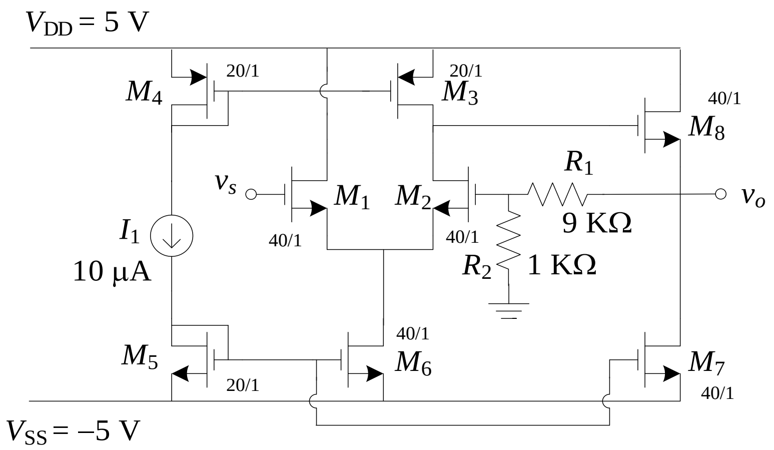


Fig. 3.17 Original negative feedback CMOS RF amplifier.

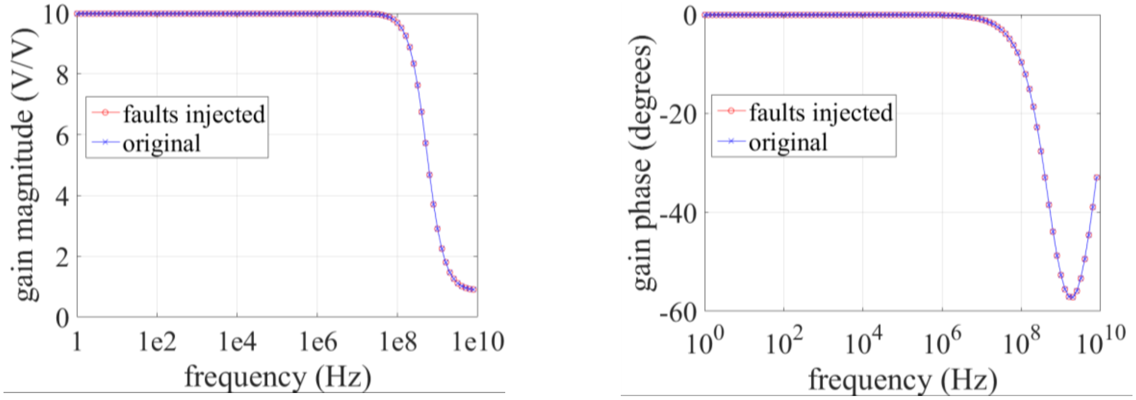


Fig. 3.18 Response (voltage gain) comparison between the original circuit and the fault-injected circuit with all faults inactive.

3.13.3Parameter Extraction to Faults Identification

To validate the effectiveness of our proposal, we selected two different random faults as targets, and followed the PE procedure. The values of $\boldsymbol{x}$ for the actual faults and the faults predicted by the PE outcome are shown in Table 3.2. The resulting values of $\boldsymbol{x}$ match exactly on x_1 and x_2, thus the fault location within the circuit and the fault type (open or short) are identified precisely on each case. There is, however, a slight variation between the predicted and the actual value in the variable corresponding to the deviation from the nominal fault value, x_3. Nevertheless, the simulated outputs from the circuit with the identified fault closely resemble the outputs with the original injected fault, as shown in Fig. 3.15 for the first fault and in Fig. 3.16 for the second one.

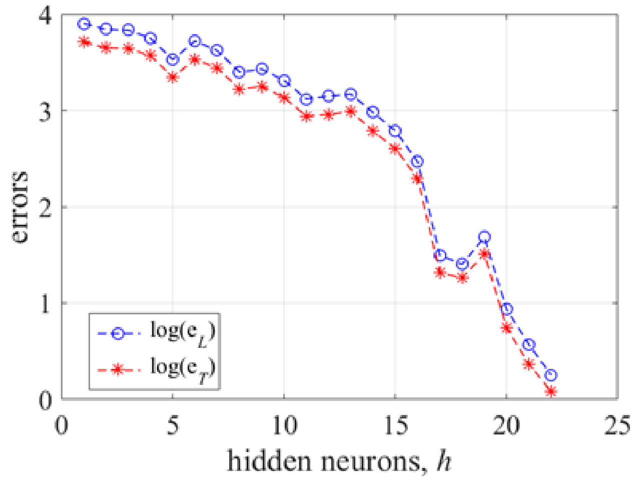


Fig. 3.19 ANN performance while increasing h, the number of neurons in the hidden layer.

3.14. Example 2: CMOS Negative Feedback RF Voltage Amplifier

3.14.1Circuit Description

The second circuit example to illustrate our fault injection and identification procedure is the classical CMOS negative feedback RF voltage amplifier depicted in Fig. 3.18, which uses an external series-parallel ideal feedback network formed by R_1 and R_2. Its nominal voltage gain is shown in Fig. 3.18. We inject an open to the drain and source pins of each transistor, and a short between each pair of transistor nodes, in a parametrized manner, in such a way that each fault can be individually activated and have a specific resistive value. When faults are not active, the value used for opens is 1 mΩ and for shorts is 200 MΩ. In this way, we guarantee that under no-fault conditions, the fault-injected circuit behaves as the original circuit, as it is confirmed in Fig. 3.18.

3.14.2Faults Neural Model and Training

We define the vector of ANN inputs, $\boldsymbol{x}$, as follows: x_1 represents the location of the fault, from one out of 8 transistors; x_2 represents one out of the 5 possible faults for each transistor (R_D, R_S, R_{DS}, R_{DG} and R_{SG}), as seen in Fig. 3.8; and x_3 represents the amount of deviation from the nominal fault value. Given that the faults are analog, their values could take in theory an infinite

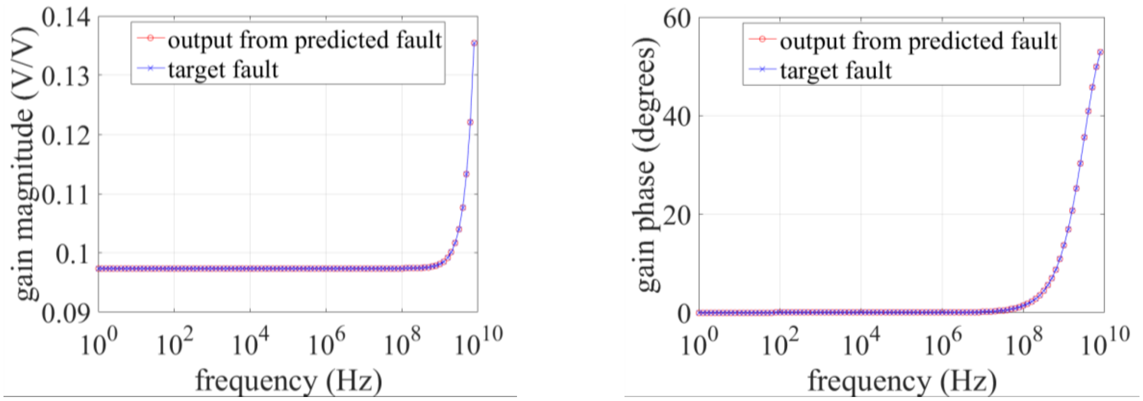


Fig. 3.20 Comparison between the circuit responses at predicted fault and those at the actual fault injected (gain magnitude and phase).

number of possible values. However, for this work we employ a reduced range from –5% to +5% for x_3, which is a reasonable manufacturing tolerance.

The output $\boldsymbol{R} \in \Re^m$ for the ANN model, is defined as

$$\boldsymbol{R} = \begin{bmatrix} \sum_{i}^{N} (\text{Re}\{Av_i\} - \text{Re}\{Av_i^{\text{nf}}\}) \\ \sum_{i}^{N} (\text{Im}\{Av_i\} - \text{Im}\{Av_i^{\text{nf}}\}) \end{bmatrix} \tag{3-5}$$

where Av^{nf} is the complex amplifier voltage gain when no faults are injected, Av is the gain with the injected fault, and N is the number of sampled frequency points. In this case, the number of responses of interest is $m = 2$, corresponding to the real and imaginary accumulative difference of the voltage gain with respect to the no-failure case in the complete frequency sweep. In other words, the neuro-modeled output represents the deviation of the circuit voltage gain from a no-failure condition. The ANN performance while increasing h is seen in Fig. 3.19. The final value of h is 21.

Once the ANN is trained, we test it using 100 extra base points not used during training. The output from the ANN model is compared against actual circuit (SPICE) simulations. The ANN can closely predict the circuit faulty response, with around 0.00635% of maximum relative error.

3.14.3Parameter Extraction to Faults Identification

We select a random fault as target, and followed the PE procedure. The values of $\boldsymbol{x}$ for the actual faults are [2 3 0.1498%]. The resulting values of $\boldsymbol{x}$ match exactly on x_1 and x_2, thus the fault

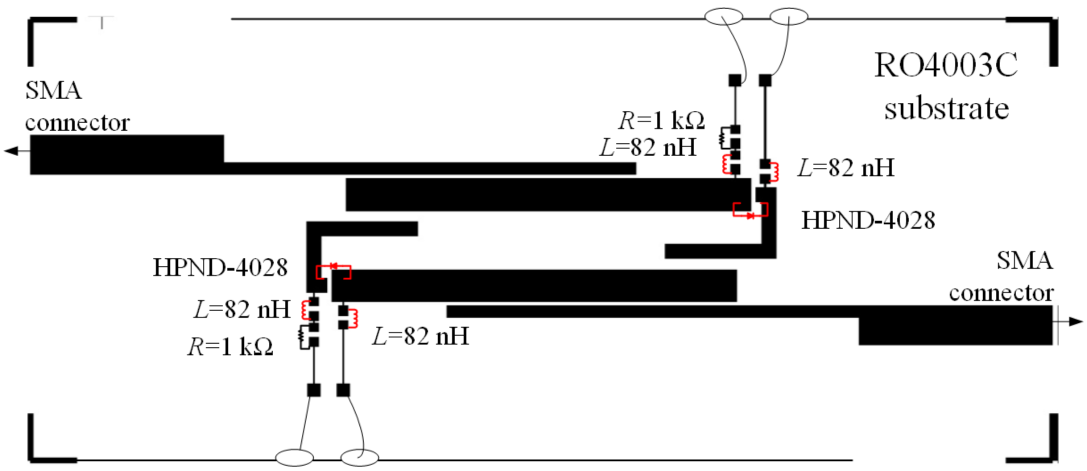


Fig. 3.21 Reconfigurable microstrip bandpass filter topology. Taken from [Brito-Brito-09]. Highlighting (in red) the components where faults are injected.

location within the circuit and the fault type (one out of five possible faults) are identified precisely on each case. There is, however, a slight variation between the predicted (0.02967%) and the actual value in the variable corresponding to the deviation from the nominal fault value, x_3. Nevertheless, the simulated outputs from the circuit with the identified fault closely resemble the outputs with the original injected fault, as shown in Fig. 3.20. This consistency was verified for other 5 cases of different injected faults, observing a similar behavior.

3.15. Generalized Neural Model Formulation

In Section 3.12, the identification of faults was achieved through observing the output responses of the circuit under test related to its main specifications (i.e., the responses of interest), and comparing them against those of a faultless scenario. However, a certain subset of faults can yield similar or even exactly the same output responses of the circuit. This directly impacts on the decision of our proposal to identify the fault, yielding to non-uniqueness issues in the extraction of the failure cause. To overcome this issue, we propose the use of auxiliary responses other than those used as specified output responses. These auxiliary responses may include internal responses of the circuit (measured at topologically internal nodes, branches, or ports), or other overall performance metrics (input impedances, cutoff frequencies, etc.). These additional simulated responses are not directly related to the main specifications of the circuit, but are extremely useful to uniquely identify the failing component in the circuit and its kind of failure. Moreover, obtaining

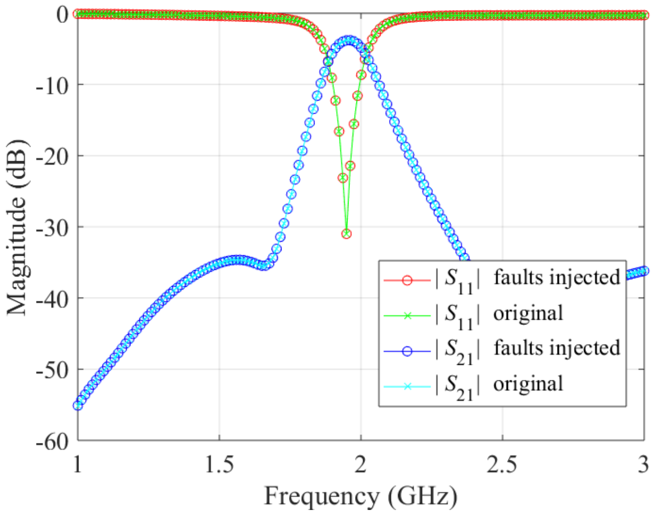


Fig. 3.22 Bandpass filter response in UMTS state. Comparing the original circuit and the fault-injected circuit with all faults inactive.

these additional responses practically do not increase the overall computational cost, since no additional simulations are implied.

We aim to neuro-model a matrix of response deviations $\boldsymbol{R} \in \Re^{m\times n}$, where m is the number of simulated responses, including the output specification-related responses, as well as the auxiliary internal simulated responses, and n is the number of components for a given response, e. g., the real and imaginary parts of the voltage gain of an amplifier, or the magnitude and phase of an S-parameter, etc. In this manner, the matrix $\boldsymbol{R}$ of response deviations to be modeled is represented by

$$\boldsymbol{R} = \begin{bmatrix} \sum_{i}^{N}\left(R_{11,i} - R_{11,i}^{\text{nf}}\right) & \cdots & \sum_{i}^{N}\left(R_{1n,i} - R_{1n,i}^{\text{nf}}\right) \\ \vdots & \ddots & \vdots \\ \sum_{i}^{N}\left(R_{m1,i} - R_{m1,i}^{\text{nf}}\right) & \cdots & \sum_{i}^{N}\left(R_{mn,i} - R_{mn,i}^{\text{nf}}\right) \end{bmatrix} \tag{3-6}$$

where N is the number of frequency points at which the response is sampled, $\boldsymbol{R}^{\text{nf}}{}_{mn}$ is the m,n^{th} response (complete frequency sweep) when no faults are injected, and the additional index i refers to the i-th simulated frequency point. We then use the same ANN characteristics, and follow the training procedure in Section 3.12.2, as well as the PE procedure from Section 3.12.3 for fault identification.

3.16. Example 3: Reconfigurable Microstrip Bandpass Filter

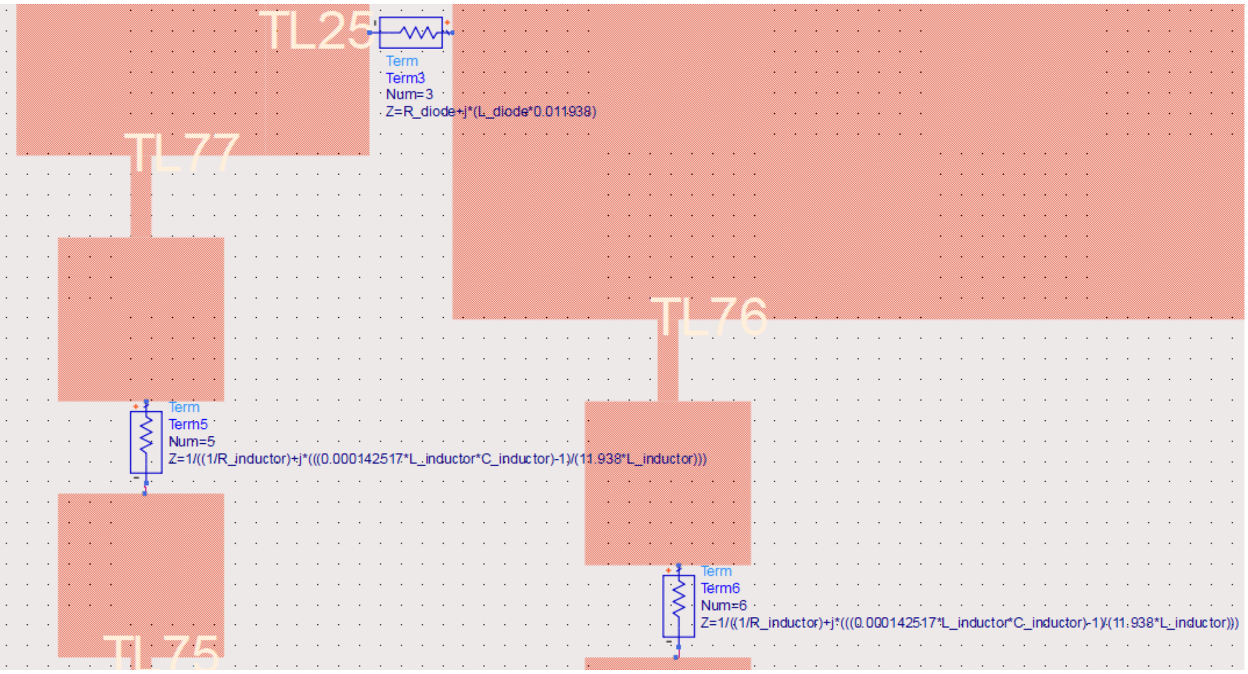

TL25
Term
Term3
Num=3
Z=R_diode+j*(L_diode*0.011938)
TL77
TL76
Term
Term5
Num=5
Z=1/((1/R_inductor)+j*(((0.000142517*L_inductor*C_inductor)-1)/(11.938*L_inductor)))
Term
Term6
Num=6
Z=1/((1/R_inductor)+j*(((0.000142517*L_inductor*C_inductor)-1)/(11.938*L_inductor)))
TL75

Fig. 3.23 Section of the circuit schematic in Fig. 3, illustrating the implementation of ports to replace a diode (top), and two inductors (bottom).

3.16.1Circuit Description

The circuit example is a reconfigurable microstrip bandpass filter able to switch between WiFi and UMTS transmit band standards [Brito-Brito-09]. When in WiFi mode, the filter center frequency is 2.44 GHz with a bandwidth of 80 MHz, and in UMTS mode, the center frequency is 1.955 GHz with a 140 MHz bandwidth. The circuit is implemented in Keysight ADS making use of co-simulation. In other words, the EM simulation was performed beforehand in Momentum and its results are stored and used in ADS. As seen in Fig. 3.21, the circuit uses two PIN diodes to switch between WiFi and UMTS states and four RF choke inductors.

We inject an open fault model and a short fault model on each inductor and diode of the circuit in its UMTS state. In this example, each fault is also activated individually in a parametrized manner. When inactive, the open fault value is 1 mΩ and the short fault value is 1 MΩ. When all faults are inactive, the circuit behaves as the original circuit, as expected (see Fig. 3.22). The responses of interest of the circuit are the return loss (S_{11}) and the insertion loss (S_{21}).

3.16.2Faults Neural Model and Training

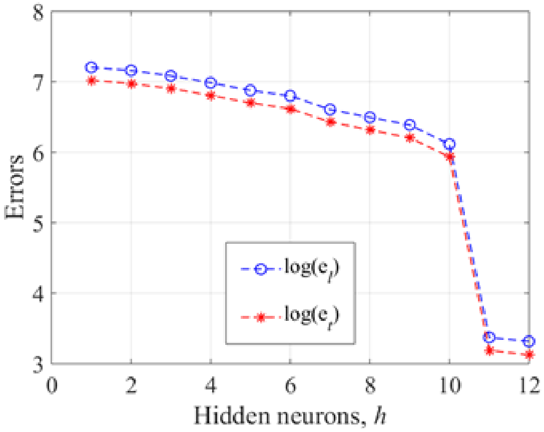


Fig. 3.24 Training the fault diagnosis neural model for the bandpass microstrip filter: ANN performance while increasing h, the number of neurons on the hidden layer.

The inputs selected for the ANN model are the fault location (x_1) from out of the 6 selected circuit elements, the fault type (x_2), meaning whether the injected fault is an open or a short, and the fault value (x_3) which relates to the deviation from the nominal value of the fault. The output for the ANN model is the deviation of the circuit responses from a no-fault condition. In this case, apart from using the return and insertion losses to calculate this deviation, we treat each fault location as a port, and use each S-parameter as auxiliary information to improve the identification uniqueness during the parameter extraction process. In order to insert a port on each fault location, the ADS schematic is modified in such a way that the actual inductors and diodes are removed, and the port is assigned a complex impedance corresponding to the impedance of the replaced circuit element, as shown in Fig. 3.23.

In this case, the selected matrix of response deviations, $\boldsymbol{R} \in \Re^{64\times 2}$, includes the magnitude and phase of the 64 S-parameters for the complete circuit (S_{11}, S_{12}, …, S_{88}), as follows:

$$\boldsymbol{R} = \begin{bmatrix} \sum_{i}^{N}\left(\left|S_{11}\right| - \left|S_{11}^{\text{nf}}\right|\right) & \sum_{i}^{N}\left(\arg\left(S_{11}\right) - \arg\left(S_{11}^{\text{nf}}\right)\right) \\ \vdots & \vdots \\ \sum_{i}^{N}\left(\left|S_{88}\right| - \left|S_{88}^{\text{nf}}\right|\right) & \sum_{i}^{N}\left(\arg\left(S_{88}\right) - \arg\left(S_{88}^{\text{nf}}\right)\right) \end{bmatrix} \tag{3-7}$$

We also select a 3-layer perceptron for our ANN, and increase the number of neurons in the hidden layer, *h*, until the generalization performance deteriorates, or until the learning and testing errors are below 0.1%. In this example, the final value of *h* is 11, as seen in Fig. 3.24. The final performance of the ANN shows a maximum relative testing error of 0.007917%.

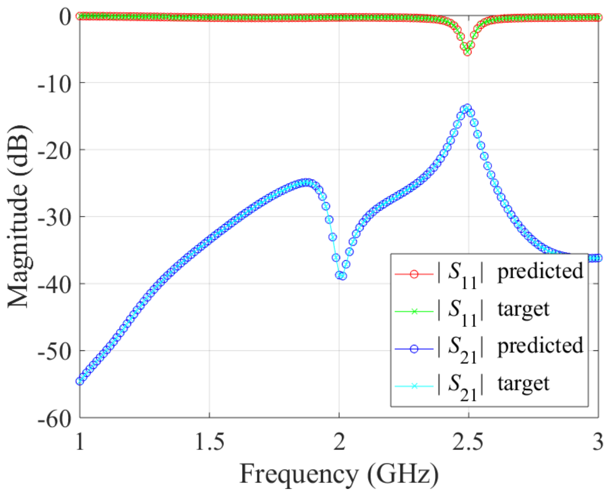


Fig. 3.25 Comparison between the circuit responses at predicted fault and those at the first actual fault injected for the RF bandpass microstrip filter.

3.16.3Parameter Extraction to Faults Identification

To validate our proposal, we select two random faults as targets (not seen during training). The values of $\boldsymbol{x}$ for the first actual fault are $[5\ \ 1\ \ 0.016231\%]^T$ and the predicted fault, following the PE process, is $[5\ \ 1\ \ 0.0373\%]^T$. As in other examples reported in [Viveros-Wacher-18b], the fault type and location are accurately identified, with a small error in the actual variable deviation. Additionally, the simulated responses of the circuit with the predicted fault closely reproduce the responses with the actual fault, as confirmed in Fig. 3.25. As an additional validation point, we use a second actual fault at $\boldsymbol{x} = [3\ \ 2\ \ -0.05445\%]^T$, for which the corresponding predicted fault, following the PE process, is $[3\ \ 2\ \ -0.02897\%]^T$. Fig. 3.26 shows the comparison between the circuit responses with the second target fault and the predicted fault. A similar performance was observed at other random faults testing points.

3.17. Conclusions

As technology continues to advance beyond the many predicted limits of Moore’s Law, and circuit and systems complexities keep increasing along the way, it is necessary to find adequate methodologies to detect failures that might occur. Analog fault modelling, diagnostics, and testing methodologies need to be developed to be able to deliver high quality SoC and mixed-signal

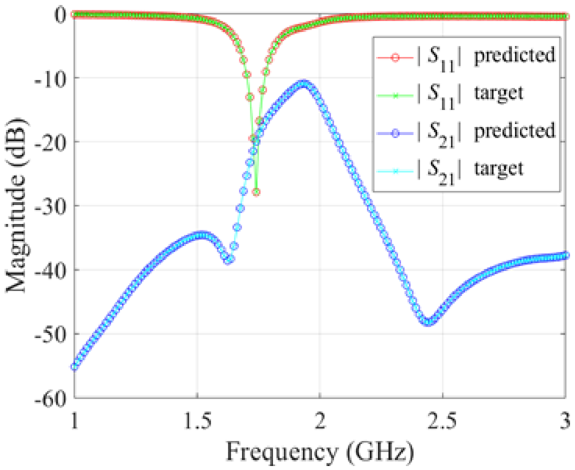


Fig. 3.26 Comparison between the circuit responses at predicted fault and those at the second actual fault injected for the RF bandpass microstrip filter.

products. This chapter presented a brief introduction to key concepts on faults and fault models, focusing on the analog type of failures.

This chapter also presented a review on the most important and most used methods for fault diagnosis. The described methods are not only applied in electronic systems, but in many different systems and processes such as chemical, nuclear, mechanical, biomedical, and electric systems, among many others. The methods presented were categorized in five main classifications, based on how faults are diagnosed, and some literature examples were provided for each method presented.

Finally, in this chapter, an analog gross fault diagnosis method based on artificial neural networks (ANN) and constrained parameter extraction was proposed. Our method was illustrated by injecting analog gross faults in two circuit examples: a state variable band rejection filter, where faults were injected in passive components, and a CMOS negative feedback RF voltage amplifier, where faults were injected in transistors. The chapter also presents a generalized formulation to increase the uniqueness of the predicted faults, by incorporating auxiliary information from internal nodes within the circuit topology. Our generalized method was illustrated by injecting analog gross faults in a circuit example: a reconfigurable bandpass microstrip filter. The gross faults were modeled as resistances with a high enough value in series to cause an open circuit and with a low enough value in parallel to cause a short circuit. The ANN was then used as a metamodel, with an extremely low computational cost, to automatically identify faults through a constrained statistical parameter extraction process. Following this process, we were able to properly identify the actual injected faults in both circuits.

4. Deep Neural Modeling of BER Extrapolation in HSIO Links

Post-silicon electrical validation of high-speed input/output (HSIO) links in high-performance computer platforms involves measuring the receiver (Rx) system margins using on-die design for test (DFT) circuitry. The measurement system allows to dwell for a certain amount of time, to check for errors in the incoming data stream. A BER can then be calculated based on the number of errors detected and the amount of bits received within the particular dwell time. In order to comply with industry standards, most HSIO links must be capable of operating at a BER of 10^{-12}, rendering volume data collection at this target BER prohibitive under current validation timelines.

Some of the current strategies involve the use of BER extrapolation techniques from volume data collection at a high BER, therefore accelerating test time, and subsequently measuring the effect of BER on margins to extrapolate the volume measurements to the target low BER. However, under high variability between measurements, these extrapolation techniques are prone to incorrect estimations.

The problem of BER extrapolation modeling is so complex that conventional meta-models, such as decision trees, or three-layer perceptron neural networks, to name a few, fail to provide the required performance (we developed numerous experiments with such conventional meta-models; they are omitted here for the sake of brevity). The high dimensionality of this modeling problem is more suitable for deep neural networks, where rather than only increasing the number of hidden neurons in a shallow network, the number of layers is also increased [Telgarsky-16]. This type of neural networks has been extensively used in numerous fields, such as image and video processing [Farabet-13], language processing [Collobert-08] and [Cho-14], and speech recognition [Chen-15], and even recent applications in microwave engineering [Jin-19] and [Ogut-19], among others.

In this chapter, an extrapolation modeling problem formulation is proposed. The chapter starts by revisiting BER definition, followed by describing BER extrapolation concepts. Next, it presents a measurement methodology based on the use of two BER values to derive a margin value extrapolated to the target BER. We aim to model the accuracy of the extrapolated margin by

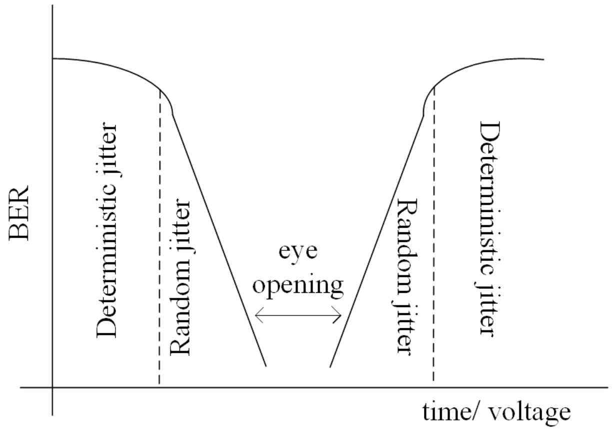


Fig. 4.1 Deterministic and random jitter effects on eye opening with respect to BER.

determining whether the obtained result is over-estimated, under-estimated or within a pre-defined error threshold. The main chapter's contribution consists of presenting a deep neural model capable of predicting the accuracy of the BER extrapolated Rx margin in HSIO links, based on statistical data from actual measurements. This chapter is based on a future publication [Viveros-Wacher-21]; for that reason, it should be temporarily treated as confidential.

4.1. BER Definitions

HSIO links standards define BER as a measure of performance. In current industrial practice, the usual expected BER is 10^{-12}, which both transmitters and receivers must comply with. Post-silicon validation tests performed at the target BER are very time-consuming, therefore, extrapolation techniques based on high BER measurements are usually employed to accelerate test time.

On the transmitter side, the concept of bathtub is usually employed [Pizano-Escalante-19]. A simple histogram-based model of the measured total jitter (TJ) distribution is composed of a bounded deterministic jitter (DJ) component, and an unbounded random jitter (RJ) component, where the latter one is modeled as a Gaussian distribution [Erb-12]. These types of models allow to identify the Gaussian model parameters and to extrapolate the histogram tails down to the desired target BER level, as seen in Fig. 4.1, where the eye opening for a specified BER level can be calculated based on the RJ parameters once the DJ boundary is passed.

On the receiver side, design-for-test (DFT) circuitry allows to measure the eye opening in terms of margin, by systematically adjusting the sampling point and checking for errors [Rangel-

Patiño-20]. In order to measure margins at the target BER, each time the sampling point is moved, the error checking engine should operate while at least 10^{12} bits are received, rendering this type of test prohibitive under current validation time constraints. Q-slope is a technique usually employed to extrapolate Rx margin measurements performed at a high BER to the target BER [Erb-09] and [Erb-10]. Q-slope is defined as the rate of change of BER with respect to the DFT step size (otherwise known as tick), where BER is measured in quantiles (Q). Formally, Q is defined as:

$$Q(\mathrm{BER}) = \sqrt{2} erf^{-1}\left[1 - \frac{1}{\rho_T}\mathrm{BER}(t)\right] \tag{4-1}$$

where erf^{-1} is the complementary error function, and ρ_T is the transition density of the data, or the ratio of the number of logic transitions to the total number of bits [Stephens-04], and t represents time-delay, or the horizontal axis on the eye diagram. Since there is no closed form solution to (4-1) for t, it is usually approximated with the normal inverse cumulative distribution function:

$$\mathrm{norminv}(t) = -\sqrt{2} erf^{-1}(2t) \tag{4-2}$$

Assuming a balanced transition density (ρ_T = ½), Q is then approximated as:

$$Q(\mathrm{BER}) = -\mathrm{norminv}(\mathrm{BER}) \tag{4-3}$$

The advantage of converting BER to a Q-scale relies in the fact that a Gaussian distribution, such as the RJ effects, behaves linearly with respect to Q, allowing to perform extrapolations to lower BER values very simply. In this way, the traditional methodology entails measuring margins at a high BER, and then perform a dedicated experiment to calculate the value of the Q-slope. Finally, the margins are extrapolated to the target BER using the Q-slope value.

4.2. Measurement Methodology in an Industrial Environment

Next-generation physical layer designs include DFT circuitry that allows to measure Rx eye height margins at a specific BER value. The DFT allows to configure error ratio parameters: N_{sym}, or the number of bits to be sampled, and an error threshold e_{th}, i.e. the amount of errors tolerated during the margin measurement. Based on the DFT architectural changes, the margin methodology is modified so that two different sets of eye height margins (EHM), named as EHM_1 and EHM_2, are now taken at two different BER levels. Using these two margin sets, the margin at

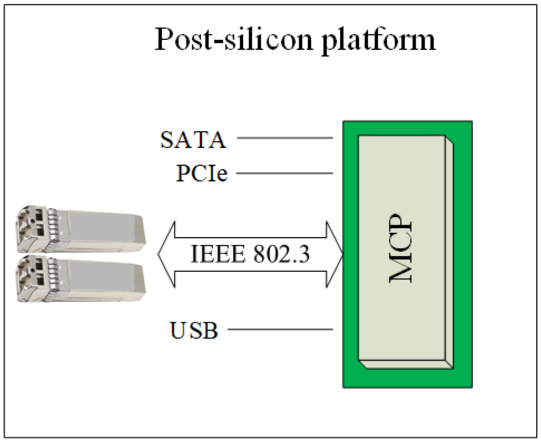


Fig. 4.2 HSIO post-silicon validation platform highlighting the multi-protocol ethernet interface.

the target BER, or EHM_{12}, is obtained by performing a linear extrapolation in the Q scale:

$$EHM_{12} = EHM_1 + \frac{EHM_1 - EHM_2}{Q_1 - Q_2}(Q_{12} - Q_1) \tag{4-4}$$

where Q_1, Q_2, and Q_{12} are calculated with (4-3), and correspond to the BER values of EHM_1, EHM_2, and EHM_{12}.

Since the EHM has inherent variations that depend on the real-time errors detected, the method to use this DFT includes a certain amount of EHM results or readouts, n, typically set at $n = 8$. This implies that for each EHM measurement at any specified BER level, a vector of 8 margin results is stored.

The BER values for both EHM measurements are selected in such a way to avoid the DJ region of a traditional bathtub curve, to guarantee that the extrapolation is performed in the linear RJ region. Any residual DJ effects found in EHM measurements can be removed by using moving average methods [Kim-06], [Hansun-17] and [Zhang-12].

The system under test is an Intel server post-silicon validation platform in an industrial environment, whose block diagram is shown in Fig. 4.2. The platform is comprised of a multichip package (MCP) that contains both the CPU and the platform control hub (PCH) dies. Within the many HSIO interfaces on the MCP, we aim to measure margins and extrapolate them in the multi-protocol ethernet interface, capable of operating at up to 25Gb/s per lane [IEEE-18], as shown in Fig. 4.2.

4.3. Synthetic Data Generation

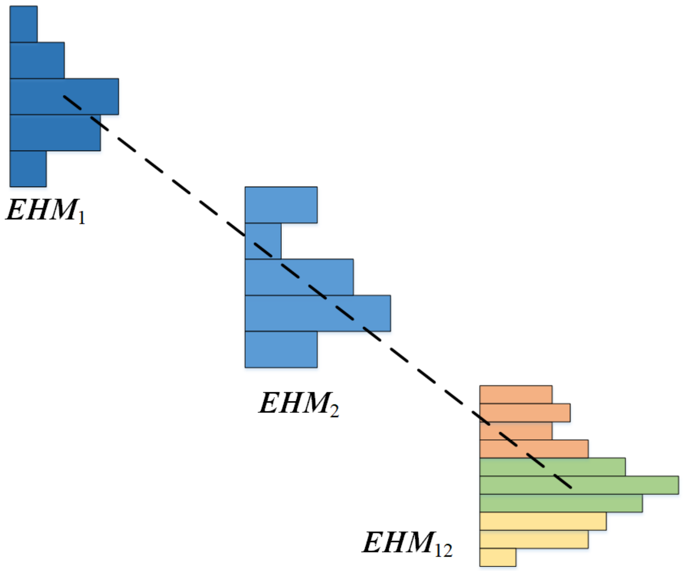


Fig. 4.3 BER extrapolation problem based on two EHM distributions (***EHM***$_1$ and ***EHM***$_2$), and the resulting ***EHM***$_{12}$ distribution showing under-estimated values (in orange), over-estimated values (in yellow) and accurate estimations (in green).

The nature of the EHM measurements described in the previous section allows to synthetically generate random sets of EHM_1 and EHM_2 values, and along with the known N_{sym} and e_{th} parameter values, the corresponding Q values can also be synthetically derived. Furthermore, from actual measurements, a covariance matrix between EHM_1 and EHM_2 can be calculated. In this way, we generate two sets of random synthetic data scaled to the variation of actual measurements taken in a post-silicon validation scenario, obtaining a vector $\boldsymbol{EHM} \in \Re^{ns}$ given by

$$\boldsymbol{EHM} = \mathrm{randn}(1, ns) \times \sqrt{K_{xx}} \tag{4-5}$$

where randn is a function, used in this case as a Matlab command, that generates a normally distributed pseudorandom number vector of size ns (n being the number of EHM readouts, and s, the amount of EHM measurements being generated), and K_{xx} is the variance of the EHM measurements obtained from the covariance matrix previously calculated from post-silicon tests.

4.4. BER Extrapolation Modeling Problem Formulation

Given the two EHM measurement sets and the previously defined extrapolation technique, variations are expected in the resulting EHM_{12} value. Fig. 4.3 shows measurement distributions for EHM_1 and EHM_2, and the resulting over-estimated, under-estimated or accurate extrapolated margin. We aim to model the extrapolated response, $R_{\text{extrap}} \in \{-1, 0, 1\}$, denoting the three possible outcomes: an under-estimated extrapolation (−1), an accurate extrapolation (0), and an over-estimated extrapolation (1). The extrapolation accuracy is defined by a threshold ε_{th}, such that

$$R_{extrap} = \left[\!\left[\left| \boldsymbol{EHM}_{12} - \frac{1}{n}\sum_{i=1}^{n} EHM_{12,i} \right| > \varepsilon_{th} \right]\!\right] \left[\!\left[\operatorname{sgn}\left(\frac{1}{n}\sum_{i=1}^{n} EHM_{12,i} - \boldsymbol{EHM}_{12} \right) \right]\!\right] \qquad (4\text{-}6)$$

where sgn(·) is the sign function. In other words, EHM_{12} values that deviate below ε_{th} from the mean are under-estimated, values above ε_{th} from the mean are over-estimated, and values within ± ε_{th} from the mean are deemed accurate. Notice that subtraction operations in (4-6) are implemented element-wise.

We initially define the vector of model inputs, $\boldsymbol{x} \in \Re^{2n}$, as the two sets of EHM measurements:

$$\boldsymbol{x} = \left[\boldsymbol{EHM}_1 \;\; \boldsymbol{EHM}_2 \right]^{\mathrm{T}} \qquad (4\text{-}7)$$

where $\boldsymbol{EHM}_1$, $\boldsymbol{EHM}_2 \in \Re^n$. We therefore aim to find a model f such that

$$R_{extrap} = f(\boldsymbol{x}) \qquad (4\text{-}8)$$

The model inputs $\boldsymbol{x}$ can be extended to include additional statistical measurements for each EHM set. Analyzing each set of measurements as a distribution, we include as model inputs the standard deviation, minimum value, maximum value, skew, and kurtosis of each EHM set. Additionally, we divide the distribution in five equidistant bins, as illustrated in Fig. 4.3 for $\boldsymbol{EHM}_1$ and $\boldsymbol{EHM}_2$, and count the number of occurrences that a measurement falls within one of these five bins, as well as the number of bins with zero occurrences.

The bin-related inputs allow to measure the shape of the distribution. Once these statistical features are generated, the ratio between the $\boldsymbol{EHM}_1$ and $\boldsymbol{EHM}_2$ statistics is also calculated, i.e. we include the ratio of standard deviations, the ratio of skews, the ratio of kurtosis, etc., between the $\boldsymbol{EHM}_1$ and $\boldsymbol{EHM}_2$ distributions. Moreover, the two distributions can be combined in a single one by normalizing the mean values, and additional statistical information can be further derived. Therefore, the number of inputs can increase up to 50 when n, the number of EHM readouts, is 8. A full list of model inputs is shown in Table 4.1 for m EHM distributions, where j=1, 2, …, m is the j-th measured BER. The inputs selected are intended to provide additional information to improve the performance of the model in (4-8), thus enabling the proper classification of the extrapolated margins to ultimately enable test time acceleration without compromising on

TABLE 4.1. LIST OF INPUTS FOR THE MODELING PROBLEM FORMULATION

Input	Definition	Formulation
$\boldsymbol{EHM}_j$	Vector of size *n* of measurements at the *j*-th BER	Direct or synthetically generated measurements
SS_j	Standard deviation of $\boldsymbol{EHM}_j$	$\sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(x_i-\mu\right)^2}$
SK_j	Skew of $\boldsymbol{EHM}_1$ distribution	$\frac{E\left(x-\mu\right)^3}{\sigma^3}$
K_j	Kurtosis of $\boldsymbol{EHM}_j$ distribution	$\frac{E\left(x-\mu\right)^4}{\sigma^4}$
$MaxEHM_j$	Maximum value from $\boldsymbol{EHM}_j$	$\max(\boldsymbol{EHM}_j)$
$MinEHM_j$	Minimum value from $\boldsymbol{EHM}_j$	$\min(\boldsymbol{EHM}_j)$
$\boldsymbol{B}_j$	Vector of 5 bin values from the $\boldsymbol{EHM}_j$ distribution	Generate a linearly spaced histogram of 5 bins and count occurrences in each bin
nZB_j	Number of bins with a value of zero from the $\boldsymbol{EHM}_j$ distribution	$\sum_{i=1}^{5}\boldsymbol{B}_{1,i}=0$
SS_{all}	Standard deviation of the two distributions combined	$\sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(x_i-\mu\right)^2}$
SK_{all}	Skew of the two distributions combined	$\frac{E\left(x-\mu\right)^3}{\sigma^3}$
K_{all}	Kurtosis of the two distributions combined	$\frac{E\left(x-\mu\right)^4}{\sigma^4}$
$\boldsymbol{B}_{\mathrm{all}}$	Vector of 5 bin values from the two distributions combined	Generate a linearly spaced histogram of 5 bins and count occurrences in each bin
nZB_{all}	Number of bins with a value of zero from the two distributions combined	$\sum_{i=1}^{7}\boldsymbol{B}_i=0$
SS ratio	Ratio between SS_1 and SS_2	$\frac{SS_1}{SS_2}$
SK ratio	Ratio between SK_1 and SK_2	$\frac{SK_1}{SK_2}$
K ratio	Ratio between K_1 and K_2	$\frac{K_1}{K_2}$

accuracy.

As described in the next section, a deep neural network model to approximate (4-8) becomes a natural candidate for this modeling scenario.

4.5. Deep Neural Models

Conventional artificial neural networks (ANN), also known as shallow ANNs, typically use a few layers of parallel processing units (neurons), interconnected by artificial synapses [Haykin-99]. The most widely used topology for shallow ANNs, as applied to electrical, RF, and microwave circuits, is the 3-layer perceptron (3LP) [Rayas-Sanchez-04]. In contrast, deep neural networks (DNN) use a massive topology, with many layers each containing many neurons. DNNs are able to perform complex processing tasks remarkably well. The training procedure of these networks is usually performed via supervised learning algorithms, typically based on gradient descent techniques, where the input data is forward-propagated through the neuron layers, and the outputs are compared against previously obtained data, then the comparison errors are back-propagated through the network [Sebastian-19].

Shallow machine learning techniques (such as 3LP ANNs, support vector machines, and decision trees, among many others) are capable of transforming input data into one or two representation stages. In complex problems, techniques such as feature engineering are needed prior to making use of shallow learning methods, such that the inputs, or engineered features, can be processed to obtain the desired outputs [Chollet-17].

In deep learning, feature engineering is automated in such a way that while the model is being trained, it is learning about the required features. The two essential characteristics of deep learning are: 1) the layer-by-layer incremental way in which complex representations are generated, and 2) the fact that these incremental intermediate representations are learned jointly. This second characteristic is the reason why deep learning outperforms shallow learning models, even when two or more of them are stacked.

Shallow ANNs typically use smooth switch activation functions, such as the sigmoid and hyperbolic tangent functions; they are also usually trained with gradient-based algorithms. As the number of layers increases, the calculated gradients diminish exponentially. This is known as the vanishing gradient problem [Bengio-94] and [Hochreiter-01]. To overcome this, deep neural networks typically use the rectified linear unit (ReLu) as activation function. In a sigmoid function, the gradient becomes zero when the activation value is very high (large positive) or very low (large negative). However, in a ReLu function, the gradient is zero when the activation is negative and unit positive when the activation is positive (see Fig. 4.4). Recent advances also propose the use

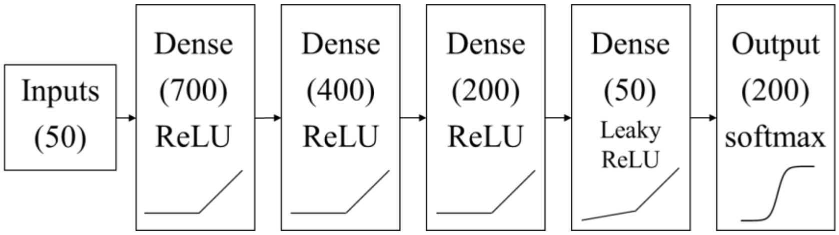


Fig. 4.4 Deep neural network (DNN) model topology, including the type of layer, the number of neurons (in parenthesis) and the type of activation function for the neurons on each layer.

of leaky ReLus, which allows for a small positive gradient when the unit is not active [Maas-13], parametric ReLus, which include a leakage coefficient that is also learned during the training procedure [He-15], and exponential linear units, which saturate to a negative value with smaller inputs [Clevert-15]. Due to the size of deep neural networks, the training procedure entails tuning a large number of hyper parameters, therefore, several techniques have been reported to improve training performance. Batch normalization is a commonly used technique to reduce the phenomenon known as internal covariate shift during training, by generating mini batches of inputs and normalizing within each batch [Ioffe-15] and [Thakkar-18]. To avoid overfitting, regularization techniques such as dropout are used. Dropout entails randomly dropping units and their corresponding connections, so even with small training data sets, sampling noise issues can be avoided [Srivastava-14] and [Gal-16].

The most common application area for DNNs usage is in classification problems, where the desired output is a categorical variable representing discrete values. However, networks with discrete variables present difficulties in back-propagation due to the inability to differentiate. To overcome this, a layer with softmax activation functions (see Fig. 4.4) is typically employed [Jang-17], where a discrete output with *k* different categories is encoded into a *k*-dimensional one-hot encoded vector, which is a vector that encodes a categorical output employing the one-hot encoding [Potdar-17]. In this manner, an output with three possible values is represented by a one-hot encoded vector composed of three bits, where each bit is a flag representing each category.

4.6. Proposed DNN Architecture

As mentioned before, we aim to exploit deep neural networks to model the BER extrapolation accuracy, due to the complexity and high dimensionality of the problem. We defined

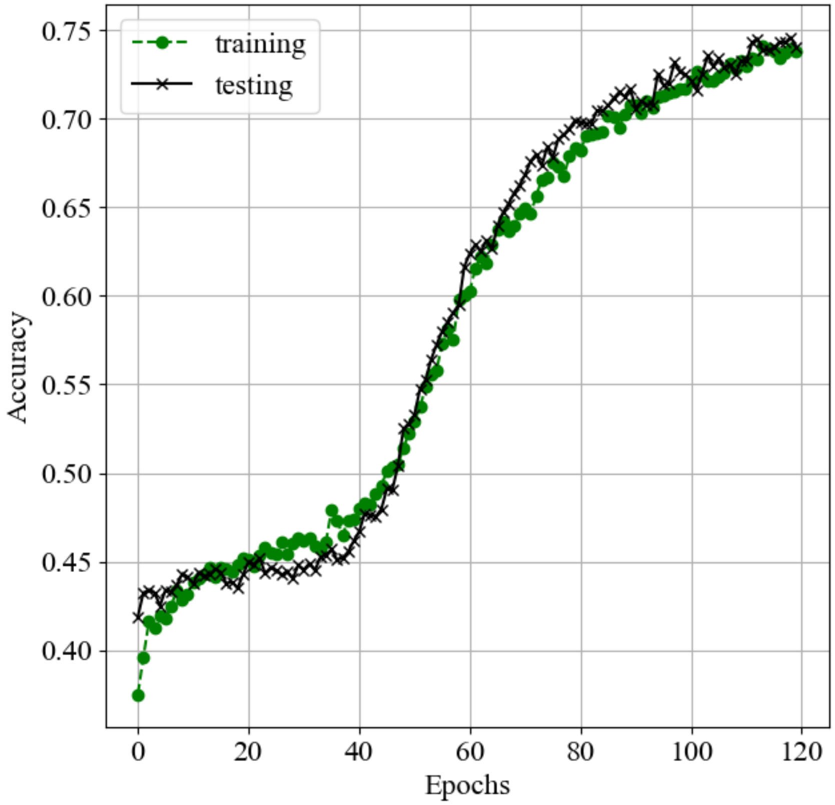


Fig. 4.5 Initial performance of the DNN model in terms of accuracy for the training (green dotted line) and testing (black crossed line) data sets.

in Chapter 4.4 the categorical output corresponding to the accurate, under-estimated, or over-estimated responses, as well as the vector of inputs related to statistical information from margin measurements.

Our model is comprised of a four-layer fully connected network, which uses a funnel structure encoding our input into a new representation on the embedded layer. The first three layers use ReLu as the activation function, while layer four uses a leaky ReLu function. Finally, since the problem we aim to solve is a classification problem in nature, the output layer employs softmax as the activation function. The topology of the network is depicted in Fig. 4.4. To avoid over-fitting, we use batch normalization in the leaky ReLu layer, and L2 normalization in the rest of the fully connected layers. We also exploit the dropout technique [Srivastava-14] in between each fully connected layer. The input layer is equal in size to the number of inputs, while the output layer is a vector of size 3, due to the one-hot encoding employed for the three possible categories of the extrapolated margin accuracy response previously defined.

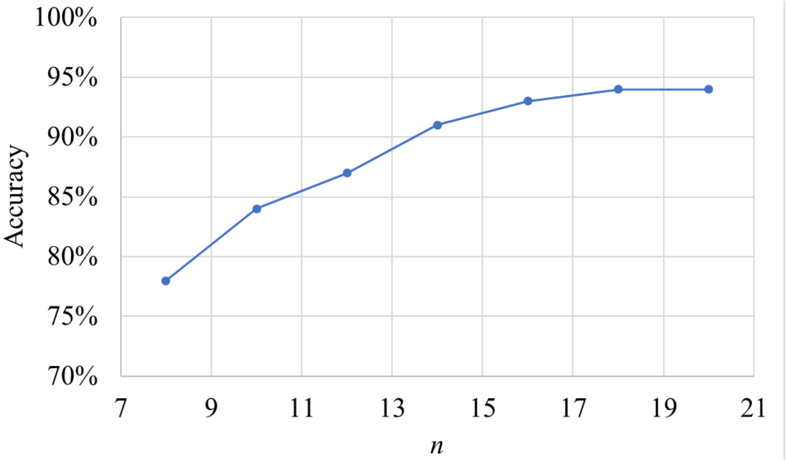


Fig. 4.6 Performance improvement of the DNN model by varying n, the number of EHM measurements.

To compile the model, we employ the Keras Python libraries, with Tensorflow as the backend. The training loss function we select is categorical cross-entropy, due to the nature of the problem. The algorithm selected for hyper-parameters optimization is Keras' implementation of the Adam optimizer algorithm [Kingma-15], using a learning rate of 0.001. We employ 10,000 test points for training, out of which 80% are randomly selected for learning and the remaining 20% are used for testing.

4.7. Results

As an initial approach, we define $n = 8$ as the number of eye height margin (EHM) measurements performed at each BER. This brings the total number of model inputs to 50, after including the statistical information calculated from the measurements. The value of ε_{th} (the threshold for the extrapolated response accuracy) we select is 1. Fig. 4.5 shows the performance of the DNN model during training. After training, the performance of the DNN model shows a 75% of accuracy (approximately), which is very similar to the performance obtained with shallow machine learning techniques (those results are omitted for the sake of brevity).

In order to improve accuracy, we increased n from the starting value of 8 up to 20, generating new training data sets for each value of n. Fig. 4.7 shows the accuracy improvement with respect to n; it is seen that accuracy is enhanced from approximately 78% to 94%. We also observe that starting from a value of $n = 18$, accuracy reaches a level of 94%. Further increasing n does not provide any improvements on the model accuracy. Moreover, the major improvement

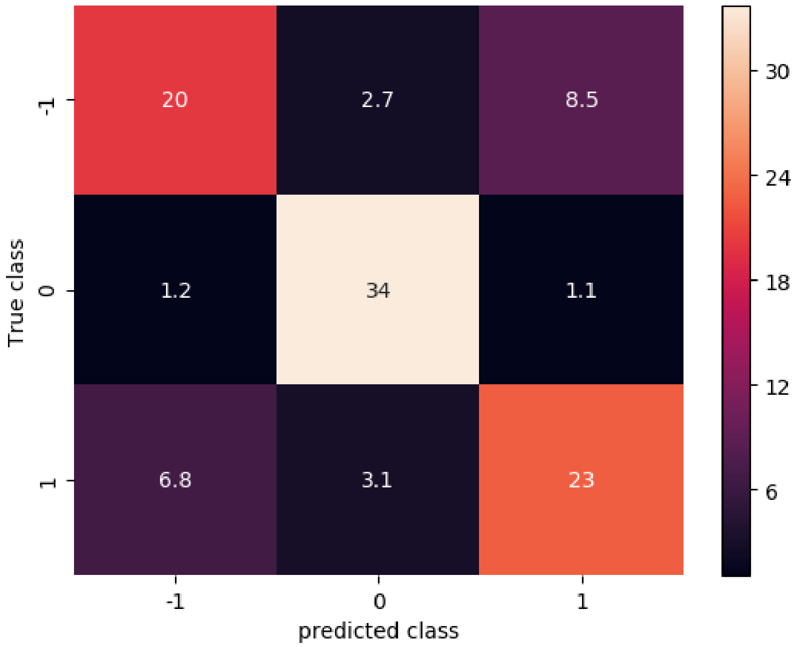


Fig. 4.7 Confusion matrix of the DNN model generated with $n = 8$.

found is on the under- and over-estimated responses. Fig. 4.7 shows the confusion matrix from the deep neural model trained with $n = 8$, while Fig. 4.8 shows the confusion matrix from the model trained with $n = 20$. The vertical colored bar in Fig. 4.7 and Fig. 4.8 represents the number of cases that fall within each classification case. By comparing Fig. 4.7 and Fig. 4.8, it is clear that not only does accuracy improves, but also sensitivity (the ability to select a true value in a true population) and specificity (the ability to rule out a failure in a failure-free population), thus improving the capability of the DNN model to provide valuable information on the extrapolated margin responses.

4.8. Conclusions

In this chapter, a modeling problem formulation to estimate the accuracy of BER extrapolation in Rx margin measurements of HSIO links was presented. The proposed formulation considers statistical information of eye height margin data at two different high BER values and establishes a metric to classify the extrapolated margin at a low BER value as over-estimated, under-estimated, or accurate. The formulation is then used in a deep neural network (DNN) model to efficiently classify the accuracy of the extrapolated receiver margin under a specification bit error rate (BER). The best DNN model accuracy was obtained by increasing the number of

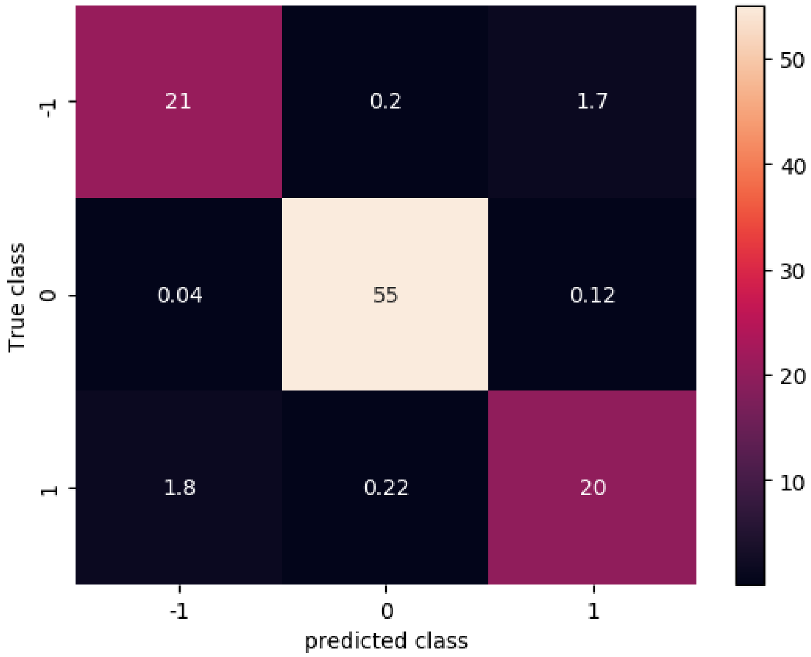


Fig. 4.8 Confusion matrix of the DNN model generated with $n = 20$.

measurement samples at each BER. By using the proposed DNN model, post-silicon validation is capable of extrapolating system margins to the specification BER to accelerate validation, and at the same time, classifying the accuracy of the extrapolated margins.

General Conclusions

Analog validation and test engineers need to find innovative solutions to tackle continuously increasing challenges, such as technology process shrinkage, product complexity increase, and data rates increase, among others, in order to enable product competitiveness in the market. This doctoral dissertation proposes machine learning and optimization techniques as feasible paths towards creating viable solutions to improve and accelerate validation and testing processes in the analog domain.

In Chapter 1, two optimization procedures were described to accelerate analog validation. The first one is based on DoE techniques to automatically optimize Rx eye diagrams in a PCIe gen2 HSIO link. DoE techniques allowed to find the most significant variables through a fractional factorial design, and then to find the optimal values of these variables using RSM CCD DoE models, which were optimized in closed form. The second proposed procedure exploited the golden search optimization approach to find the pass/fail boundary in an accelerated manner during JTOL execution, while at the same time achieving a high precision on the JTOL results by executing a downwards search at the target BER. This allows to accelerate JTOL tests in HSIO links by more than 90%.

PHY tuning is known to be one of the most time-consuming processes in analog validation, since current practices are based on exhaustive enumeration techniques or fully based on empirical expert knowledge. In Chapter 2 of this dissertation, a series of optimization methods were proposed to accelerate tuning by exploiting machine learning surrogate models. First, several surrogate modeling methods and DoE sampling approaches were evaluated to approximate the response of an HSIO link with respect to different EQ combinations. Then, with the optimal combination, SBO is performed to find the optimal EQ values. An ANN-based metamodeling technique was also described in Chapter 2, where a sufficiently accurate model was built with the least computational effort in the validation laboratory. Finally, using this low-cost machine learning model, we accelerated the PHY tuning process with the use of the ASM optimization algorithm. In Chapter 2, all results were based on a real validation platform to demonstrate the efficiency of our proposals.

Chapter 3 is focused on analog fault diagnosis in the testing field. First, it presented analog faults and fault modeling concepts. Then, a review of the most important fault diagnosis methods was presented. Finally, an analog faults diagnosis method based on ANN and constrained PE was proposed. The proposed method is based on comparing the circuit output response of a faulty circuit against a faultless one. This method was exemplified with two different analog circuits; results showed that the method efficiently identifies faults in both passive and active components. Then, the chapter also presented a generalized formulation to increase the uniqueness of the identified faults by using auxiliary information from internal circuit nodes. In Chapter 3, the machine learning techniques, based on ANN, were used as metamodels to identify faults with an extremely low computational cost, using an optimization process known as constrained statistical parameter extraction. Following the proposed methods, faults were properly identified in all the analog test case circuits.

When the modeling problem is too complex, traditional metamodels fail to provide the required performance, since high dimensionality is more suitable for deep learning techniques. In Chapter 4, a deep neural model was proposed to classify the precision of BER extrapolation of margins. The chapter first provided BER definitions, and BER extrapolation concepts. Then, an extrapolation modeling problem formulation was presented, by determining whether the extrapolated margin is over-estimated, under-estimated, or within a predefined error threshold, using statistical information derived from the measurements. Results show that by using our proposed DNN model, post-silicon validation and testing is capable of extrapolating margins to the specification BER to accelerate tests, while at the same time classifying the accuracy of the resulting margins.

In summary, this doctoral dissertation proposes a series of machine learning- and optimization-based methods to improve analog validation and test processes. Many proposals focus on accelerating the current industrial practices, such as JTOL tests and PHY tuning, to reduce TTM without sacrificing precision, and even improving results quality. Others, not only automate and accelerate current practices, but also improve precision on the testing and validation results, such as fault identification and BER extrapolation.

This doctoral dissertation offers a plethora of possible future research opportunities. Simple, yet efficient, optimization techniques, such as the golden section search used to accelerate JTOL tests, can be used in similar characterization tests in both validation and test scenarios, where the pass/fail boundary is not well known due to its analog nature, thus possibly offering multimillion dollar savings in test cost. DoE methodologies used in this dissertation to find the most significant EQ variables in tuning procedures, are open to be exploited for debug and triage activities, where variables interactions are not well understood.

Machine learning-based surrogate models have been proven effective in this dissertation for optimization-based tuning in a single test condition. However, validation engineers need to ensure product robustness against all valid conditions, such as process variations, temperature, and voltage conditions, channel loss specifications, and end point devices variations. So, future research should be focused on finding solutions to find optimal EQ values that satisfy all these multi-physics variables interactions. Additionally, parallel buses such as DDR have not been yet considered and pose a significant challenge in next-generation data rates for memory interfaces.

The fault identification methods presented in this dissertation have been proven to work for gross faults, i.e. short and open faults. Future work should also consider parametric faults; by doing this, fault diagnosis can be accelerated in many real-life analog faults scenarios, such as process shifts, aging effects, and non-catastrophic variations. Furthermore, as an initial approach, our fault-injection proposal is based on the premise that a single fault occurs at any given time. Future work must also consider multiple concurring faults.

The DNN models presented in this dissertation are limited to classifying whether the resulting extrapolated margin is precise or not. Future work should not only classify the extrapolation precision, but also predict the precise extrapolated margin. Future work could also delve into finding cost-effective models that can be implemented in memory-constrained architectures, such as embedded microcontrollers, so that the predicted extrapolated margins can be calculated in real-time within the next-generation PHY architectures.

Conclusiones generales

Ingenieros de validación y de pruebas analógicas necesitan encontrar soluciones innovadoras para abordar retos continuamente crecientes, tales como la permanente miniaturización de los procesos de tecnología, los incrementos en la complejidad de los productos, y el aumento en la velocidad de datos, entre otros, para lograr la competitividad de los productos en el mercado. Esta tesis doctoral propone técnicas de aprendizaje automático y de optimización como posibles caminos hacia la creación de soluciones viables para mejorar y acelerar los procesos de validación y pruebas en el dominio analógico.

En el Capítulo 1, dos procedimientos de optimización fueron descritos para acelerar la validación analógica. El primero está basado en técnicas de DoE para optimizar automáticamente diagramas de ojo en el receptor de un enlace HSIO de PCIe generación 2. Las técnicas DoE permitieron encontrar las variables más significativas mediante un diseño factorial fraccionado, y posteriormente ayudaron a encontrar los valores óptimos de dichas variables usando modelos DoE RSM CCD, los cuales fueron optimizados en forma cerrada. El segundo procedimiento propuesto aprovecha el enfoque de optimización de búsqueda áurea para encontrar la frontera de paso/falla de una manera acelerada durante la ejecución de JTOL, logrando al mismo tiempo una alta precisión en los resultados de JTOL al ejecutar una búsqueda hacia abajo en el BER objetivo. Esto permite acelerar pruebas de JTOL en enlaces HSIO por más del 90%.

La sintonización del PHY es uno de los procesos que más tiempo consumen en validación analógica, ya que las prácticas actuales se basan ya sea en técnicas de enumeración exhaustiva o bien son dependientes por completo del conocimiento experto empírico. En el Capítulo 2 de esta tesis, una serie de métodos de optimización fueron propuestos para acelerar la sintonización aprovechando modelos sustitutos de aprendizaje automático. Primero, varios métodos de modelado sustituto y técnicas de muestreo DoE fueron evaluadas para aproximar la respuesta de un enlace HSIO con respecto a diferentes combinaciones de EQ. Posteriormente, con la combinación óptima, se ejecutó SBO para encontrar los valores óptimos de EQ. Una técnica de meta-modelado basada en ANN también fue descrita en el Capítulo 2, donde se construyó un modelo suficientemente preciso con el menor esfuerzo computacional en un laboratorio de validación. Finalmente, usando este modelo de aprendizaje automático, aceleramos el proceso de sintonización del PHY con el uso del algoritmo ASM de optimización. En el Capítulo 2, todos los resultados están basados en una plataforma de validación industrial para demostrar la eficiencia de nuestras propuestas.

El Capítulo 3 está enfocado en el diagnóstico de fallas analógicas en el campo de pruebas. Primero presentó conceptos de fallas analógicas y de modelado de fallas. Luego, una revisión de los métodos de diagnóstico de fallas más importantes fue presentado. Finalmente, un método de diagnóstico de fallas analógicas basado en ANN y PE restringida fue propuesto. El método propuesto se basa en comparar la respuesta de salida de un circuito con fallas contra la de un circuito sin fallas. Este método fue ejemplificado con dos diferentes circuitos analógicos; los resultados mostraron que el método identifica eficientemente fallas tanto en componentes pasivos como activos. Luego, el capítulo también presenta una formulación generalizada para aumentar la unicidad de las fallas identificadas mediante el uso de información auxiliar de nodos circuitales internos. En el Capítulo 3, las técnicas de aprendizaje automático, basadas en ANN, fueron usadas como metamodelos para identificar fallas con un costo computacional extremadamente bajo, usando un proceso de optimización conocido como extracción de parámetros estadística restringida. Al seguir los métodos propuestos, las fallas fueron adecuadamente identificadas en todos los circuitos analógicos probados.

Cuando el problema de modelado es demasiado complejo, metamodelos tradicionales no logran proveer el rendimiento requerido, ya que la alta dimensionalidad es más adecuada para técnicas de aprendizaje profundo. En el Capítulo 4 se propuso un modelo neuronal profundo para clasificar la precisión de la extrapolación de BER de márgenes. Dicho capítulo primero provee definiciones de BER y conceptos de extrapolación de BER. Luego se presentó una formulación del problema de modelado de extrapolación, determinando si el margen extrapolado está sobreestimado, subestimado, o dentro de un límite de error predefinido, empleando información estadística derivada de mediciones. Los resultados muestran que al emplear el modelo DNN propuesto, la validación post-silicio y las pruebas son capaces de extrapolar márgenes al BER especificado para acelerar pruebas, y al mismo tiempo clasificar la precisión de los márgenes resultantes.

En resumen, esta tesis doctoral propone una serie de métodos basados en aprendizaje automático y en optimización para mejorar los procesos de validación y pruebas analógicas. Muchas de las propuestas se enfocan en acelerar las prácticas industriales actuales, como pruebas de JTOL y sintonización del PHY, para reducir TTM sin sacrificar precisión, e incluso, mejorando la calidad de los resultados. Otros, no solo automatizan y aceleran las prácticas actuales, sino también mejoran la precisión en los resultados de pruebas y de validación, como la identificación de fallas y la extrapolación de BER.

Esta tesis doctoral ofrece una plétora de posibles oportunidades de investigación futura. Técnicas de optimización simples, pero efectivas, como la búsqueda basada en la sección áurea usada para acelerar pruebas de JTOL, pueden ser usadas en pruebas similares de caracterización tanto en escenarios de validación como de pruebas, donde la frontera de paso/falla no es bien conocida debido a su naturaleza analógica, por lo que posiblemente ofrecerían ahorros multimillonarios en el costo de las pruebas. Metodologías DoE empleadas en esta tesis para encontrar las variables de EQ más significativas en procedimientos de sintonización, pueden ser aprovechadas para actividades de depuración y triaje, donde la interacción entre variables no es del todo conocida.

Los métodos de identificación de fallas presentados en esta tesis han sido probados que funcionan para fallas burdas, por ejemplo, fallas de cortocircuitos y circuitos abiertos. El trabajo futuro también podría considerar fallas paramétricas; al hacer eso, el diagnóstico de fallas puede ser acelerado en muchos escenarios reales de fallas analógicas, como cambios de proceso, efectos de envejecimiento y variaciones no catastróficas. Además, como un abordaje inicial, nuestra propuesta de inyección de fallas se basa en la premisa de que una única falla ocurre en determinado tiempo. El trabajo futuro también debería considerar múltiples fallas concurrentes.

Los modelos DNN presentados en esta tesis están limitados a clasificar si el margen resultante extrapolado de BER es preciso o no. Trabajo futuro podría no solo clasificar la precisión de la extrapolación de BER, sino también predecir el margen extrapolado preciso. El trabajo futuro también podría ahondar en encontrar modelos económicos o frugales que puedan ser implementados en arquitecturas con memoria restringida, como microcontroladores embebidos, para que los márgenes extrapolados predichos puedan ser calculados en tiempo real dentro de las arquitecturas de PHY de próxima generación.

Appendix

A. LIST OF INTERNAL RESEARCH REPORTS

1) A. Viveros-Wacher and J. E. Rayas-Sánchez, “Maximizing the area of a receiver eye diagram using central composite design,” Internal Report *PhDEngScITESO-16-09-R (CAECAS-16-06-R)*, ITESO, Tlaquepaque, Mexico, Jul. 2016.

2) A. Viveros-Wacher and J. E. Rayas-Sánchez, “An introduction to analog faults,” Internal Report *PhDEngScITESO-16-14-R (CAECAS-16-11-R)*, ITESO, Tlaquepaque, Mexico, Nov. 2016.

3) A. Viveros-Wacher and J. E. Rayas-Sánchez, “A brief review of fault diagnosis methods,” Internal Report *PhDEngScITESO-17-01-R (CAECAS-17-01-R)*, ITESO, Tlaquepaque, Mexico, Jan. 2017.

4) F. E. Rangel-Patiño, J. L. Chávez-Hurtado, A. Viveros-Wacher, J. E. Rayas-Sánchez, and N. Hakim, “Coarse surrogates for eye diagram system margining optimization in a server post-silicon validation platform,” Internal Report *PhDEngScITESO-17-08-R (CAECAS-17-06-R)*, ITESO, Tlaquepaque, Mexico, May 2017.

5) A. Viveros-Wacher and J. E. Rayas-Sánchez, “Analog fault modeling with artificial neural networks,” Internal Report *PhDEngScITESO-17-16-R (CAECAS-17-08-R)*, ITESO, Tlaquepaque, Mexico, May 2017.

6) F. E. Rangel-Patiño, J. E. Rayas-Sánchez, A. Viveros-Wacher, J. L. Chávez-Hurtado, E. A. Vega-Ochoa, and N. Hakim, “Post-silicon receiver equalization metamodeling by using artificial neural networks,” Internal Report *PhDEngScITESO-17-29-R (CAECAS-17-13-R)*, ITESO, Tlaquepaque, Mexico, Aug. 2017.

7) A. Viveros-Wacher, F. E. Rangel-Patiño, J. L. Chavez-Hurtado, and J. E. Rayas-Sánchez, “Eye diagram system margining surrogate modeling and optimization,” Internal Report *PhDEngScITESO-17-49-R (CAECAS-17-22-R)*, ITESO, Tlaquepaque, Mexico,

Dec. 2017.

8) F. E. Rangel-Patiño, J. E. Rayas-Sánchez, A. Viveros-Wacher, E. A. Vega-Ochoa, and N. Hakim, "High-speed links receiver optimization in post-silicon validation exploiting Broyden-based input space mapping," Internal Report *PhDEngScITESO-18-04-R (CAECAS-18-02-R)*, ITESO, Tlaquepaque, Mexico, Apr. 2018.

9) A. Viveros-Wacher, R. Baca-Baylon, F. E. Rangel-Patiño, M. A. Davalos-Santana, E. A. Vega-Ochoa, and J. E. Rayas-Sánchez, "Jitter tolerance acceleration using the golden section optimization technique," Internal Report *PhDEngScITESO-18-07-R (CAECAS-18-04-R)*, ITESO, Tlaquepaque, Mexico, Apr. 2018.

10) A. Viveros-Wacher and J. E. Rayas-Sánchez, "Analog fault identification in RF circuits using artificial neural networks and constrained parameter extraction," Internal Report *PhDEngScITESO-18-21-R (CAECAS-18-08-R)*, ITESO, Tlaquepaque, Mexico, Aug. 2018.

11) A. Viveros-Wacher, J. E. Rayas-Sánchez, and Z. Brito-Brito "A generalized formulation for neural modeling of analog gross faults," Internal Report *PhDEngScITESO-19-01-R (CAECAS-19-01-R)*, ITESO, Tlaquepaque, Mexico, Jan. 2019.

12) A. Viveros-Wacher, A. Norman, and J. E. Rayas-Sánchez, "An introduction to bit error rate extrapolation modeling," Internal Report *PhDEngScITESO-19-17-R (CAECAS-19-14-R)*, ITESO, Tlaquepaque, Mexico, Nov. 2019.

13) A. Viveros-Wacher, A. Norman, and J. E. Rayas-Sánchez, "Deep neural modeling for BER extrapolation," Internal Report *PhDEngScITESO-20-06-R (CAECAS-20-01-R)*, ITESO, Tlaquepaque, Mexico, Apr. 2020.

B. LIST OF PUBLICATIONS

B.1. Journal Papers

1) F. E. Rangel-Patiño, J. L. Chávez-Hurtado, **A. Viveros-Wacher**, J. E. Rayas-Sánchez, and N. Hakim, "System margining surrogate-based optimization in post-silicon validation," *IEEE Trans. Microwave Theory Techn.*, vol. 65, no. 9, pp. 3109-3115, Sep. 2017. (p-ISSN: 0018-9480; e-ISSN: 1557-9670; INSPEC: 17155317; published online: 29 May 2017; DOI: 10.1109/TMTT.2017.2701368)

2) F. E. Rangel-Patiño, **A. Viveros-Wacher**, J. E. Rayas-Sánchez, I. Durón-Rosales, E. A. Vega-Ochoa, N. Hakim, and E. López-Miralrio, "A holistic formulation for system margining and jitter tolerance optimization in industrial post-silicon validation," *IEEE Trans. Emerging Topics Computing*, vol. 8, no. 2, pp. 453-463, Apr.-Jun. 2020. (p-ISSN: 2376-4562; e-ISSN: 2168-6750; published online: 29 Sep. 2017; INSPEC: 19826401; DOI: 10.1109/TETC.2017.2757937)

3) F. E. Rangel-Patiño, J. E. Rayas-Sánchez, **A. Viveros-Wacher**, J. L. Chávez-Hurtado, E. A. Vega-Ochoa, and N. Hakim, "Post-silicon receiver equalization metamodeling by artificial neural networks," *IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems*, vol. 38, no. 4, pp. 733-740, Apr. 2019. (p-ISSN: 0278-0070; e-ISSN: 1937-4151; INSPEC: 18529158; published online: 8 May 2018; DOI: 10.1109/TCAD.2018.2834403)

4) **A. Viveros-Wacher**, J. E. Rayas-Sánchez, and Z. Brito-Brito, "Analog gross fault identification in RF circuits using neural models and constrained parameter extraction," *IEEE Trans. Microwave Theory Techn.*, vol. 67, no. 6, pp. 2143-2150, Jun. 2019. (p-ISSN: 0018-9480; e-ISSN: 1557-9670; INSPEC: 18734819; published online: 21 May 2019; DOI: 10.1109/TMTT.2019.2914106)

5) **A. Viveros-Wacher**, R. Baca-Baylón, F. E. Rangel-Patiño, J. L. Silva-Cortés, E. A. Vega-Ochoa, and J. E. Rayas-Sánchez, "Fast jitter tolerance testing for high-speed serial links in

post-silicon validation," *IEEE Trans. Microwave Theory Techn.*, 2021 (work submitted).

6) **A. Viveros-Wacher**, A. Norman, and J. E. Rayas-Sánchez, "Deep neural modeling of BER extrapolation in HSIO links," *IEEE Trans. Electromagnetic Compatibility*, 2021 (work to be submitted).

B.2. Conference Papers

1) **A. Viveros-Wacher** and J. E. Rayas-Sánchez, "Eye diagram optimization based on design of experiments (DoE) to accelerate industrial testing of high speed links," in *IEEE MTT-S Latin America Microwave Conf. (LAMC-2016),* Puerto Vallarta, Mexico, Dec. 2016, pp. 1-3. (e-ISBN: 978-1-5090-4287-6, p-ISBN: 978-1-5090-4288-3, INSPEC: 16670752, DOI: 10.1109/LAMC.2016.7851249).

2) F. Rangel-Patino, **A. Viveros-Wacher**, J. E. Rayas-Sánchez, E. A. Vega-Ochoa, I. Duron-Rosales, and N. Hakim, "A holistic methodology for system margining and jitter tolerance optimization in post-silicon validation," in *IEEE MTT-S Latin America Microwave Conf. (LAMC-2016),* Puerto Vallarta, Mexico, Dec. 2016, pp. 1-4. (ISBN: 978-1-5090-4288-3; e-ISBN: 978-1-5090-4287-6; INSPEC: 16670749; DOI: 10.1109/LAMC.2016.7851268).

3) F. Rangel-Patino, J. L. Chávez-Hurtado, **A. Viveros-Wacher**, J. E. Rayas-Sánchez, and N. Hakim, "Eye diagram system margining surrogate-based optimization in a server silicon validation platform," in *European Microwave Conf. (EuMC-2017),* Nuremberg, Germany, Oct. 2017, pp. 540-543. (ISBN: 978-1-5386-3964-1; e-ISBN: 978-2-87487-047-7; https://www.researchgate.net/publication/323571676).

4) **A. Viveros-Wacher**, R. Baca-Baylón, F. E. Rangel-Patiño, M. A. Dávalos-Santana, E. A. Vega-Ochoa, and J. E. Rayas-Sánchez, "Jitter tolerance acceleration using the golden section optimization technique," in *IEEE Latin American Symp. Circuits and Systems Dig. (LASCAS 2018),* Puerto Vallarta, Mexico, Feb. 2018, pp. 1-4. (e-ISSN: 2473-4667; p-

ISBN: 978-1-5386-2312-1; e-ISBN: 978-1-5386-2311-4; DOI: 10.1109 /LASCAS.2018.8399908).

5) **A. Viveros Wacher** and J. E. Rayas-Sánchez, "Analog fault identification in RF circuits using artificial neural networks and constrained parameter extraction," in *2018 IEEE MTT-S Int. Conf. on Numerical Electromagnetic and Multiphysics Modeling and Optimization (NEMO)*, Reykjavik, Iceland, Aug. 2018, pp. 1-3. (e-ISBN: 978-1-5386-5204-6; p-ISBN: 978-1-5386-5205-3; DOI: 10.1109/NEMO.2018.8503117).

6) F. E. Rangel-Patiño, J. E. Rayas-Sánchez, **A. Viveros-Wacher**, E. A. Vega-Ochoa, and N. Hakim, "High-speed links receiver optimization in post-silicon validation exploiting Broyden-based input space mapping," in *2018 IEEE MTT-S Int. Conf. on Numerical Electromagnetic and Multiphysics Modeling and Optimization (NEMO)*, Reykjavik, Iceland, Aug. 2018, pp. 1-3. (e-ISBN: 978-1-5386-5204-6; p-ISBN: 978-1-5386-5205-3; DOI: 10.1109/NEMO.2018.8503099).

7) J. E. Rayas-Sánchez**,** F. E. Rangel-Patiño, **A. Viveros-Wacher**, J. L. Chávez-Hurtado, J. R. del-Rey, F. Leal-Romo, and Z. Brito-Brito, "Industry-oriented research projects on computer-aided design of high-frequency circuits and systems at ITESO Mexico," in *European Microwave Conf. (EuMC-2018)*, Madrid, Spain, Sept. 2018, pp. 588-591. (p-ISBN: 978-1-5386-5285-5; e-ISBN: 978-2-87487-051-4; https://www.researchgate.net/publication/328346442).

Bibliography

[Abdul-11] M. A. Abdul Latif, N. B. Zain Ali and F. A. Hussin, “IDVP (intra-die variation probe) for system-on-chip (SoC) infant mortality screen,” in *IEEE International Symposium of Circuits and Systems (ISCAS)*, Rio de Janeiro, May 2011, pp. 2055-2058.

[Agilent-05] Agilent Tech (2005), *Total Jitter Measurement at Low Probability Levels, Using Optimized BERT Scan Methods* [Online]. Available: www.agilent.com

[Alwi-14] H. Alwi and C. Edwards, “Robust fault reconstruction for linear parameter varying systems using sliding mode observers,” *Int. J. Robust Nonlin. Control*, vol. 24, no. 14, pp. 1947–1968, Sep. 2014.

[Angiulli-07] G. Angiulli, M. Cacciola, and M. Versaci, “Microwave devices and antennas modelling by support vector regression machines,” *IEEE Trans. Magn.*, vol. 43, no. 4, pp. 1589-1592, Apr. 2007.

[Arogeti-12] S. Arogeti, D. Wang, C. Low, and M. Yu, “Fault detection isolation and estimation in a vehicle steering system,” *IEEE Trans. Ind. Electron.*, vol. 59, no. 12, pp. 4810–4820, Dec. 2012.

[Athavale-05] A. Athavale and C. Christensen, *High-Speed Serial I/O Made Simple A Designer's Guide, with FPGA Applications*, San Jose, CA: Xilinx Connectivity Solutions, 2005.

[Bandler-85] J. W. Bandler and A. E. Salama, “Fault diagnosis of analog circuits,” *Proc. IEEE,* vol. 73, no. 8, pp. 1279-1325, Aug. 1985.

[Bandler-95] J. W. Bandler, R. M. Biernacki, S. H. Chen, R. H. Hemmers, and K. Madsen, “Electromagnetic optimization exploiting aggressive space mapping,” *IEEE Trans. Microw. Theory Techn.*, vol. 41, no. 12, pp. 2874-2882, Dec. 1995.

[Bandler-04] J. W. Bandler, Q. Cheng, S. A. Dakroury, A. S. Mohamed, M. H. Bakr, K. Madsen and J. Søndergaard, “Space mapping: the state of the art,” *IEEE Trans. Microwave Theory Tech.*, vol. 52, no. 1, pp. 337-361, Jan. 2004.

[Barsky-04] R. Barsky and I. A. Wagner, “Electromigration-dependent parametric yield estimation,” in *Proceedings of the 11th IEEE International Conference on Electronics*, Circuits and Systems, Tel Aviv, Israel, Dec. 2004, pp. 121-124.

[Benbouzid-00] M. Benbouzid, “A review of induction motor signature analysis as a medium for faults detection,” *IEEE Trans. Ind. Electron.*, vol. 47, no. 5, pp. 984–993, Oct. 2000.

[Bengio-94] Y. Bengio, P. Simard, and P. Frasconi, “Learning long-term dependencies with gradient descent is difficult,” *IEEE Trans. Neural Netw.*, vol. 5, no. 2, pp. 157–166, Mar. 1994.

[Bergano-93] N. S. Bergano, F. W. Kerfoot, and C. R. Davidsion, “Margin measurements in optical amplifier system,” *IEEE Photonics Technology Letters*, vol. 5, no. 3, pp. 304-306, Mar. 1993.

BIBLIOGRAPHY

[Beyene-07] W. T. Beyene, “Application of artificial neural networks to statistical analysis and nonlinear modeling of high-speed interconnect systems,” *IEEE Trans. Computer-Aided Design of Integrated Circuits Systems*, vol. 26, no. 1, pp. 166-176, Jan. 2007.

[Bhatta-13] D. Bhatta, I. Mukhopadhyay, S. Natarajan, P. Goteti, and B. Xue, “Framework for analog test coverage,” in *Int. Symp. on Quality Electronic Design (ISQED)*, Santa Clara, CA, Mar. 2013, pp. 468-475.

[Binu-19] D. Binu and B. S. Kariyappa, “RideNN: a new rider optimization algorithm-based neural network for fault diagnosis in analog circuits,” *IEEE Trans. Instrumentation and Measurement*, vol. 68, no. 1, pp. 2-26, Jan 2019.

[Bistola-15] S. Bistola, “High-speed interconnect simulation using artificial neural networks”, in *Intel Design & Test Technology Conference (DTTC)*, Oregon, CA, Oct. 2015.

[Blish-97] R.C. Blish, “Temperature cycling and thermal shock failure rate modeling,” in *IEEE International Reliability Physics Symposium*, 35th Annual Proceedings, Denver, CO, 1997, pp. 110-117.

[Booker-99] A. J. Booker, J. E. Dennis Jr., P. D. Frank, D. B. Serafini, V. Torczon, and M. W. Trosset, “A rigorous framework for optimization of expensive functions by surrogates,” *Struct. Optim.*, vol. 17, pp. 1–13, Feb. 1999.

[Brito-Brito-09] Z. Brito-Brito, I. Llamas-Garro, G. Navarro-Munoz, J. Perruisseau-Carrier, and L. Pradell, “UMTS-WiFi switchable bandpass filter,” in *2009 European Microw. Conf. (EuMC)*, Rome, 2009, pp. 125-128.

[Bushnell-02] M. Bushnell and V. Agrawal, *Essentials of Electronic Testing for Digital, Memory and Mixed-Signal VLSI Circuits*, New York, NY: Springer, 2002.

[Cabasino-10] M. Cabasino, A. Giua, and C. Seatzu, “Fault detection for discrete event systems using Petri nets with unobservable transitions,” *Automatica*, vol. 46, no. 9, pp. 1531–1539, Sep. 2010.

[Campos-Delgado-08] D. Campos-Delgado, D. Espinoza-Trejo, and E. Palacios, “Fault-tolerant control in variable speed drives: A survey,” *IET Elect. Power Appl.*, vol. 2, no. 2, pp. 121–134, Apr. 2008.

[Chakrabarty-98] S. Chakrabarty, V. Rajan, J. Ying, M. Mansjur, K. Pattipati, and S. Deb, “A virtual test-bench for analog circuit testability analysis and fault diagnosis,” in *Proc. IEEE Syst. Readiness Tech. Conf. Test Technology for the 21st Century*, Salt Lake City, UT, Aug. 1998, pp. 337-352.

[Chang-05] Y. P. Chang et al, “Design of discrete-value passive harmonic filters using sequential neural-network approximation and orthogonal array”, in *IEEE PES Transmission and Distribution Conf. & Expo.: Asia and Pacific*, Dalian, China, Aug. 2005, pp. 1-6.

[Chen-15] D. Chen and B. K.-W. Mak, “Multitask learning of deep neural networks for low-resource speech recognition,” *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 23, no. 7, pp. 1172–1183, Jul. 2015.

[Cheng-00] J. Cheng and M. J. Druzdzel, “Computational investigation of low-discrepancy sequences in simulation algorithms for Bayesian networks”, in *Proc. Conf. Uncertainty in Artif. Intell.*, San Francisco, CA, Jun. 2000, pp. 72-81.

[Cheng-11] Y. S. Cheng and R. B. Wu, “Direct eye diagram optimization for arbitrary transmission lines using FIR filter,” *IEEE Trans. Components, Packaging and Manufacturing Technology,* vol. 1, no. 8, pp. 1250-1258, Aug. 2011.

[Cho-10] H. Cho, J. Knowles, M. Fadali, and K. Lee, “Fault detection and isolation of induction motors using recurrent neural networks and dynamic Bayesian modelling,” *IEEE Trans. Control Syst. Technol.*, vol. 18, no. 2, pp. 430–437, Feb. 2010.

[Cho-14] K. Cho et al., “Learning phrase representations using RNN encoderdecoder for statistical machine translation,” in *Proc. Conf. Empirical Methods Natural Lang. Process.*, Doha, Qatar, Oct. 2014, pp. 1724–1734.

[Chollet-17] F. Chollet, *Deep Learning with Python*. Shelter Island, NY, USA: Manning Publications, 2017.

[Chong-96] E. K. P. Chong and S. H. Zak, *An Introduction to Optimization*, New York, NY: Wiley-Interscience, 1996.

[Chunlai-16] L. Chunlai, Z. Xianshuang, and Gudake, “A survey of online fault diagnosis for PV module based on BP neural network,” in *Intl. Conf. on Smart City and Systems Engineering (ICSCSE)*, Hunan, China, Nov. 2016, pp. 483-486.

[Clevert-15] D. A. Clevert, T. Unterthiner, and S. Hochreiter, “Fast and accurate deep network learning by exponential linear units (elus),” in *Int. Conf. Learning Representations (ICLR)*, San Jose, Puerto Rico, May 2015, pp. 1-14.

[Collobert-08] R. Collobert and J. Weston, “A unified architecture for natural language processing: deep neural networks with multitask learning,” in *Proc. Int. Conf. Mach. Learn.*, Helsinki, Finland, Jul. 2008, pp. 160–167.

[Crosby-95] Crosby PB, *Quality Without Tears: The Art of Hassle-Free Management*. New York, NY: McGraw-Hill, 1995.

[Dasgupta-91] A. Dasgupta, M. Pecht, “Material failure mechanisms and damage models,” *IEEE Transactions on Reliability*, vol. 40, no. 5, Dec. 1991.

[DeOrio-13] A. DeOrio, Q. Li, M. Burgess, and V. Bertacco, “Machine learning-based anomaly detection for post-silicon bug diagnosis,” in *Europe Conf. & Exhibition (DATE) in Design, Automation & Test*, Grenoble, France, March 2013.

[Deyati-14] S. Deyati, B. Muldrey, A. Banerjee, and A. Chatterjee, “Atomic model learning: A machine learning paradigm for post silicon debug of RF/analog circuits,” in *IEEE 32nd VLSI Test Symp. (VTS)*, Napa, CA, April 2014.

[Eichelberger-77] E. B. Eichelberger and T.W. Williams, “A logic design structure for LSI testing,” in *Proceedings, IEEE/ACM 14th Design Automation Conference*, Jan. 1977, pp. 462-468.

[Erb-09] S. Erb and W. Pribyl, “An accurate and efficient method for BER analysis in high-speed communication systems,” in *2009 European Conf. Circuit Theory and Design*, Antalya, Turkey, Aug. 2009, pp. 731-734.

[Erb-10] S. Erb and W. Pribyl, “Comparison of jitter decomposition methods for BER analysis of high-speed serial links,” in *13th IEEE Symp. Design and Diagnostics of Electronic Circuits and Systems*, Vienna, Austria, Jun. 2010, pp. 370-375.

[Erb-12] S. Erb and W. Pribyl, “Design specification for BER analysis methods using built-in jitter measurements,” *IEEE Trans. Very Large Scale Integration (VLSI) Systems*, vol. 20, no. 10, pp. 1804-1817, Oct. 2012.

[Fan-09a] Y. Fan and Z. Zilic, “Accelerating jitter tolerance qualification for high speed serial interfaces,” in *10th Int Symp on Quality Electronic Design*, San Jose, CA, Mar. 2009.

[Fan-09b] Y. Fan and Z. Zilic, “A versatile scheme for the validation, testing and debugging of high speed serial interfaces,” in *IEEE International High Level Design Validation and Test Workshop*, San Francisco, CA, 2009, pp. 114-121.

[Fan-11] Y. Fan and Z. Zilic, *Accelerating Test, Validation and Debug of High Speed Serial Interfaces*, New York, NY: Springer, 2011.

[Farabet-13] C. Farabet, C. Couprie, L. Najman, and Y. LeCun, “Learning hierarchical features for scene labeling,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1915–1929, Aug. 2013.

[Feng-13] Z. Feng, M. Liang, and F. Chu, “Recent advances in time–frequency analysis methods for machinery fault diagnosis: A review with application examples,” *Mech. Syst. Signal Process.*, vol. 38, no. 1, pp. 165–205, Jul. 2013.

[Gal-16] Y. Gal and Z. Ghahramani, “Dropout as a Bayesian approximation: Representing model uncertainty in deep learning,” in *Int. Conf. Machine Learning*, New York, NY, Jun. 2016, pp. 1050-1059.

[Gao-04] Z. Gao and D. Ho, “Proportional multiple-integral observer design for descriptor systems with measurement output disturbances,” *Proc. Inst. Elect. Eng.—Control Theory Appl.*, vol. 151, no. 3, pp. 279–288, May 2004.

[Gao-07] Z. Gao, S. Ding, and Y. Ma, “Robust fault estimation approach and its application in vehicle lateral dynamic systems,” *Opt. Control Appl. Methods*, vol. 28, no. 3, pp. 143–156, May 2007.

[Gao-08] Z. Gao, T. Breikin, and H. Wang, “Discrete-time proportional and integral observer and observer-based controller for systems with both unknown input and output disturbances,” *Opt. Control Appl. Methods*, vol. 29, no. 3, pp. 171–189, May 2008.

[Gao-10] D. Gao, C. Wu, B. Zhang, and X. Ma, “Signed directed graph and qualitative trend analysis based fault diagnosis in chemical industry,” *Chin. J. Chem. Eng.*, vol. 18, no. 2, pp. 265–276, Apr. 2010.

[Gao-15a] Z. Gao, C. Cecati, and S. X. Ding, “A survey of fault diagnosis and fault-tolerant techniques—Part I: Fault diagnosis with model-based and signal-based approaches,” *IEEE Trans. on Ind. Electron.*, vol. 62, no. 6, pp. 3757-3767, Jun. 2015.

[Gao-15b] Z. Gao, C. Cecati and S. X. Ding, "A survey of fault diagnosis and fault-tolerant techniques—Part II: Fault diagnosis with knowledge-based and hybrid/active approaches," *IEEE Trans. on Ind. Electron.*, vol. 62, no. 6, pp. 3768-3774, Jun. 2015.

[Garistelov-12] O. Garitselov, S. P. Mohanty, and E. Kougianos, “A comparative study of metamodels for fast and accurate simulation of nano-CMOS circuits,” *IEEE Trans. Semicond. Manufac.*, vol. 25, no.1, pp. 26-36, Feb. 2012.

[Goel-81] P. Goel, “An Implicit enumeration algorithm to generate tests for combinational circuits,” *IEEE Transactions on Computers*, vol. C-30, no. 3, pp. 215-222, March 1981.

[Goulermas-07] J. Y. Goulermas, P. Liatsis, Z. Xiao-Jun, and P. Cook, “Density-driven generalized regression neural networks (DD-GRNN) for function approximation,” *IEEE Trans. Neural Netw.*, vol. 18, no. 6, pp. 1683-1696, Nov. 2007.

[Goay-17] C. H. Goay and P. Goh, “Neural networks for eye height and eye width prediction with an improved adaptive sampling algorithm,” in *Asian Simulation Conf. (AsiaSim)*, Melaka, Malaysia, Aug. 2017.

[Grzechca-02] D. Grzechca and J. Rutkowski, “Use of neural network and fuzzy logic to time domain analog tasting,” in *Proc. Int. Conf. Neural Inform. Processing*, 2002. ICONIP '02, Nov. 2002, vol. 5, pp. 2601-2604.

[Grzechca-11] D. Grzechca, “Simulated annealing with artificial neural network fitness function for ECG amplifier testing,” in *European Conf. Circuit Theory and Design (ECCTD)*, Linkoping, Sweden, Aug. 2011, pp. 49-52.

[Gu-12] C. Gu, “Challenges in post-silicon validation of high-speed I/O links,” in *2012 IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, San Jose, CA, 2012, pp. 547-550.

[Guoming-15] S. Guoming, L. Qi, L. Gang, J. Shuyan, and W. Houjun, “Analog circuit fault diagnosis using wavelet feature optimization approach,” in *IEEE Int. Conf. Electronic Measurement & Instruments (ICEMI)*, Qingdao, China, Jul. 2015, pp. 119-124.

[Haijun-12] L. Haijun, F. Yao, X. Zhicheng, and Z. Xuhui, “The study about feature selection of analog circuit fault diagnosis based on annealing genetic hybrid algorithm,” in *Int. Forum Strategic Technology (IFOST)*, Tomsk, Russia, Sept. 2012, pp. 1-4.

[Haini-07] Q. Haini, X. Weisheng and Y. Youling, “Design of neural network output layer in fault diagnosis of analog circuit,” in *Int. Conf. Electronic Measurement Instruments*, Xi'an, China, Aug. 2007, pp. 3-639-3-642.

[Ham-05] B. Ham, “Methodologies for jitter and signal quality specification,” *INCITS Tech. Rep.*, June 2005.

[Hansun-17] S. Hansun and M. B. Kristanda, “Performance analysis of conventional moving average methods in forex forecasting,” in *Int. Conf. Smart Cities, Automation & Intelligent Computing Systems (ICON-SONICS)*, Yogyakarta, Indonesia, Nov. 2017, pp. 11-17.

[Haykin-99] S. Haykin, *Neural Networks: A Comprehensive Foundation*. New Jersey, MA: Prentice Hall, 1999.

[He-00] J-Z. He, Z-H. Zhou, X-R. Yin, and S-F. Chen, “Using neural networks for fault diagnosis,” in *Proc. Int. Joint Conf. Neural Networks (IJCNN)*, Como, Italy, Jul. 2000, pp. 217-220.

[He-13a] J-Z. He, Z-H. Zhou, X-R. Yin, and S-F. Chen, “Using neural networks for fault diagnosis,” in *Proc. of the Intl. Joint Conf. on Neural Networks (IJCNN)*, Como, Italy, July 2000, pp. 217-220.

[He-13b] D. He, R. Li, and J. Zhu, “Plastic bearing fault diagnosis based on a twostep data mining approach,” *IEEE Trans. Ind. Electron.*, vol. 60, no. 8, pp. 3429–3440, Aug. 2013.

[He-15] K. He, X. Zhang, S. Ren, and J. Sun, “Delving deep into rectifiers: Surpassing human-level performance on imagenet classification,” in *Proc. IEEE Int. Conf. Computer Vision*, Santiago, Chile, Dec. 2015, pp. 1026-1034.

BIBLIOGRAPHY

[Hochreiter-01] S. Hochreiter, Y. Bengio, P. Frasconi, and J. Schmidhuber, “Gradient flow in recurrent nets: the difficulty of learning long-term dependencies,” in *A Field Guide to Dynamical Recurrent Neural Networks*, S. C. Kremer and J. F. Kolen, Eds. Piscataway, NJ, USA: IEEE Press, 2001.

[Hodgkiss-83] W. Hodgkiss and L. F. Turner. “Practical equalization and synchronization strategies for use in serial data transmission over H.F. channels,” *Radio and Electronic Engineer*, vol. 53, no. 4, pp. 141-145, Apr. 1983.

[Hong-08] D. Hong and K.T. Cheng, “Bit-error rate estimation for bang-bang clock and data recovery circuit,” in *26th IEEE VLSI Test Symp.*, San Diego, CA, April, 2008, pp. 17-22.

[Hwang-10] I. Hwang, S. Kim, Y. Kim, and C. Seah, “A survey of fault detection, isolation, and reconfiguration methods,” *IEEE Trans. Control Syst. Technol.*, vol. 18, no. 3, pp. 636–653, May 2010.

[IEEE-93] *IEEE Standard Test Access Port and Boundary Scan Architecture*, IEEE Standard 1149.1 – 1990. The IEEE, Inc., New York, 1993.

[IEEE-18] *IEEE Standard for Ethernet*, IEEE Std 802.3-2018 (Revision of IEEE Std 802.3-2015), pp.1-5600, 31 Aug. 2018, doi: 10.1109/IEEESTD.2018.8457469.

[Ioffe-15] S. Ioffe, and C. Szegedy, “Batch normalization: accelerating deep network training by reducing internal covariate shift,” *arXiv preprint arXiv:1502.03167*, 2015.

[Jang-17] E. Jang, S. Gu, and, B. Poole, “Categorical reparameterization with gumbel-softmax,” in *Int. Conf. Learning Representations (ICLR)*, Toulon, France, Apr. 2017, pp. 1-13.

[Jin-19] J. Jin, C. Zhang, F. Feng, W. Na, J. Ma, and Q. Zhang, “Deep neural network technique for high-dimensional microwave modeling and applications to parameter extraction of microwave filters,” *IEEE Trans. Microwave Theory Techn.*, vol. 67, no. 10, pp. 4140-4155, Oct. 2019.

[Kabisatpathy-05] P. Kabisatpathy, A. Barua, and S. Sinha, *Fault Diagnosis of Analog Integrated Circuits*, New York, NY: Springer, 2005.

[Katipamula-05a] S. Katipamula and M. Brambley, “Methods for fault detection, diagnostics and prognostics for building systems—A review, Part I,” *HVAC & R Res.*, vol. 11, no. 1, pp. 3–25, Jan. 2005.

[Katipamula-05b] S. Katipamula and M. Brambley, “Methods for fault detection, diagnostics and prognostics for building systems—A review, Part II,” *HVAC & R Res.*, vol. 11, no. 2, pp. 169–187, Apr. 2005.

[Keliris-13] C. Keliris, M. Polycarpou, and T. Parisini, “A distributed fault detection filtering approach for a class of interconnected continuous time nonlinear systems,” *IEEE Trans. Autom. Control*, vol. 58, no. 8, pp. 2032–2047, Aug. 2013.

[Keshava-10] J. Keshava, N. Hakim, C. Prudvi, “Post-silicon validation challenges: How EDA and academia can help,” *Design Automation Conference (DAC), 47th ACM/IEEE,* June 2010, pp. 3 –7.

[Kim-06] Young-Han Kim, “Feedback capacity of the first-order moving average Gaussian channel,” *IEEE Trans. Information Theory*, vol. 52, no. 7, pp. 3063-3079, Jul. 2006.

[Kingma-15] D. P. Kingma and J. Ba, “Adam: a method for stochastic optimization,” in *Int. Conf. Learning Representations (ICLR)*, San Diego, CA, May 2015, pp. 1-15.

[Koenemann-98] B. Koenemann, “Creature of the deep submicron lagoon,” invited lecture in *10th Workshop Test Methods and Reliability of Circuits and Systems*, Herrenberg, Germay, March 1998.

[Koenig-05] D. Koenig, “Unknown input proportional multiple-integral observer design for linear descriptor systems: Application to state and fault estimation,” *IEEE Trans. Autom. Control*, vol. 50, no. 2, pp. 212–217, Feb. 2005.

[Koren-10] I. Koren and C. M. Krishna, *Fault Tolerant Systems*, California, USA: Morgan Kaufmann, 2010.

[Kuo-04] A. Kuo, T. Farahmand, N. Ou, S. Tabatabaei, and A. Ivanov, “Jitter models and measurement methods for high-speed serial interconnects,” in *Proc. IEEE Int. Test Conf.*, Charlotte, NC, Oct. 2004, pp. 1295-1302.

[Lagarias-98] J. C. Lagarias, J. A. Reeds, M. H. Wright, and P. E. Wright, “Convergence properties of the Nelder-Mead simplex method in low dimensions,” *SIAM J. on Optimization*, vol. 9, no. 1, pp. 112–147, 1998.

[Lee-11] B. T. Lee, M. Mazumder, and R. Mellitz, “High speed differential I/O overview and design challenges on Intel enterprise server platforms,” in *IEEE Int. Symp. on Electromagnetic Compatibility*, Long Beach, CA, Aug. 2011, pp. 779-784.

[Li-09] H. Li and Y. Zhang, “An algorithm of soft fault diagnosis for analog circuit based on the optimized SVM by GA,” in *Int. Conf. Electronic Measurement & Instruments*, Beijing, China, Aug. 2009, pp. 4-1023-4-1027.

[Li-10] M. Li, Y. He, L. Yuan, and M. Li, “Fault diagnosis of analog circuit based on wavelet neural networks and chaos differential evolution algorithm,” in *Int. Conf. Electrical and Control Eng.*, Wuhan, China, Nov. 2010, pp. 986-989.

[Li-12] X. Li, Y. Zhang, S. Wang, and G. Zhai, “Analog circuits fault diagnosis by GA-RBF neural network and virtual instruments,” in *Int. Symp. on Instrumentation & Measurement, Sensor Network and Automation (IMSNA)*, Sanya, China, Aug. 2012, pp. 236-239.

[Liang-03] G. Liang and Y. He, “A fault identification approach for analog circuits using fuzzy neural network mixed with genetic algorithms,” in *Proc. IEEE Int. Conf. on Robotics, Intelligent Systems and Signal Processing*, Changsha, Hunan, China, Oct. 2003, pp. 1267-1272 vol.2.

[Lind-15] D. A. Lind, W. G. Marchal and S.A. Wathen, *Estadistica Aplicada a los Negocios y la Economia*, New York, NY: McGraw-Hill, 2015.

[Liu-87] R.W. Liu, *Selected Papers on Analog Fault Diagnosis*, Advances in Circuits and Systems, New York, NY: IEEE Press, 1987.

[Liu-91] R. Liu, *Testing and Diagnosis of Analog Circuits and Systems*. New York: Van Nostrand Reinhold, 1991

[Liu-15] M. Liu and J. H. Tsai, “USB3.1 silicon and channel design optimization using artificial neural network modeling”, in *IEEE Electromagnetic Compatibility and Signal Integrity Symp.*, Santa Clara, CA, May. 2015.

[Liu-17] Z. Liu, T. Liu, J. Han, S. Bu, X. Tang, and M. Pecht, "Signal model-based fault coding for diagnostics and prognostics of analog electronic circuits," *IEEE Trans. on Ind. Electron.*, vol. 64, no. 1, pp. 605-614, Jan. 2017.

[Ma-99] H. Ma, D. Xu, and Y-S. Lee, "Fault diagnosis of power electronic circuits based on neural network and waveform analysis," in *Proc. of the IEEE 1999 Intl. Conf. on, Power Electronics and Drive Systems (PEDS)*, Hong Kong, July 1999, pp. 234-237.

[Maas-13] A. L. Maas, A. Y. Hannun, and A. Y. Ng, "Rectifier nonlinearities improve neural network acoustic models," in *Proc. Int. Conf. Machine Learning*, Atlanta, GA, Jun. 2013, vol. 30, no. 1, p. 3.

[Mack-07] Y. Mack, T. Goel, W. Shyy, and R. Haftka, "Surrogate model-based optimization framework: A case study in aerospace design," *Stud. Comput. Intell.*, vol. 51, no. 2, pp. 323-342, 2007.

[MacKay-92] D. J. C. MacKay, "Bayesian interpolation," *Neural Comput.*, vol. 4, no 3, pp. 415–447, Mar. 1992.

[Madani-99] K. Madani, "A survey of artificial neural networks based fault detection and fault diagnosis techniques," in *Intl. Joint Conf. on Neural Networks (IJCNN)*, Washington, DC, July 1999, pp. 3442-3446.

[Mahouti-14] P. Mahouti, F. Günes, S. Demirel, A. Uluslu, and M. A. Belen, "Efficient scattering parameter modeling of a microwave transistor using generalized regression neural network," in *IEEE Int. Conf. on Microwaves, Radar, and Wireless Communication (MIKON)*, Gdansk, Poland, Jun. 2014, pp. 1-4.

[Maurya-07] M. Maurya, R. Rengaswamy, and V. Venkatasubramanian, "A signed directed graph and qualitative trend analysis-based framework for incipient fault diagnosis," *Chem. Eng. Res. Des.*, vol. 85, no. 10, pp. 1407-1422, Oct. 2007.

[McKillup-03] S. McKillup, *Statistics Explained: An Introductory Guide for Life Scientists*, Cambridge, UK: Cambridge University Press, 2006.

[Menon-14] P. Menon and C. Edwards, "Robust fault estimation using relative information in linear multi-agent networks," *IEEE Trans. Autom. Control*, vol. 59, no. 2, pp. 477–482, Feb. 2014.

[Ming-09] Y. Ming, "Application of particle swarm optimization and RBF neural network in fault diagnosis of analogue circuits," in *Int. Symp. Intelligent Information Technol. Applic.*, Shanghai, China, Nov. 2009, pp. 176-178.

[Mirafzal-14] B. Mirafzal, "Survey of fault-tolerance techniques for three-phase voltage source inverters," *IEEE Trans. Ind. Electron.*, vol. 61, no. 10, pp. 5192–5202, Oct. 2014.

[Mishra-17] P. Mishra, R. Morad, A. Ziv, and S. Ray, "Post-silicon validation in the SoC era: A tutorial introduction," *IEEE Design & Test*, vol. 34, no. 3, pp. 68-92, June 2017.

[Montgomery-14] D. C. Montgomery, *Diseño y Analisis de Experimentos*, Second Edition, New York, NY: Limusa Wiley, 2014.

[Mortazavizadeh-14] S. Mortazavizadeh and M. Mousavi, "A review on condition monitoring and diagnostic techniques of rotating electric machines," *Phys. Sci. Int. J.*, vol. 4, no. 3, pp. 310–338, May 2014.

[Moreira-10] J. Moreira and H. Werkmann, *An Engineer's Guide to Automated Testing of High-Speed Interfaces*, Norwood, MA: Artech House, 2010.

[Nagi-92] N. Nagi, J.A. Abraham, "Hierarchical fault modeling for analog and mixed-signal circuits," in *Digest of papers, VLSI Test Symposium*, Atlantic City, NJ, USA, April 1992, pp. 96-101.

[Nandi-05] S. Nandi, H. Toliyat, and X. Li, "Condition monitoring and fault diagnosis of electric motors—A review," *IEEE Trans. Energy Convers.*, vol. 20, no. 4, vol. 20, no. 4, pp. 719–725, Dec. 2005.

[Ogut-19] M. Ogut, X. Bosch-Lluis, and S. C. Reising, "A deep learning approach for microwave and millimeter-wave radiometer calibration," *IEEE Trans. Geoscience Remote Sensing*, vol. 57, no. 8, pp. 5344-5355, Aug. 2019.

[Ozgonenel-11] O. Ozgonenel and T. Yalcin, "A complete motor protection algorithm based on PCA and ANN: A real time study," *Turkish J. Elect. Eng. Comput. Sci.*, vol. 19, no. 3, pp. 317–334, May 2011.

[Panda-14] B. N. Panda, M. V. A. R. Bahubalendruni, and B. B. Biswal, "Optimization of resistance spot welding parameters using differential evolution algorithm and GRNN," in *IEEE Int. Conf. on Intelligent Systems and Control (ISCO)*, vol. 2, no. 6, Coimbatore, India, Jan. 2014, pp. 50-55.

[Park-09] S. Park, T. Hong, and S. Mitra, "Post-silicon bug localization in processors using instruction footprint recording and analysis (IFRA)," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 28, no. 10, pp. 1545-1558, Oct. 2009.

[PCISIG-16] PCI SIG Org. (2016), *Peripheral Component Interconnect Express 3.1 Specification* [Online]. Available: https://pcisig.com/specifications.

[Pencole-05] Y. Pencole and M. Cordier, "A formal framework for the decentralized diagnosis of large scale discrete event systems and its application to telecommunication network," *Artif. Intell.*, vol. 164, no. 2, pp. 121–170, May 2005.

[Pizano-Escalante-19] L. Pizano-Escalante, O. Longoria-Gandara, R. Parra-Michel, and F. Peña-Campos, "Simulation model to predict BER based on S-parameters of high-speed interconnects," *IEEE Design & Test*, vol. 36, no. 1, pp. 31-39, Feb. 2019.

[Popper-02] K. Popper, *The Logic of Scientific Discovery*, New York, NY: Routledge Classics. 2002.

[Potdar-17] K. Potdar, T. S. Pardawala, and C. D. Pai, "A comparative study of categorical variable encoding techniques for neural network classifiers," *Int. J. Computer Applications*, vol. 175, no. 4, pp. 7-9, Oct. 2017.

[Prates-16] R. Prates, M. Oliveira, and W. R. Schwartz, "Kernel partial least squares for person re-identification," in *13th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, Colorado Springs, CO, 2016, pp. 249-255.

[Pridhiviraj-15] P. Pridhiviraj, T. Dheerendra, and P. Muralidhar, "Adaptive post-silicon server validation using machine learning," *Int. J. Applied Information Systems (IJAIS)*, vol. 9, no. 1, pp. 24-32, June 2015.

[Putra-16] D. S. Putra, A. D. Wibawa, and M. H. Purnomo, "Classification of EMG during walking using principal component analysis and learning vector quantization for biometrics study," in *Intl. Seminar on Intelligent Technology and Its Applications (ISITIA)*, Lombok, Indonesia, 2016, pp. 145-150.

[Qin-14] L. Qin, X. He, and D. Zhou, “A survey of fault diagnosis for swarm systems,” *Syst. Sci. Control Eng.*, vol. 2, no. 1, pp. 13–23, Jan. 2014.

[Queipo-05] N. V. Queipo, R. T. Haftka, W. Shyy, T. Goel, R. Vaidyanathna, and P. K. Tucker, “Surrogate-based analysis and optimization,” *Prog. in Aerospace Sciences*, vol. 41, no. 1, pp. 1-28, Jan. 2005.

[Rahmani-17] K. Rahmani, S. Ray, and P. Mishra, “Postsilicon trace signal selection using machine learning techniques,” *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 25, no. 2, pp. 570-580, Aug. 2017.

[Rahme-13] S. Rahme, Y. Labit, F. Gouaisbaut, and T. Floquet, “Sliding modes for anomaly observation in TCP networks: From theory to practice,” *IEEE Trans. Control Syst. Technol.*, vol. 21, no. 3, pp. 1031–1038, May 2013.

[Rajan-98] V. Rajan, Jie Yang, S. Chakrabarty, and K. Pattipati, “Machine learning algorithms for fault diagnosis in analog circuits,” in *IEEE Int. Conf. Syst., Man, Cybernetics*, San Diego, CA, Oct. 1998, vol. 2, pp. 1874-1879.

[Rangel-Patiño-14] F. E. Rangel-Patiño, J. E. Rayas-Sánchez, and N. Hakim, “Challenges and opportunities in post-silicon electrical validation of high speed I/O’s,” Internal Report *PhDEngScITESO-14-08-R (CAECAS-14-08-R)*, ITESO, Tlaquepaque, Mexico, Oct. 2014.

[Rangel-Patiño-15] F. E. Rangel-Patiño and J. E. Rayas-Sánchez, “Towards a suitable objective function formulation for equalizer optimization for post-silicon electrical validation,” Internal Report *PhDEngScITESO-15-06-R (CAECAS-15-09-R)*, ITESO, Tlaquepaque, Mexico, Jun. 2015.

[Rangel-Patiño-16] F. Rangel-Patino, A. Viveros-Wacher, J. E. Rayas-Sánchez, E. A. Vega-Ochoa, I. Duron-Rosales, and N. Hakim, “A holistic methodology for system margining and jitter tolerance optimization in post-silicon validation,” in *IEEE MTT-S Latin America Microwave Conf. (LAMC-2016)*, Puerto Vallarta, Mexico, Dec. 2016, pp. 1-4.

[Rangel-Patiño-17a] F. E. Rangel-Patiño and J. E. Rayas-Sánchez, “Circuit modeling with artificial neural networks,” Internal Report *PhDEngScITESO-17-25-R (CAECAS-17-12-R)*, ITESO, Tlaquepaque, Mexico, July 2017.

[Rangel-Patiño-17b] F. E. Rangel-Patiño, J. L. Chávez-Hurtado, A. Viveros-Wacher, J. E. Rayas-Sánchez and N. Hakim, “System margining surrogate-based optimization in post-silicon validation,” *IEEE Trans. Microwave Theory Techn.*, vol. 65, , no. 9, pp. 3109-3115, Sep. 2017.

[Rangel-Patiño-17c] F. Rangel-Patino, J. L. Chávez-Hurtado, A. Viveros-Wacher, J. E. Rayas-Sánchez, and N. Hakim, “Eye diagram system margining surrogate-based optimization in a server silicon validation platform,” in *European Microw. Conf. (EuMC-2017)*, Nuremberg, Germany, Oct. 2017, pp. 540-543.

[Rangel-Patiño-18] F. E. Rangel-Patiño, J. E. Rayas-Sánchez, A. Viveros-Wacher, E. A. Vega-Ochoa, and N. Hakim, “High-speed links receiver optimization in post-silicon validation exploiting Broyden-based input space mapping,” in *IEEE MTT-S Int. Conf. Numer. EM Mutiphysics Modeling Opt. (NEMO-2018)*, Reykjavik, Iceland, Aug. 2018, pp. 1-3.

[Rangel-Patiño-19] F. E. Rangel-Patiño, J. E. Rayas-Sánchez, A. Viveros-Wacher, J. L. Chávez-Hurtado, E. A. Vega-Ochoa, and N. Hakim, “Post-silicon receiver equalization metamodeling by artificial neural networks," *IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems*, vol. 38, no. 4, pp. 733-740, Apr. 2019.

[Rangel-Patiño-20] F. E. Rangel-Patiño, A. Viveros-Wacher, J. E. Rayas-Sánchez, I. Durón-Rosales, E. A. Vega-Ochoa, N. Hakim, and E. López-Miralrio, "A holistic formulation for system margining and jitter tolerance optimization in industrial post-silicon validation," *IEEE Trans. Emerging Topics Computing*, vol. 8, no. 2, pp. 453-463, Apr.-Jun. 2020.

[Rayas-Sánchez-01] J. E. Rayas-Sánchez, *Neural Space Mapping Methods for Modeling and Design of Microwave Circuits*, Ph.D. Thesis, Dept. of Electrical and Comp. Eng., McMaster University, Hamilton, Canada, 2001.

[Rayas-Sánchez-04] J. E. Rayas-Sánchez, "EM-based optimization of microwave circuits using artificial neural networks: The state of the art," *IEEE Trans. Microw. Theory Techn.*, vol. 52, no. 1, pp. 420-435, Jan. 2004.

[Rayas-Sánchez-06] J. E. Rayas-Sánchez and V. Gutiérrez-Ayala, "EM-based Monte Carlo analysis and yield prediction of microwave circuits using linear-input neural-output space mapping," *IEEE Trans. Microw. Theory Techn.*, vol. 54, no. 12, pp. 4528-4537, Dec. 2006.

[Rayas-Sánchez-11] J. E. Rayas-Sánchez. (2011). *An Introduction to Space Mapping*. Lecture of the graduate course "Optimization-Based Modeling and Design of Electronic Circuits". Dept. of Electronics, Systems and Informatics, ITESO – The Jesuit U. of Guadalajara, Tlaquepaque, Mexico, 2011. Available: https://desi.iteso.mx/erayas/documents/CirOpt_course/Lectures/Space_Mapping/SM_intro.pdf.

[Rayas-Sánchez-16] J. E. Rayas-Sánchez, "Power in simplicity with ASM: tracing the aggressive space mapping algorithm over two decades of development and engineering applications," *IEEE Microwave Magazine*, vol. 17, no. 4, pp. 64-76, Apr. 2016.

[Roth-66] J. P. Roth, "Diagnosis of automata failures: a calculus and a method," *IBM Journal of Research and Development*, vol. 10, no. 4, pp. 278-291, Jul. 1966.

[SATAOrg-16] SATA Org. (2016), *Serial Advanced Technology Attachment 3.2 Specification* [Online]. Available: http://www.sata-io.org/

[Sebastian-19] A. Sebastian *et al.*, "Computational memory-based inference and training of deep neural networks," in *2019 Symp. VLSI Technology*, Kyoto, Japan, Jun. 2019, pp. T168-T169.

[Sela-16] M. Sela and R. Kimmel, "Randomized independent component analysis," in *IEEE International Conference on the Science of Electrical Engineering (ICSEE)*, Eilat, Israel, 2016, pp. 1-5.

[Seshadrinath-14] J. Seshadrinath, B. Singh, and B. Panigrahi, "Vibration analysis based interturn fault diagnosis in induction machines," *IEEE Trans. Ind. Informat.*, vol. 10, no. 1, pp. 340–350, Feb. 2014.

[Sheibat-Othman-14] N. Sheibat-Othman, N. Laouti, J. Valour, and S. Othman, "Support vector machines combined to observers for fault diagnosis in chemical reactors," *Can. J. Chem. Eng.*, vol. 92, pp. 685–694, Apr. 2014.

[Sobol-67] I. M. Sobol, "On the distribution of points in a cube and the approximate evaluation of integrals," *U.S.S.R. Computational Mathematics and Mathematical Physics*, vol. 7, no. 1, pp. 86-112, Jan. 1967.

[Song-13] Y. Song and B. Wang, “Survey of reliability of power electronic systems,” *IEEE Trans. Power Electron.*, vol. 28, no. 1, pp. 591–604, Jan. 2013.

[Sorin-09] D. Sorin, *Fault Tolerant Computer Architecture*, Vermont, USA: Morgan & Claypool Publishers, 2009.

[Sottile-94] J. Sottile Jr. and L. Holloway, “An overview of fault monitoring and diagnosis in mining equipment,” *IEEE Trans. Ind. Appl.*, vol. 30, no. 5, pp. 1326–1332, Oct. 1994.

[Specht-91] D. F. Specht, “A general regression neural network,” *IEEE Trans. on Neural Networks*, vol. 2, no. 6, pp. 568-576, Nov. 1991.

[Srivastava-14] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: a simple way to prevent neural networks from overfitting,” *The Journal of Machine Learning Research*, vol. 15, no. 1, pp.1929-1958, Jun. 2014.

[Stanley-87] W. D. Stanley, *Operational Amplifiers with Linear Integrated Circuits*. Columbus, OH: Merrill, 1989, pp. 373-381.

[Stengel-91] R. Stengel, “Intelligent failure-tolerant control,” *IEEE Control Syst. Mag.*, vol. 11, no. 4, pp. 14–23, Jun. 1991.

[Stephens-04] R. Stephens, “Jitter analysis: the dual-Dirac model, rj/dj, and q-scale,” Agilent Technologies, Santa Clara, CA, White Paper, Dec. 2004.

[Strunz-16] B. Strunz, C. Flanagan, *Design for Testability in Digital Integrated Circuits*, Tim Hall University of Limerick, Ireland [Online]. Available: http://www.cs.colostate.edu/~cs530/digital_testing.pdf

[Swingler-14] J. Swingler, *Reliability Characterisation of Electrical and Electronic Systems*, Cambridge, UK: Woodhead Publishing, 2014.

[Talebi-07] H. A. Talebi and K. Khorasani, “A neural network-based actuator gain fault detection and isolation strategy for nonlinear systems,” in *46th IEEE Conference on Decision and Control*, New Orleans, LA, Dec. 2007, pp. 2614-2619.

[Tang-09] J. Tang, Y. Shi and D. Jiang, “Analog circuit fault diagnosis with hybrid PSO-SVM,” in *IEEE Circuits and Systems Int. Conf. Testing and Diagnosis*, Chengdu, China, Apr. 2009, pp. 1-5.

[Tankovska-20] H. Tankovska (2020). *Internet of Things – Active Connections Worldwide 2015-2025* [Online]. Available: https://www.statista.com/statistics/1101442/iot-number-of-connected-devices-worldwide/

[Telgarsky-16] M. Telgarsky, “Benefits of depth in neural networks,” in *Proc. Conf. Learning Theory*, New York, NY, USA, Jun. 2016, pp. 1-23.

[Thakkar-18] V. Thakkar, S. Tewary, and C. Chakraborty, “Batch normalization in convolutional neural networks — a comparative study with CIFAR-10 data,” in *Fifth Int. Conf. Emerging Applications of Information Technology (EAIT)*, Kolkata, India, Jan. 2018, pp. 1-5.

[USBOrg-16a] USB Org. (2016). *Universal Serial Bus Revision 3.1 Specification* [Online]. Available: http://www.usb.org/developers/doc

[USBOrg-16b] USB Org. (2016). *USB3.0 Electrical Compliance Methodology White Paper* [Online]. Available: http://www.usb.org/developers/doc

[VanBeers-05] W. C. M. Van Beers, “Kriging metamodeling in discrete-event simulation: An overview,” in *Proc. Simulation Conf.*, Dec. 2005, pp. 202-208.

[Venkatasubramanian-03] V. Venkatasubramanian, R. Rengaswamy, S. Kavuri, and K. Yin “A review of process fault detection and diagnosis - Part III: Process history based methods,” *Comput. Chem. Eng.*, vol. 27, no. 3, pp. 313–326, Mar. 2003.

[Vicario-16] G. Vicario, G. Craparotta, and G. Pistone, “Meta-models in computer experiments: Kriging versus artificial neural network,” *Qual. Reliab. Engng. Int.*, vol. 32, no. 6, pp. 2055-2065, June 2016.

[Viveros-Wacher-14] A. Viveros-Wacher, R. Alejos, L. Alvarez, I. Diaz-Castro, B. Marcial, G. Motola-Acuna, and E. A. Vega-Ochoa, “SMV methodology enhancements for high speed IO links of SoCs,” in *IEEE VLSI Test Symp. (VTS-2014)*, Napa, CA, May. 2014, pp. 1-5.

[Viveros-Wacher-16] A. Viveros-Wacher and J. E. Rayas-Sánchez, “Eye diagram optimization based on design of experiments (DoE) to accelerate industrial testing of high speed links,” in *2016 IEEE MTT-S Latin America Microwave Conference (LAMC)*, Puerto Vallarta, Dec. 2016, pp. 1-3.

[Viveros-Wacher-18a] A. Viveros-Wacher, R. Baca-Baylón, F. E. Rangel-Patiño, M. A. Dávalos-Santana, E. A. Vega-Ochoa, and J. E. Rayas-Sánchez, “Jitter tolerance acceleration using the golden section optimization technique,” in *IEEE Latin American Symp. Circuits and Systems Dig. (LASCAS 2018)*, Puerto Vallarta, Mexico, Feb. 2018, pp. 1-4.

[Viveros-Wacher-18b] A. Viveros-Wacher and J. E. Rayas-Sánchez, “Analog fault identification in RF circuits using artificial neural networks and constrained parameter extraction,” in *IEEE MTT-S Int. Conf. Num. EM Mutiphysics Modeling Opt. (NEMO-2018)*, Reykjavik, Iceland, Aug. 2018, pp. 1-3.

[Viveros-Wacher-19] A. Viveros-Wacher, J. E. Rayas-Sánchez, and Z. Brito-Brito, “Analog gross fault identification in RF circuits using neural models and constrained parameter extraction,” *IEEE Trans. Microwave Theory Techn.*, vol. 67, no. 6, pp. 2143-2150, Jun. 2019.

[Viveros-Wacher-21] A. Viveros-Wacher, A. Norman, and J. E. Rayas-Sánchez, “Deep neural modeling of BER extrapolation in HSIO links,” *IEEE Trans. Electromagnetic Compatibility*, 2021 (work submitted).

[Wang-12] Z. Wang, “Fault diagnosis method based on fuzzy support vector machines and self-organizing map neural network,” *Int. J. Adv. Comput. Technol.*, vol. 4, no. 9, pp. 139–147, Oct. 2012.

[Wang-15] X. Wang and Q. Hu, “Analysis and optimization of combined equalizer for high speed serial link,” in *IEEE 9th Int. Conf. Anti-counterfeiting, Security, and Identification (ASID)*, Xiamen, China, Sept. 2015, pp. 43-46.

[Wang-16] F. Wang, P. Cachecho, W. Zhang, S. Sun, X. Li, R. Kanj, and C. Gu, “Bayesian model fusion: large-scale performance modeling of analog and mixed-signal circuits by reusing early-stage data,” *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 35, no. 8, pp. 1255-1268, Aug. 2016.

[Widodo-07] A. Widodo and B. Yang, “Support vector machine in machine condition monitoring and fault diagnosis,” *Mech. Syst. Signal Process.*, vol. 21, no. 6, pp. 2560–2574, Aug. 2007.

[Willsky-76] A. Willsky, “A survey of design methods for failure detection in dynamic systems,” *Automatica*, vol. 12, no. 6, pp. 601–611, Nov. 1976.

BIBLIOGRAPHY

[Wu-00] C. F. J. Wu and M. Hamada, *Experiments: Planning, Analysis, and Parameter Design Optimization*, New York, NY: Wiley, 2000.

[Xia-06] L. Xia, J. Meng, R. Xu, B. Yan, and Y. Guo, "Modeling of 3-D vertical interconnect using support vector machine regression," *IEEE Microw. Wireless Compon. Lett.*, vol. 16, no. 12, pp. 639-641, Dec. 2006.

[Xu-16] X. Xu, N. Lu, J. Yong, and B. Jiang, "Fault propagation analysis of IGBT fault in CRH5 traction system based on signed directed graph," in *Prognostics and System Health Management Conference (PHM-Chengdu)*, Chengdu, China, Oct. 2016, pp. 1-6.

[Xue-11] Z. Xue, Y. Li, and Y. Cao, "Simulation of improved BP algorithm in the fault diagnosis of analog circuit," in *Int. Conf. Electrical and Control Eng.*, Yichang, China, Sept. 2011, pp. 3148-3150.

[Yang-00] Z. R. Yang, M. Zwolinski, C. D. Chalk, and A. C. Williams, "Applying a robust heteroscedastic probabilistic neural network to analog fault detection and classification," *IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems*, vol. 19, no. 1, pp. 142-151, Jan. 2000.

[Yang-08] S. Yang, W. Li, and C. Wang, "The intelligent fault diagnosis of wind turbine gearbox based on artificial neural network," in *Intl. Conf. on Condition Monitoring and Diagnosis*, Beijing, Apr. 2008, pp. 1327-1330.

[Yau-71] S. S. Yau and Y. S. Tang, "An efficient algorithm for generating complete test sets for combinational logic circuits," *IEEE Transactions on Computers*, vol. C-20, no. 11, pp. 1245-1251, Nov. 1971.

[Yelten-12] M. B. Yelten, T. Zhu, S. Koziel, P. D. Franzon, and M. B. Steer, "Demystifying surrogate modeling for circuits and systems," *IEEE Circuits Systems Magazine*, vol. 12, pp. 45-63, First Quarter 2012.

[Yelten-13] M. B. Yelten, S. Natarajan, B. Xue, and P. Goteti, "Scalable and efficient analog parametric fault identification," in *IEEE/ACM Int. Conf. on Computer-Aided Design (ICCAD)*, San Jose, CA, Nov. 2013, pp. 387-392.

[Yigit-17] B. Yigit, G. Li Zhang, B. Li, Y. Shi, and U. Schlichtmann, "Application of machine learning methods in post-silicon yield improvement," in *30th IEEE International System-on-Chip Conference (SOCC)*, Munich, Sept. 2017, pp. 243-248.

[Yin-14] S. Yin, X. Gao, H. Karimi, and X. Zhu, "Study on support vector machine-based fault detection in Tennessee Eastman process," *Abstract Appl. Anal.*, vol. 2014, 2014.

[Ying-00] D. Ying and H. Yigang, "On the application of artificial neural networks to fault diagnosis in analog circuits with tolerances," in *Proc. Int. Conf. Signal Processing World Computer Cong.*, Beijing, China, Aug. 2000, pp. 1639-1642 vol. 3.

[Yuan-06] H. Yuan, G. Chen, S. Shi, and H. Chen, "Research on fault diagnosis in analog circuit based on wavelet-neural network," in *World Cong. Intelligent Control Automation*, Dalian, China, Jun. 2006, pp. 2659-2662.

[Zhang-00] Q. J. Zhang and K. C. Gupta, *Neural Networks for RF and Microwave Design*. Norwood, MA: Artech House, 2000.

[Zhang-12] Z. Zhang, Z. Jiang, X. Meng, S. Cheng, and W. Sun, “Research on prediction method of API based on the enhanced moving average method,” in *Int. Conf. Systems and Informatics (ICSAI2012)*, Yantai, China, Jun. 2012, pp. 2388-2392.

[Zhang-15] H. Zhang, S. Krooswyk, and J. Ou, *High Speed Digital Design, Design of High Speed Interconnects and Signaling*, California, USA: Morgan Kaufmann, 2015.

[Zhang-16] B. Zhang, R. Xu, X. Yin, and Z. Gao, “Research on fault diagnosis for rail vehicle compartment of LED lighting system of analog circuit based on WP-EE and BP neural network,” in *Chinese Control Decision Conf. (CCDC)*, Yinchuan, China, May 2016, pp. 2989-2993.

[Zhao-05] F. Zhao, K. Koutsoukos, H. Haussecker, J. Reich, and P. Cheung, “Monitoring and fault diagnosis of hybrid systems,” *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 35, no. 6, pp. 1225–1240, Dec. 2005.

[Zidani-08] F. Zidani, D. Diallo, M. Benbouzid, and R. Nait-Said, “A fuzzy-based approach for the diagnosis of fault modes in a voltage-fed PWM inverter induction motor drive,” *IEEE Trans. Ind. Electron.*, vol. 55, no. 2, pp. 586–593, Feb. 2008.

[10GEA-16] 10GEA Org. (2016), *XAUI interface* [Online]. Available: http://www.10gea.org/whitepapers/xaui-interface/.

Author Index

Subject Index