

© 2024 Bobi Shi

EYE DIAGRAM MODELING AND STATISTICAL SIMULATION IN
NONLINEAR HIGH-SPEED LINK SYSTEMS

BY

BOBI SHI

DISSERTATION

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Electrical and Computer Engineering
in the Graduate College of the
University of Illinois Urbana-Champaign, 2024

Urbana, Illinois

Doctoral Committee:

Professor José E. Schutt-Ainé, Chair
Professor Jennifer Bernhard
Associate Professor Peter D Dragic
Professor Pavan Kumar Hanumolu

ABSTRACT

The exponential growth in data rates of high-speed link channels, coupled with the emergence of severe nonlinear phenomena, highlights the urgent requirement for efficient and accurate simulation of high-speed link systems. A commonly utilized method for assessing signal integrity in these channels is through eye diagram analysis, facilitating the identification of jitter and noise by engineers. However, conventional approaches to conducting such analysis often prove resource-intensive regarding both time and memory or are limited to linear time-invariant systems.

This thesis introduces two innovative methodologies designed to accurately and efficiently derive the eye diagram within nonlinear systems. The first approach uses a high-speed link surrogate model based on the polynomial chaos expansion method. This model enables rapid prediction of both the eye diagram and the full waveform, outperforming traditional transient simulations in terms of speed. The second approach uses an enhanced statistical analysis tailored for nonlinear systems, utilizing the Wiener model and random variable transformation. This methodology yields a three-dimensional statistical eye diagram, offering deeper insights into system behavior.

Numerical examples presented in this study demonstrate the efficacy and accuracy of the proposed methodologies. By overcoming the limitations of traditional approaches, these methods pave the way for more effective analysis and design of high-speed link systems in the face of nonlinear phenomena.

To my parents, for their love and support.

ACKNOWLEDGMENTS

The completion of this dissertation marks the culmination of my academic journey as a student. When I first arrived at the University of Illinois Urbana-Champaign Electrical and Computer Engineering Department as an undergraduate student in 2013, earning a doctoral degree was not even on my radar. However, through dedication and perseverance, I have traversed the path from being a freshman newcomer to proudly earning the title of Dr. Shi.

I would like to extend my heartfelt gratitude to my advisor, Professor José E. Schutt-Ainé, for his exceptional support and guidance throughout my doctoral journey. He has been an invaluable mentor, providing invaluable insights, encouragement, and support at every stage of my research. His greatest asset when working with students is his flexibility, allowing them the freedom to explore all possibilities without constraint and harsh deadlines.

I extend my heartfelt appreciation to the members of my thesis committee, Professor Jennifer Bernhard, Professor Peter D Dragic, and Professor Pavan Kumar Hanumolu, for their insightful feedback, constructive criticism, and scholarly contributions that have enriched the quality of this work. Thanks for their precious time dedicated to my work.

I am indebted to the ECE department for providing the resources, facilities, and conducive environment essential for conducting this research. I would like to express my gratitude to the Electromagnetic Laboratory and the Center for Computational Electromagnetic staff and faculty members, especially Karen Kuhns, on the fifth floor of the ECE building, who have assisted me during various stages of my academic pursuit.

Next, I would like to express my sincere appreciation to Dr. Thong Nguyen in our research group. His vast knowledge and guidance have been invaluable to me. I can not count the number of times he patiently helped me navigate research hurdles during our lengthy Zoom calls in the COVID-19 period. I

am profoundly inspired by his innovative perspectives and his ability to approach problems from various angles, which has enriched my understanding of the subject matter. In addition, I would like to thank Dr. Nancy Zhao who introduced me to the research group as my ECE 451 TA. Without her introduction and encouragement, I might not choose to pursue a doctoral degree. Furthermore, I also would like to thank our group members: Professor Hanzhi Ma, Dr. Xiao Ma, Dr. Xinying Wang, Ph.D. candidate Juhitha Konduru, Ph.D. candidate Yi Zhou, Haofeng Wang, Halim Park, Zohreh Salehi, Tahsin Shameem, Qiuyu Wang, and many others.

In addition, I want to express my appreciation to my dearest friends outside the academic sphere, who have brought so much joy and laughter to my life: the culinary maestro Jiahao Wen, the epitome of charisma Runlai Zeng, the adventurous Web3 entrepreneur Siyun He, the brilliant scientist Dr. Lianghao Cao, and the unparalleled poet extraordinaire Professor Shangyang Fang. You’ve made my social life richer and more vibrant, and I’m incredibly lucky to have you all by my side.

Furthermore, I want to give a big shout-out to Xiaofan Wu, my partner in crime since 2015. Her unwavering support and love have been my rock throughout this academic journey. As she embarks on her own doctoral adventure, I couldn’t be prouder or more excited for her. Here’s to us, taking on the world together, one degree at a time!

Finally, I reserve the most heartfelt gratitude for my parents, Ying Hu and Zhongci Shi. Their unconditional love and support have been my guiding light throughout this journey, especially considering the immense sacrifices they made both financially and emotionally to provide me with overseas education opportunities when I was just 16 years old. My Chinese name, “博 (bó)”, bestowed upon me by them, signifies erudition and versatility, akin to a person holding a doctoral degree, while “(比 bǐ)” symbolizes competition. With their blessings, I have fulfilled their initial aspirations, and now, the journey continues, marked by the spirit of competition.

TABLE OF CONTENTS

LIST OF TABLES	viii
LIST OF FIGURES	ix
CHAPTER 1 INTRODUCTION	1
1.1 Problem Statement	1
1.2 Review of Circuit Modeling and Simulation	3
1.3 Overview of Eye Diagram	7
CHAPTER 2 HIGH-SPEED LINK MODELING WITH POLY- NOMIAL CHAOS METHOD	16
2.1 Introduction	16
2.2 Polynomial Chaos Theory	17
2.3 Eye diagram Estimation via Polynomial Chaos Model	20
2.4 Decision-Feedback Equalizer (DFE) Taps Estimation with Surrogate Modeling Methods [25]	26
2.5 Implementation in Microwave Circuit	33
CHAPTER 3 HIGH-SPEED LINK METHODS IN LINEAR TIME- INVARIANT SYSTEM	38
3.1 Introduction	38
3.2 Peak Distortion Analysis	39
3.3 Double-Edge-Based Bit-by-Bit Approach	42
3.4 Statistical Analysis in the Linear Time-invariant System	49
CHAPTER 4 STATISTICAL HIGH-SPEED LINK METHOD IN NONLINEAR SYSTEM	57
4.1 Introduction	57
4.2 Nonlinearity Modeling	59
4.3 Validation of the Proposed Method	65
4.4 Conclusion	81
CHAPTER 5 CONCLUSION AND FUTURE WORK	83
5.1 Conclusion	83
5.2 Limitations and Future Work	85

REFERENCES	87
----------------------	----

LIST OF TABLES

2.1	Eye diagram comparison	24
4.1	Comparison between the transient and nonlinear statistical simulation methods in numerical Wiener model example. .	71
4.2	Comparison between the transient and nonlinear statistical simulation methods in the differential FinFET buffer example.	73
4.3	Comparison between the transient and nonlinear statistical simulation methods in the high-speed link example. . . .	81
4.4	Runtime comparison between the transient and nonlinear statistical simulation methods in the high-speed link example.	81

LIST OF FIGURES

1.1	Flowchart of a transient SPICE simulation [24].	6
1.2	Ideal eye diagram in high-speed link signaling.	8
1.3	Typical eye diagram in high-speed link signaling.	9
1.4	A 3D eye diagram visualization.	11
1.5	Voltage and time PDF from eye diagram.	11
1.6	Gaussian distribution.	12
1.7	Amplitude definitions of eye diagram.	13
1.8	Time definitions of eye diagram.	15
2.1	The surrogate model black-box.	16
2.2	The orthonormal Hermite polynomials.	17
2.3	Decomposition of single bit sequence.	22
2.4	Four surrogate models.	23
2.5	Superposition to eye diagram.	23
2.6	Eye diagram from PCE.	24
2.7	Eye diagram from transient simulation.	25
2.8	Waveform comparison between ADS simulation and PC.	25
2.9	Intersymbol interference effect.	26
2.10	Post response variation due to various tap combinations.	30
2.11	DFE tap 1.	31
2.12	DFE tap 2.	31
2.13	DFE tap 3.	32
2.14	Filter insertion loss variations [12].	36
2.15	Center frequency as output [12].	37
2.16	Bandwidth as output [12].	37
2.17	Shape factor as output [12].	37
3.1	Single bit response.	40
3.2	The worst eye-opening.	41
3.3	Rising edge response with notations.	43
3.4	Rising and falling edge responses.	43
3.5	Rising edge response at 1 ns and falling edge response at 2 ns.	45
3.6	The channel response of bit sequence “0100”.	46
3.7	The channel response of bit sequence “101001”.	47
3.8	Transient eye diagram for 5 Gbps 200-bit data sequence.	48

3.9	Bit-by-bit eye diagram for 5 Gbps 200-bit data sequence. . . .	48
3.10	Single bit response with 1 pre-cursor and 2 post-cursors. . . .	50
3.11	Individual PDF for 4 cursors.	51
3.12	Convolution resultant PDF.	52
3.13	ISI combination at t_0	53
3.14	ISI combination at t_1	53
3.15	ISI combination from t_0 to t_{20}	54
3.16	Eye diagram from 1 pre-cursor and 2 post-cursors.	54
3.17	Eye diagram from 2 pre-cursors and 5 post-cursors.	55
3.18	Eye diagram from 3 pre-cursors and 7 post-cursors.	55
3.19	Eye height and eye width convergence.	56
4.1	The different block-oriented structures, connected by LTI blocks $G(q)$ and $S(q)$ and static nonlinear blocks $f(\cdot)$ and $g(\cdot)$	60
4.2	The Wiener model, where the input $u(t)$ and the output $y(t)$ are measurable, but not the intermediate signal $x(t)$	62
4.3	The flowchart of the proposed method through the random variable transformation.	63
4.4	Nonlinear transformation of a standard normal random variable. . . .	64
4.5	The mathematical model of a Wiener system.	65
4.6	Receiver nonlinearity.	66
4.7	The single-ended signal eye diagrams through the direct statistical analysis.	67
4.8	The comparison between the transient simulation and pro- posed nonlinear statistical analysis in single-ended signal. . . .	68
4.9	The differential signal eye diagrams through the direct sta- tistical analysis.	69
4.10	The comparison between the transient simulation and pro- posed nonlinear statistical analysis in the differential signal. . .	70
4.11	The schematic of differential FinFET buffer system.	72
4.12	2.5D package model with two chiplets.	72
4.13	The comparison between the transient simulation and pro- posed nonlinear statistical analysis in 32 Gbps differential FinFET buffer.	74
4.14	The comparison between the transient simulation and pro- posed nonlinear statistical analysis in 64 Gbps differential FinFET buffer.	75
4.15	The schematic of the high-speed link system with Rx CTLE and DFE.	76
4.16	The comparison between the transient simulation and pro- posed nonlinear statistical analysis of the eye diagram at V_{CH}	78

4.17	The comparison between the transient simulation and proposed nonlinear statistical analysis of the eye diagram at V_{CTLE}	79
4.18	The comparison between the transient simulation and proposed nonlinear statistical analysis of the eye diagram at V_{OUT}	80
5.1	The illustration of reversing the statistical eye diagram to an artificial pulse response.	86

CHAPTER 1

INTRODUCTION

1.1 Problem Statement

Six decades ago, Gordon E. Moore, the co-founder of Intel, made a seminal observation in the semiconductor industry, noting that the number of transistors on a microchip roughly doubles every two years. This observation, famously known as Moore's Law, has since become a guiding principle for research and development teams in the industry. The world has reaped the benefits of this trend, witnessing the integration of smaller transistors into integrated circuits and the consequent increase in data transfer speeds. Today, microchips featuring a 4 nm process boast billions of transistors, with the industry setting its sights on even smaller nodes like 3 nm, 2 nm, or even 1 nm in the near future. Similarly, high-speed interconnect technologies such as Peripheral Component Interconnect Express (PCIe), Universal Serial Bus (USB), and High-Definition Multimedia Interface (HDMI) have seen significant advancements, with speeds now ten times faster than a decade ago. However, while shrinking transistor sizes and boosting data rates enhance performance, they also present considerable challenges in signal integrity and power integrity during the design process. Issues such as skin effect, crosstalk, dielectric loss, and intersymbol interference can lead to signal errors. Aside from the high-frequency effects, the nonlinearity-related issue becomes increasingly severe in the high-speed link [1–3]. To address these complexities and ensure successful designs before mass fabrication, reliable modeling techniques and simulation tools are essential. To fully capture the nonlinear aspects of the system, one attempt is to utilize a conventional Simulation Program with Integrated Circuit Emphasis (SPICE)-based simulation tool [4]. This simulator accurately models the physical properties and electrical characteristics of circuit elements, computing voltage and current

values at each time point precisely. However, the simulation runtime escalates in proportion to the transition period, leading to convergence challenges when nonlinearity is present. Consequently, this simulation method is not suitable for high-speed link systems with weakly nonlinear transmitters (TX) or receivers (RX) operating over long sequences of bits.

To overcome the runtime challenge, a surrogate model is introduced to help reduce simulation runtime by providing a faster alternative for evaluating the system's response. Rather than performing detailed simulations for each time point, which involves solving complex equations or running computationally intensive simulations, the surrogate model can quickly provide approximate results based on its trained knowledge of the system's behavior. In [5–9], an Artificial Neural Network (ANN) is applied in high-speed link systems to estimate the equalized output waveform and the eye diagram opening. The Support Vector Machine (SVM) is used to estimate the eye height and optimize the channel design in the high-speed link system in [8, 10, 11]. Gaussian Process (GP) regression [12] and Polynomial Chaos (PC) [13, 14] expansion are also widely developed as surrogate models in designing filters in microwave systems and estimating jitters in high-speed link systems. In this dissertation, the stability and scalability of single dimension to multiple dimensions of the PC surrogate model provide the most advantage for approximating the output waveform in the time domain given the bit sequence. The eye diagram is then recreated from the waveform to evaluate the signal integrity of the system.

In order to address the nonlinearity issue while maintaining efficiency, the statistical analysis developed for nonlinear systems is introduced. The current methods assuming linearity and using the superposition concept in high-speed signaling systems underestimate the impact of nonlinear effects in circuits. Therefore, the current conventional fast simulation method is no longer valid for analyzing high-speed signaling systems with nonlinear components. To properly predict eye diagrams with nonlinear components in a fast and efficient path, there are several approaches. One approach uses the Volterra series to decompose a nonlinear network into multiple linear systems and then applies the peak distortion analysis to estimate the worst-case eye margin [15]. Next, the nonlinear network is represented by the Hammerstein or Wiener models and then the eye diagram is analyzed through the bit sequence transient simulation and the voltage distribution function within the

nonlinear system can be approximated [16]. In [17], a matrix is created from all combinations of nonlinear responses and a low-rank matrix representation is found to reduce the number of nonlinear responses needed to complete the eye diagram. The study [18] proposes multiple edge responses beyond the single bit response, following the same methodology in [19] to linearize the nonlinear system and perform the superposition to calculate the statistical eye diagram.

In a brief overview of current methods, either the lengthy bit sequence transient simulation is time-demanding or the majority of fast and efficient methods for analyzing eye diagrams are designed for LTI systems with non-linearity trade-off. To tackle these limitations, the surrogate model of the high-speed link through the Polynomial Chaos Expansion method is introduced to accelerate the simulation and the statistical analysis method in the eye diagram to properly capture nonlinear features.

The dissertation is organized as follows: Chapter 1 describes the motivation and provides an overview of the circuit modeling and simulation and the fundamental characteristics of eye diagrams. In Chapter 2, the surrogate model is first introduced. The PC expansion and its training process are reviewed with the example of eye diagram estimation through PC. Other applications with the PC surrogate model are also presented. In Chapter 3, the peak distortion analysis, edge-based bit-by-bit analysis, and statistical analysis of eye diagram estimation for the linear systems are discussed. Chapter 4 investigates the statistical analysis of nonlinear systems in three examples. The Volterra-Wiener system identification technique is reviewed and the random variable transformation is applied to the transfer of the PDF between nonlinear input and output. The nonlinear statistical eye diagram is obtained. At the end of Chapter 5, the conclusion and future work is given.

1.2 Review of Circuit Modeling and Simulation

A brief review of the circuit modeling methods and simulation methods is introduced in this section to readers as fundamental knowledge. Circuit simulation involves the creation and analysis of an electronic circuit model using sophisticated software algorithms. These algorithms are used to predict and validate the behavior and performance of the circuit. Given the considerable

expense and time associated with fabricating electronic circuits, particularly integrated circuits (ICs), it proves both faster and more cost-effective to verify circuit behavior and performance using a circuit simulator prior to fabrication.

1.2.1 Circuit Modeling

The initial process begins with circuit modeling, a fundamental aspect of the simulation process, which plays a crucial role in accurately representing the behavior of real-world circuits. It is categorized into two types: one is an exact voltage-current modeling of the components by their physical performance; another one is called behavioral modeling which accurately approximates the voltage-current relation in the time or frequency domain.

In exact modeling, modeling means the accurate representation of every essential component such as resistors, capacitors, inductors, and transistors in circuits, taking into account their electrical characteristics and physical properties. The most famous Metal–Oxide–Semiconductor Field-Effect Transistor (MOSFET) transistor model for integrated circuit design is the BSIM (Berkeley Short-channel IGFET Model) Group, located in the Department of Electrical Engineering and Computer Sciences (EECS) at the University of California, Berkeley, develops physics-based, accurate, scalable, robust, and predictive MOSFET SPICE models for circuit simulation and Complementary Metal-Oxide-Semiconductor (CMOS) technology development [20]. The circuit’s nonlinear behavior is depicted within the model. With the continuous shrinking of device sizes in each process generation, as stated in Moore’s law, novel models are required to precisely capture the behavior of transistors and the difficulty arises.

Behavioral modeling refers to the process of creating mathematical representations that accurately depict the electrical characteristics and responses of electronic circuits under various conditions. For an LTI system, the scattering parameters (S-parameters) [21] are commonly used to present how electrical signals propagate, reflect, or are absorbed by each component within the high-frequency microwave network. When nonlinearity exists in the system, X-parameters [22] characterize the nonlinear behavior of active components such as amplifiers and mixers and represent the relationship between the

input and output signals of a nonlinear device in terms of power, rather than voltage or current as in S-parameters. The more general standard one is the Input/Output Buffer Information Specification (IBIS) model, which is a standard file format that stores the I/O voltage and current relationship in different conditions [23]. These models may include parameters such as component values, voltage-current relationships, frequency response, and nonlinear effects. The advantage of behavioral modeling is its black-box intellectual property (IP) protection nature; however, the vendor's effect of approximating the circuit behavior dramatically increases as the device size and transmission speed push to the limit.

1.2.2 Circuit Simulation

Once the circuit model is constructed, it undergoes thorough analysis using simulation software. This software uses mathematical techniques to solve the intricate equations that dictate how the circuit behaves. It calculates important aspects like voltage levels, current flows, and signal propagation. By adjusting input conditions and component values systematically, the simulator can evaluate how the circuit reacts in various situations. There are various types of circuit simulators available to meet a range of needs, balancing accuracy, performance, and capacity. Analog simulators provide high accuracy and are suitable for small circuits, while digital simulators, described using hardware description languages (HDL), offer greater performance and capacity for a large-scale circuit but with lower accuracy.

For analog simulation, SPICE is the original simulator that analyzes the behavior of the circuit accurately with linear and nonlinear device models. It uses various integration methods like Forward Euler, Backward Euler, and Newton-Raphson, along with matrix decomposition techniques, to calculate the voltage and current at every time point throughout the simulation period. However, multiple iterations of Newton-Raphson are required to achieve convergence at every time point for a nonlinear system in a transient simulation, as shown in Figure 1.1. The computational-expensive feature comes with this type of simulation.

Unlike analog simulation, which deals with continuous signals, digital simulation works with discrete voltage levels, mainly logic 0 and logic 1. Digital

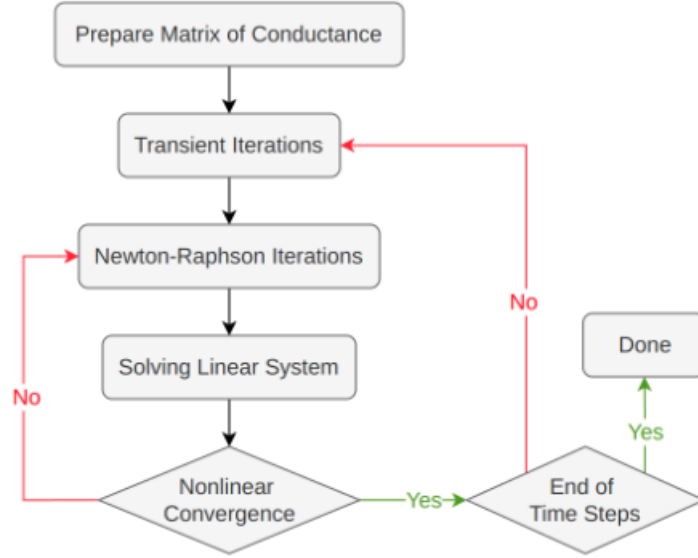


Figure 1.1: Flowchart of a transient SPICE simulation [24].

circuit simulation uses simpler models of the electronic circuit, often built in HDL such as Verilog, SystemVerilog, and SystemC. The accuracy of how these signals propagate through the circuit varies, especially in terms of the delay. This approach enables larger circuits to be simulated more quickly and with fewer computing resources compared to analog simulation but with lower accuracy.

1.2.3 Advanced Techniques on Modeling and Simulation

In recent decades, machine learning techniques have become pervasive across various fields of study, including circuit modeling and simulation. In order to alleviate the complexity of modeling and simulation, numerous machine learning and surrogate modeling approaches have emerged to capture the input-output relationships, particularly the nonlinear relationship. Researchers have actively pursued studies aimed at replacing traditional SPICE simulation tools with faster machine learning models in high-speed link systems [5–12]. Through training and testing datasets, machine learning-based behavioral models, such as Neural Networks (NN), can be developed, significantly accelerating the simulation process compared to conventional SPICE simulation methods. However, a notable drawback of machine learning is the requirement for substantial training data to ensure comprehensive model

definition without overlooking crucial details. Another more trivial method is surrogate modeling of the system, which will be discussed in Chapter 2 in detail. This includes methods such as linear regression, polynomial regression, PC expansion, and Gaussian Process regression, which have been applied in the modeling and simulation of high-speed systems [25, 26].

1.3 Overview of Eye Diagram

With the accurate representation of signal behavior from a robust circuit simulation tool, the eye pattern or the eye diagram is the fundamental visualization methodology to represent the quality of signaling in a system. As the name suggests, the eye diagram constructs the “eye” from the high-speed signal waveform by creating a 2-bit width window and overlapping each window on top of each other with 1-bit shifting. Another way of speaking is the eye diagram is the superimpose of all 0’s, 1’s, and transitions of a long pattern waveform into a signal graph representation. The resultant graph representation will resemble an eye and display the average statistics of the signal. An ideal eye diagram is constructed and shown below in Figure 1.2, where the ideal digital level 0 and level 1 signals are generated with fast rising and falling time. Since the eye diagram usually displays a 2-bit window, the horizontal axis represents the time in 2 unit-interval (UI). One UI is 1-bit width. The vertical axis represents the signal amplitude in voltage. The real-world eye diagram experiences signal impairments such as skin effect and amplitude and time distortion such as jitter, noise, crosstalk, etc. Figure 1.3 shows the eye diagram from a 1 GHz signal transmitting in a high-speed link channel. Visually comparing Figure 1.2 and Figure 1.3, the eye in Figure 1.3 is smaller than the eye in Figure 1.2, therefore the more likelihood the sampler in the high-speed link receiver will incorrectly sample a logical 1 bit for a logical 0 bit or vice versa. Such incorrect bit identifications lead to bit errors. The ratio of bit errors to overall sent bits is Bit Error Rate (BER). High-speed link designers aim to achieve the smallest BER possible when designing a transmission system. In the current design standards, a BER of less than 10^{-12} is considered the minimum target.

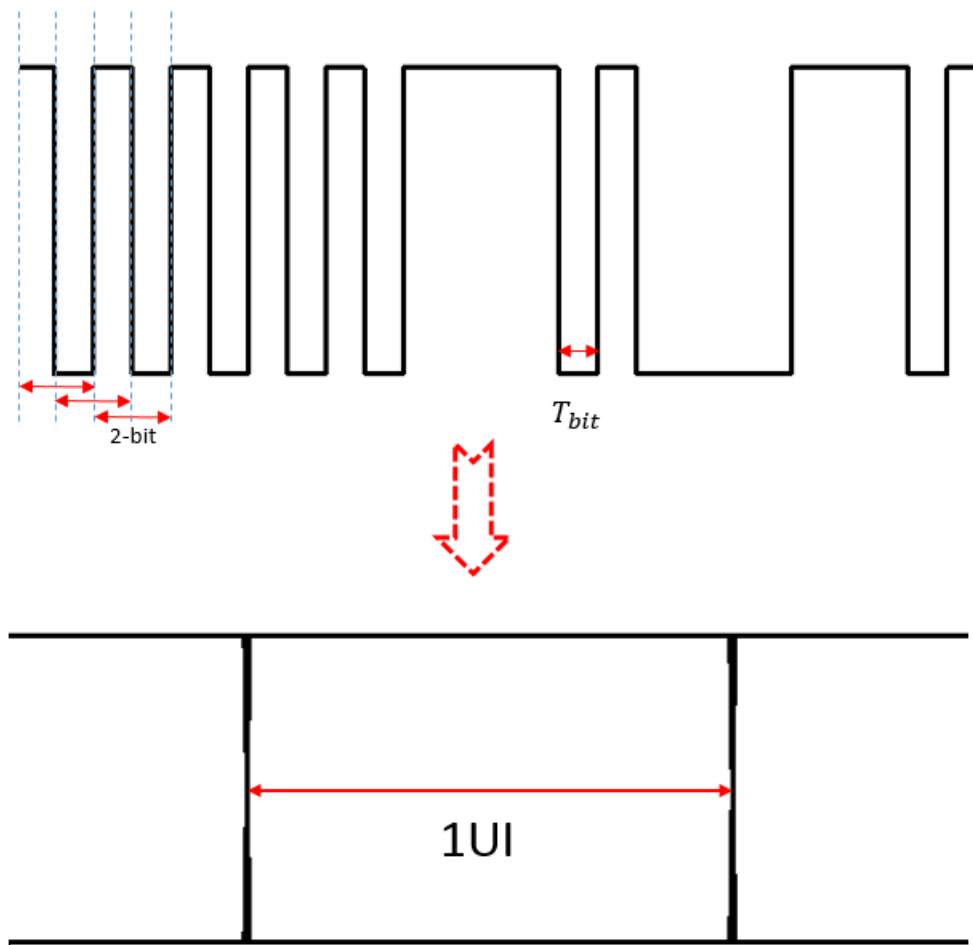


Figure 1.2: Ideal eye diagram in high-speed link signaling.

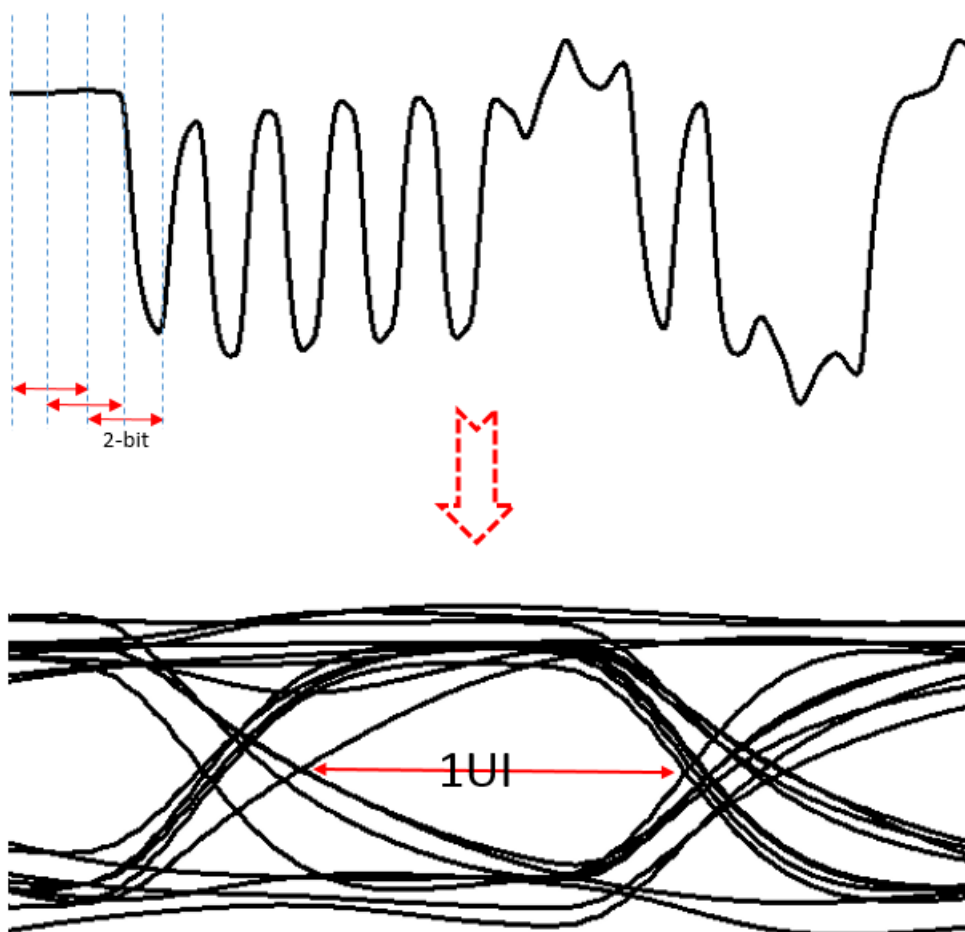


Figure 1.3: Typical eye diagram in high-speed link signaling.

1.3.1 Eye Diagram Fundamentals

In addition to providing direct insights into signal integrity, the eye diagram offers valuable quantitative information from a statistical perspective. As previously explained, the eye diagram combines multiple 1-bit shifting 2-bit windows. However, statistically, it also serves as a visualization of the Probability Density Function (PDF) of the signal's voltage across the 2-UI duration. Put simply, the eye diagram can be conceptualized as a 3D representation: the x-axis represents time, the y-axis represents voltage, and the z-axis represents the PDF. This concept is illustrated in Figure 1.4. When plotting the eye diagram in 2D with color mapping, variations in color or density signify the diverse possibilities of voltage values at specific times. Each point on the diagram represents a combination of voltage and time, with color indicating the PDF of voltage at that particular time. Essentially, within every eye diagram, there exist PDFs for both voltage and time, providing comprehensive insights into signal behavior. As illustrated in Figure 1.5, the voltage PDF is derived by slicing the vertical time frame (indicated by the vertical red line). At each time point, after aggregating all voltage densities vertically, a single voltage PDF is obtained. In this PDF, the x-axis represents voltage, while the y-axis represents density. Similarly, the time PDF is obtained by horizontally slicing the eye diagram as indicated by the horizontal red line in Figure 1.5. Each voltage level yields a distinct time PDF, where the x-axis represents time and the y-axis represents density. Understanding this statistical concept of the eye diagram is crucial before delving into the analysis of its time and amplitude information.

The most common PDF in nature is Gaussian or Normal distribution, which generally has a bell-shaped curve as shown in Figure 1.6 and has a general form as

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}, \quad (1.1)$$

where μ is the mean or average of the data set and σ is the standard deviation that approximately 68 percent of the data is located within $\pm 1\sigma$ of the mean. The μ and 3σ of Gaussian distribution are key parameters used to analyze the eye information statistically. Indeed, several terms are used to describe the characteristics of the eye diagram from both voltage and time perspectives. These terms are defined and calculated based on the distribution of the eye

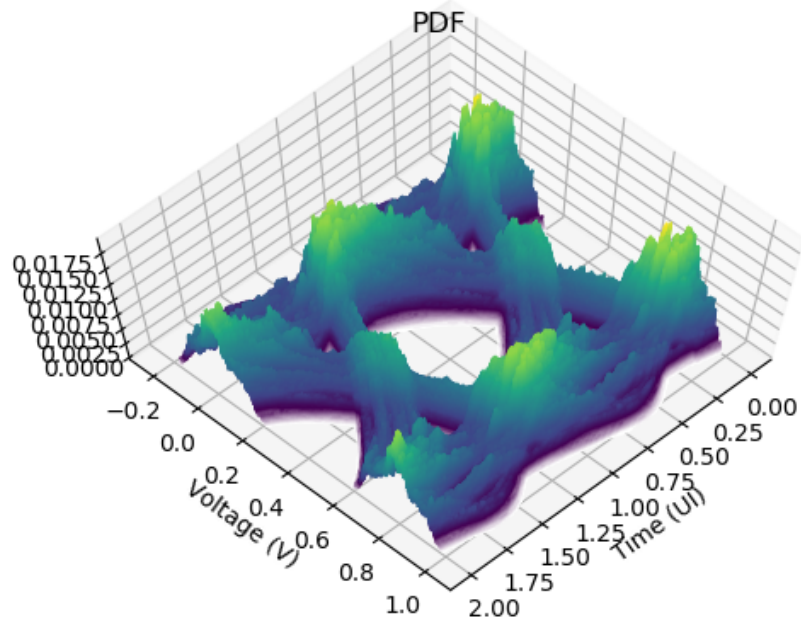


Figure 1.4: A 3D eye diagram visualization.

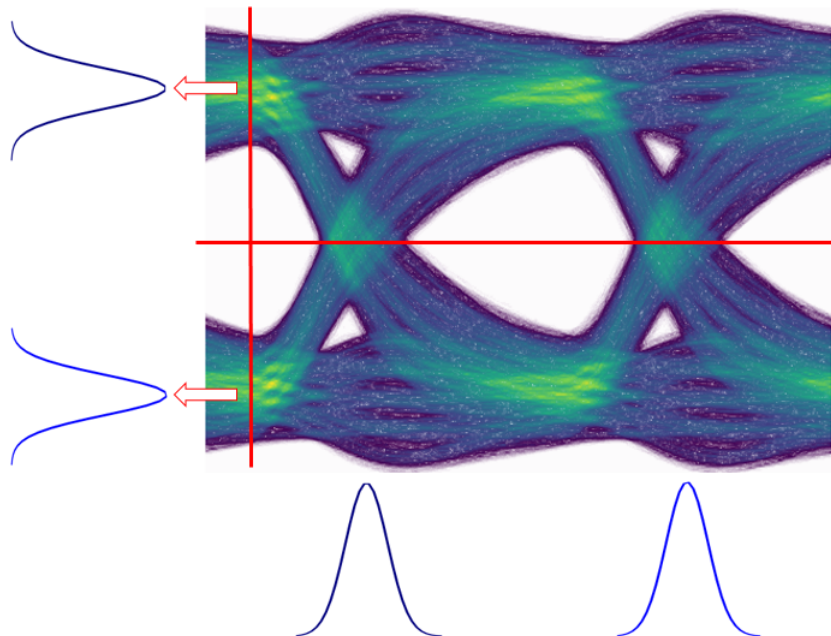


Figure 1.5: Voltage and time PDF from eye diagram.

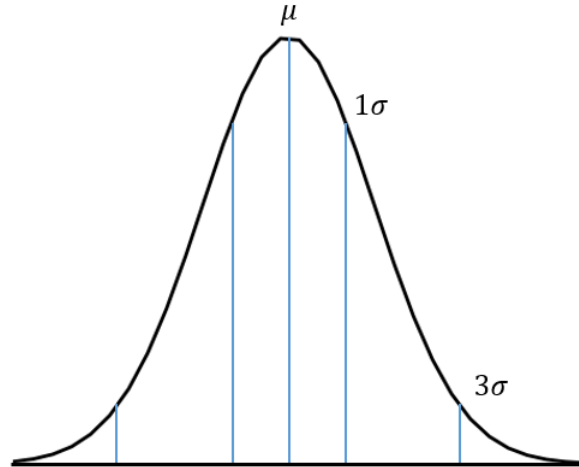


Figure 1.6: Gaussian distribution.

diagram.

1.3.2 One Level

The one level in an eye diagram is the mean value of the top eye diagram distribution in the middle 20 percent of the eye. In Figure 1.7, the highlight region is the middle 20 percent after centralizing the eye-opening, which also refers to the 40 percent to 60 percent region between the one-bit period.

1.3.3 Zero level

The zero level in an eye diagram is the mean value of the bottom eye diagram from the same 20 percent region in one level calculation.

1.3.4 Eye Amplitude

Eye amplitude is the difference between the one level and zero level, which is calculated from the mean values of the two PDF shown in Figure 1.7. The high-speed link receiver determines whether the transmitted bit is a "0" or "1" based on eye amplitude.

1.3.5 Eye Height

The definition of eye height describes the vertical eye-opening of the eye diagram and is derived from the difference between the inner 3σ points between two amplitude PDF in Figure 1.7. An ideal eye height is equal to the eye amplitude. However, due to noise and amplitude attenuation, the eye height is diminished and tends to close.

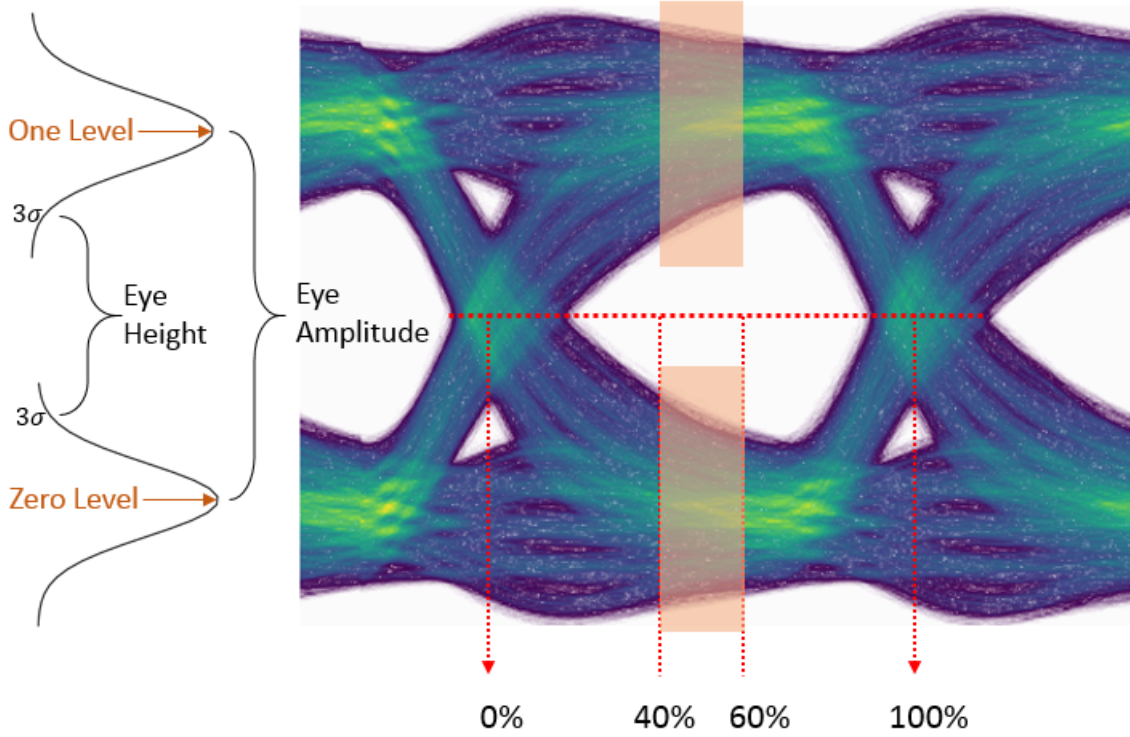


Figure 1.7: Amplitude definitions of eye diagram.

1.3.6 Eye Crossing Percentage

The eye-crossing percentage is the ratio measurement of how the crossing points relative to the one and zero levels. The crossing level is computed by the mean value of a PDF from a thin vertical window centered on the crossing point. The eye-crossing percentage is then calculated using the equation:

$$\frac{\text{crossing level} - \text{zero level}}{\text{one level} - \text{zero level}} \cdot 100\% \quad (1.2)$$

1.3.7 Signal-to-Noise Ratio (SNR)

The SNR is a ratio of the desired signal level to the level of background noise. Higher SNR values are more desirable as more signals are transmitted. The definition is as follows:

$$\frac{\text{one level} - \text{zero level}}{1\sigma \cdot \text{one level} + 1\sigma \cdot \text{zero level}} \quad (1.3)$$

1.3.8 Bit Period

The bit period is called the unit interval (UI) when describing an eye width in a normalized form. The bit period is the inverse of the data rate (i.e., 100 ps bit period is used for a 10 Gbps signal).

1.3.9 Rise Time

In an eye diagram, the rise time refers to the mean transition time it takes for a signal from a low voltage level to a high voltage level. It is typically measured as the time it takes for the signal to rise from 20% to 80% of its maximum amplitude or 10% to 90%. A slow rise time can lead to distortion and signal loss, which can degrade the eye-opening.

$$\text{Rise Time} = \text{mean}(80\% \text{ time level}) - \text{mean}(20\% \text{ time level}) \quad (1.4)$$

1.3.10 Fall Time

The fall time has a similar definition of rise time but on the downward transition time from a high voltage level to a low voltage level. The measurement is typically between 20% to 80% or 10% to 90% amplitude level.

$$\text{Fall Time} = \text{mean}(20\% \text{ time level}) - \text{mean}(80\% \text{ time level}) \quad (1.5)$$

1.3.11 Eye Width

Eye width is the horizontal eye-opening of the eye diagram and is calculated from the difference between the inner two 3σ points on the time PDF at the

two crossing points.

1.3.12 Jitter

The jitter refers to the time variation from the ideal timing position. The jitter histogram can be derived from the time PDF at the crossing point of rising and falling edges. The peak-to-peak jitter is defined as the full width of the jitter histogram as all points are present. The RMS jitter is defined as the standard deviation of the jitter histogram.

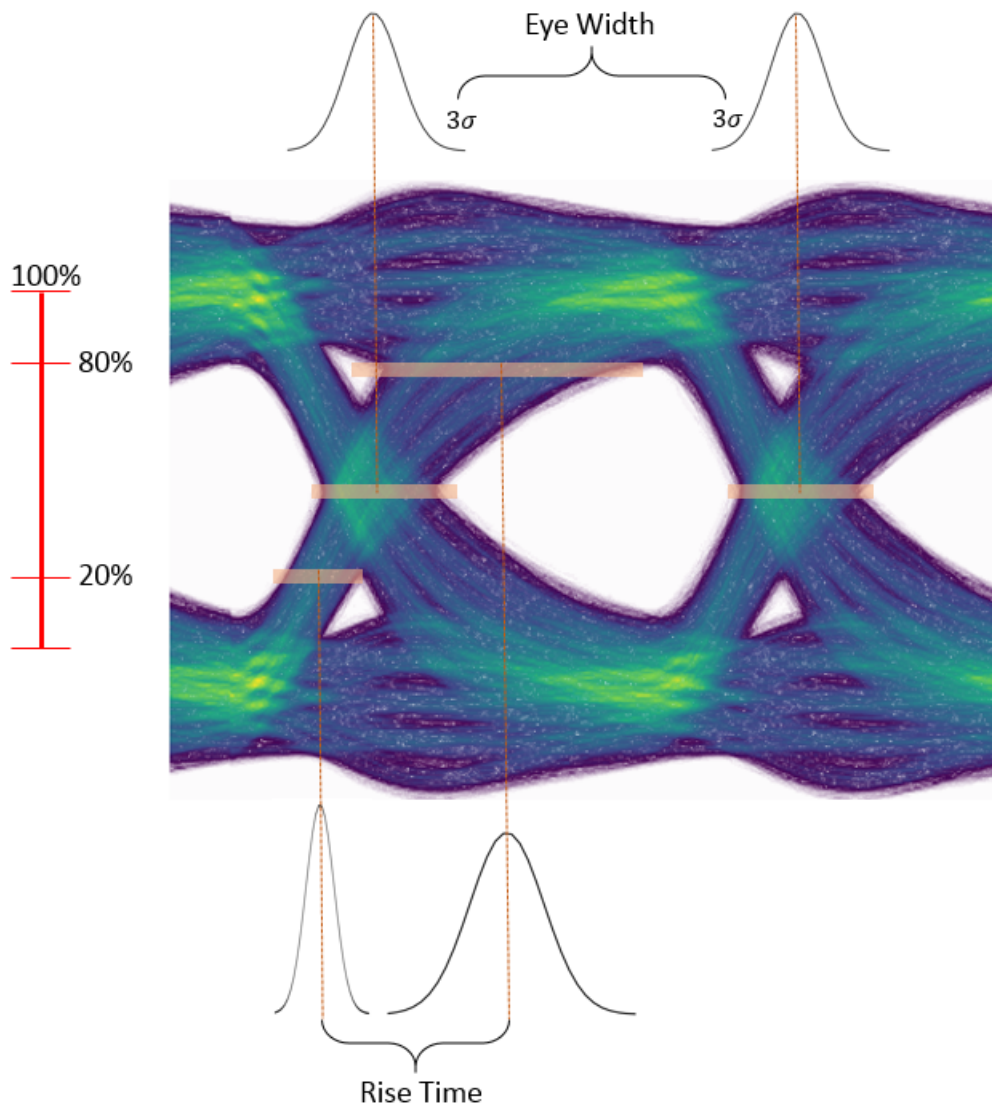


Figure 1.8: Time definitions of eye diagram.

CHAPTER 2

HIGH-SPEED LINK MODELING WITH POLYNOMIAL CHAOS METHOD

2.1 Introduction

The high-speed link system design heavily relies on a thorough performance analysis from various design parameters. Numerous time-consuming computer simulations are required to validate the design's performance and correctness. In the example of transient analysis of the high-speed link eye diagram simulation, the targeted BER of 10^{-13} and complete eye diagram would require days to accomplish. The sooner the designers collect the simulation results, the faster they can receive feedback and improve the design. Therefore, the timing of simulation analysis results becomes prominent. Toward the end, this evolved to a data-driven approach called statistical modeling or surrogate modeling that approximates the simulation output with cheaper simulation time.

The idea of the surrogate model is a black-box mapping problem that describes a relation between $\mathbf{X}$ and $\mathbf{Y}$, as shown in Figure 2.1. Given a set of input parameters, $\{x\}$, the black-box starts the learning process and produces the best regression function $f(x)$ which returns the estimated results $\hat{y}$, being close to simulation results y . The Polynomial Chaos Expansion (PCE) is one type of surrogate model.

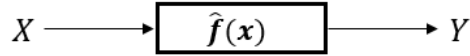


Figure 2.1: The surrogate model black-box.

2.2 Polynomial Chaos Theory

The PCE theory introduces a way to estimate an arbitrary random variable of interest as a function of another random variable with a given distribution and as a model of an orthonormal polynomial expansion. This method is known for its faster convergence and lower computation cost than Monte Carlo (MC) analysis [27]. The statistical information such as the mean and variance of the output is given at no cost with the process of solving the PCE model. Based on [28], the general form of 1-D polynomials is estimated as:

$$y = f(x) = \sum_{i=0}^m c_i \phi_i(x), \quad (2.1)$$

where x is the one-dimensional random variable, y is the predicted output and c_i denotes the unknown polynomial coefficients to be determined. $\phi_i(x)$ represents orthogonal bases with respect to the probability distribution function of x . m represents the order of polynomial expansions and there are $m+1$ terms.

The polynomial function $\phi(x)$ is chosen so they are orthogonal with respect to the probability density function of input parameter x to ensure useful statistical properties. When the input randomness x has Gaussian distribution, the polynomial basis function $\phi(x)$ orthogonal w.r.t. the Gaussian distribution is the set of i -th degree Hermite polynomials $H_i(x)$, given by the function:

$$H_i(x) = (-1)^i e^{x^2} \frac{d^i}{dx^i} e^{-x^2}. \quad (2.2)$$

	Orthonormal Hermite Polynomial
$\phi_0(\lambda)$	1
$\phi_1(\lambda)$	λ
$\phi_2(\lambda)$	$(\lambda^2 - 1)/\sqrt{2}$
$\phi_3(\lambda)$	$(\lambda^3 - 3\lambda)/\sqrt{6}$
$\phi_4(\lambda)$	$(\lambda^4 - 6\lambda^2 + 3)/(2\sqrt{6})$

Figure 2.2: The orthonormal Hermite polynomials.

Other polynomial basis functions are selected if the input random vector x has a different distribution. For example, the $\phi(x)$ is chosen to be Legendre

polynomials if the distribution of x is uniform.

However, multidimensional polynomials are more frequent in practical modeling examples with multiple random variables involved. Instead of a single input, x becomes a multidimensional vector $\mathbf{x} = [x_1, x_2, \dots, x_n]^T$ in the polynomial chaos expansion equation:

$$y = f(\mathbf{x}) = \sum_{i=0}^P c_i \Phi_i(\mathbf{x}), \quad (2.3)$$

where $f(\mathbf{x})$ is the estimated output, $\mathbf{x} = [x_1, x_2, \dots, x_n]^T$ shows the dimension of input variables, and c_i denotes the unknown polynomial coefficients to be determined. $\Phi_i(\mathbf{x})$ represents multidimensional orthonormal polynomials, constructed using the product of the 1-D orthonormal polynomials, via

$$\Phi_i = \prod_{k \in \mathcal{K}_i} \phi_k, \quad (2.4)$$

where $\mathcal{K}_i$ is the multi-index set for 1-D orthonormal polynomials

$$\mathcal{K}_i = \{k_{i1}, \dots, k_{in}\}, \sum k_{ij} \leq m. \quad (2.5)$$

The number of polynomial terms P has the equation below,

$$P + 1 = \binom{m+n}{m} = \frac{(m+n)!}{m!n!}, \quad (2.6)$$

where m is the polynomial order and n is the dimension of the input. When there is the 1-D polynomial case, $n = 1$ and thus $p = m$.

Once the polynomial function $\Phi_i(\mathbf{x})$ has been determined, the next step is to estimate the unknown polynomial expansion coefficients c_i . There are two ways of solving this unknown: intrusive and non-intrusive methods. The non-intrusive method is applied here and the other method can be found in [28]. First of all, Equation (2.3) can be rewritten into a matrix form. Let $\bar{\Phi}$ represent $\Phi_i(\mathbf{x})$:

$$\bar{\Phi} = \begin{bmatrix} \Phi_0(\mathbf{x}^1) & \Phi_1(\mathbf{x}^1) & \dots & \Phi_P(\mathbf{x}^1) \\ \Phi_0(\mathbf{x}^2) & \Phi_1(\mathbf{x}^2) & \dots & \Phi_P(\mathbf{x}^2) \\ \vdots & \vdots & \ddots & \vdots \\ \Phi_0(\mathbf{x}^N) & \Phi_1(\mathbf{x}^N) & \dots & \Phi_P(\mathbf{x}^N) \end{bmatrix} \in \mathbb{R}^{N \times (P+1)} \quad (2.7)$$

Similarly,

$$\mathbf{c} = \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_P \end{bmatrix} \in \mathbb{R}^{(P+1) \times 1} \quad (2.8)$$

as a long column matrix. Then, combining and rewriting in matrix form yields

$$\mathbf{Y} = \bar{\Phi} \cdot \mathbf{c} \quad (2.9)$$

$$\begin{bmatrix} Y^1 \\ Y^2 \\ \vdots \\ Y^N \end{bmatrix} = \begin{bmatrix} \Phi_0(\mathbf{x}^1) & \Phi_1(\mathbf{x}^1) & \dots & \Phi_P(\mathbf{x}^1) \\ \Phi_0(\mathbf{x}^2) & \Phi_1(\mathbf{x}^2) & \dots & \Phi_P(\mathbf{x}^2) \\ \vdots & \vdots & \ddots & \vdots \\ \Phi_0(\mathbf{x}^N) & \Phi_1(\mathbf{x}^N) & \dots & \Phi_P(\mathbf{x}^N) \end{bmatrix} \cdot \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_P \end{bmatrix} \quad (2.10)$$

$\mathbf{Y}$ is a size of N column vector representing the N amount of output, given from N sets of $\mathbf{x}$ random vectors in input. It should be noted that the lowercase n in $\mathbf{x} = [x_1, x_2, \dots, x_n]^T$ means the dimension of input and it is unrelated to the number of input vector N . The input dimension n can be 5 but there could be $N = 1000$ data sets. And N determines how many rows in $\bar{\Phi}$ matrix; however n decides the column size in $\bar{\Phi}$.

The coefficients $\mathbf{c}$'s are linear with respect to $\mathbf{Y}$. Therefore, given N sets of training samples, and assuming N is large enough at least 1.5 times the number of polynomials basis P [14], $\mathbf{c}$ can easily be solved by linear regression method, given by

$$\mathbf{c} = (\bar{\Phi}^T \cdot \bar{\Phi})^{-1} \cdot \bar{\Phi}^T \cdot \mathbf{Y} \quad (2.11)$$

Once the coefficients $\mathbf{c}$ are obtained, whenever new and unseen multidimensional inputs are fed into the $\bar{\Phi}$ matrix the new estimated results $\mathbf{Y}$ can be calculated by the direct matrix multiplication.

2.3 Eye diagram Estimation via Polynomial Chaos Model

2.3.1 Introduction

As the introduction in the previous chapter, the eye diagram is constructed from the output signal waveform by a 2-bit window overlapping. In the transient simulation approach, a lengthy bit pattern is generated from the signal transmitter. After transmitting through the high-speed link system, at the receiver, the corresponding output waveform is detected. Depending on the complexity of the high-speed link system, the requirement of bit numbers to create an eye diagram that well describes the system varies. Generally, the longer the bit pattern, the more comprehensive the eye diagram is. Compared to the transient eye simulation approach, the PC method builds up a surrogate model with a small number of bit sequences. In the estimation procedure, the surrogate model evaluation is much faster than the transient simulation. Therefore, the computational cost and simulation time are affordable with the surrogate model in eye diagram estimation.

2.3.2 Channel Modeling Decomposition

Following the PCE theory, the input of Equation (2.3) is a multidimensional vector $\mathbf{x} = [x_1, x_2, \dots, x_n]^T$ where n is the number of bits or the length of bit sequence. The output y is the transient output waveform after the high-speed link channel. For instance, when n has 2000 bits and the polynomial order m has the order of 3. The number of polynomial terms P is calculated as $\binom{m+n}{m} = \binom{2000+3}{3}$ that is around 1.4 billion, corresponding to the column size of $\bar{\Phi}$ matrix. In the sense of transient simulation, a 2000-bit sequence is a small amount but turns out to create such a huge matrix size due to polynomial order expansion. This matrix requires heavy computation to solve in the linear regression method. Therefore, directly applying the bit sequence and creating a PCE model is problematic and time-wasting. To overcome this issue, the input needs to be truncated and decomposed into sub-models.

The original thought was feeding the whole bit sequence into the surrogate

model as input and then getting the complete waveform. Instead of using one one-bit pattern as input, the long-bit sequence decomposes into four k -length sub-models based on the last 2-bit transitions. The transitions are zero-to-zero, zero-to-one, one-to-zero, and one-to-one. The input dimension n of the original model now reduces to the dimension k where k can be 5-bit length. As illustrated in Figure 2.3, the input bit sequence and the output signal of the channel are shown. The PCE will be used to model the channel. A single-bit pattern separates into 4 groups 5-bit length short sequence according to the last 2-bit transition, as surrogate model input. The corresponding waveform is also grouped into 4 categories as surrogate model output. Now, rather than modeling by 1 polynomial chaos function, 4 polynomial chaos surrogate models are implemented for 4 transitions to construct the full waveform, which is shown in Figure 2.4.

The 4 sub-model PCE functions can be written as

$$\mathbf{y}_{jk}(t) = \sum_{i=0}^P c_{jk_i}(t) \Phi_i(\mathbf{x}_{jk}), \quad (2.12)$$

where $0 \leq t \leq 2 \text{ UI}$ and $j, k \in \{0, 1\}$. The major difference between Equation (2.3) and Equation (2.12) is that the unknown coefficient $c_{jk_i}(t)$ is no longer a single column matrix but with the width of 2 UI. This is due to the digital-to-analog signal conversion that 1 logical bit translates into an analog signal with certain amounts of time points. For example, in general, 1 logical bit requires 16 or 32 sampling points in time with high voltage to represent. Therefore, each UI here has a length of 16 or 32.

The first step of eye diagram estimation using PCE method is to create 4 groups of bit sequences and then calculate c_{00_i} , c_{01_i} , c_{10_i} , c_{11_i} 4 unknown coefficients for steady zero, rising, falling and steady one sub-models, giving a learning bit sequence. Once the polynomial unknowns are calculated, new testing bit sequence and thus its decomposition sequences are applied to function in Equation (2.12). Then, 4 groups of 2UI-length waveforms are estimated. The superposition of results y_{00} , y_{01} , y_{10} and y_{11} is the eye diagram in Figure 2.5. This eye diagram can also be reconstructed and mapped back to a full waveform.

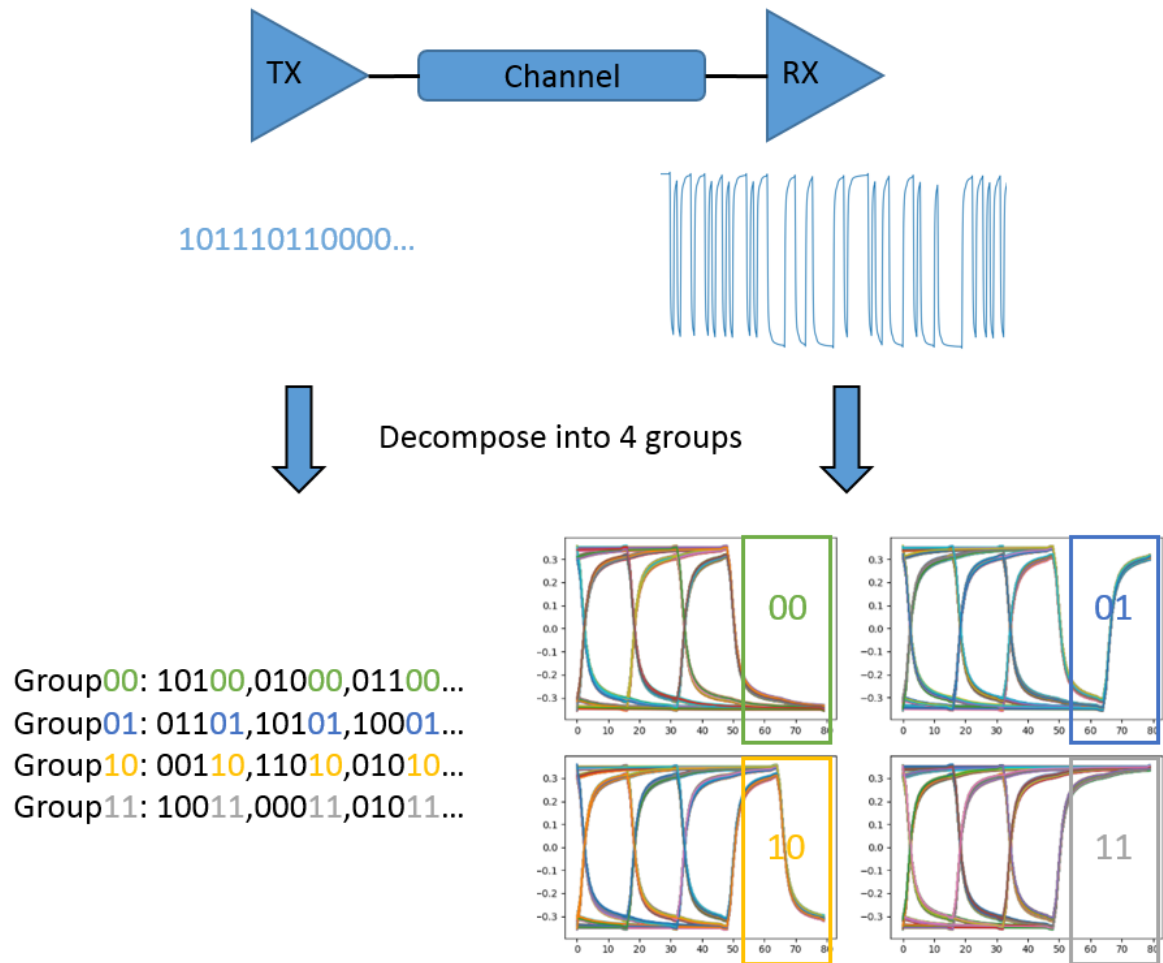


Figure 2.3: Decomposition of single bit sequence.

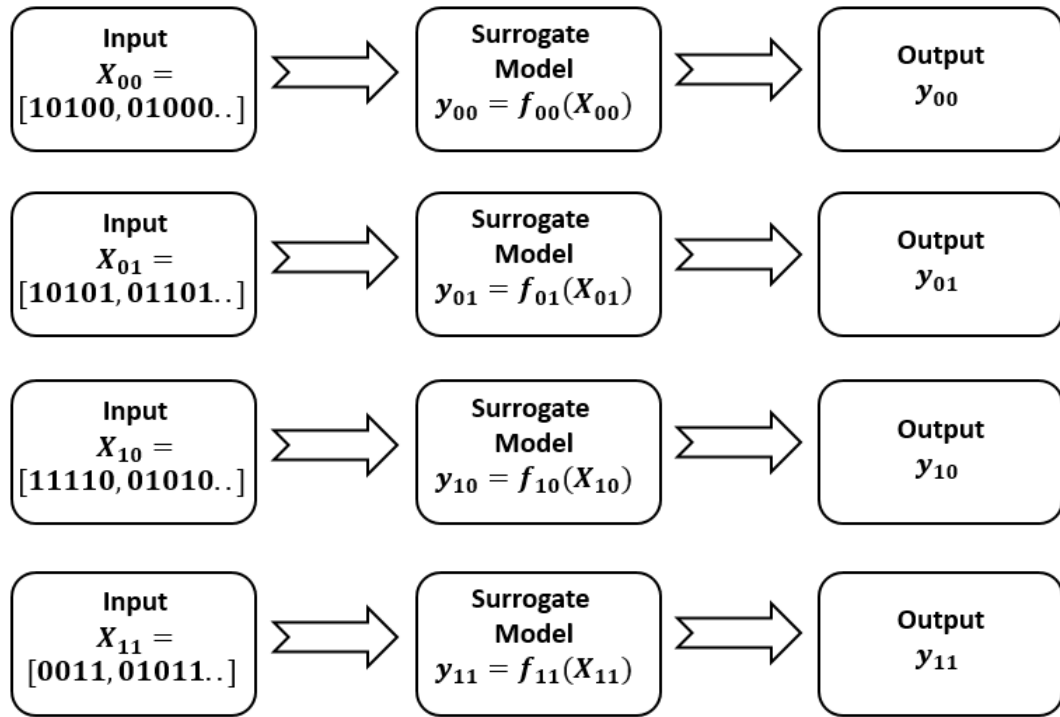


Figure 2.4: Four surrogate models.

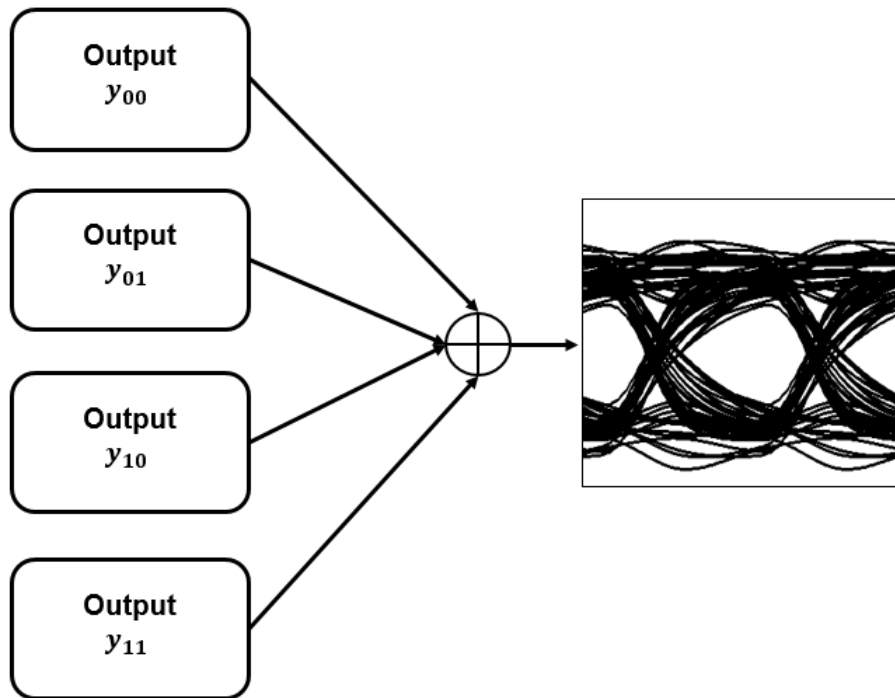


Figure 2.5: Superposition to eye diagram.

2.3.3 Validation

In order to validate the performance of PCE model, a 20000 random-bit sequence is used as a test case for eye diagram estimation. The red eye diagram in Figure 2.6 is from PCE estimation and the blue eye diagram in Figure 2.7 is the reference result from transient simulation. The waveform can also be recovered by reorganizing the eye diagram pattern. This comparison plot in Figure 2.8 shows a good agreement between the two. The Table 2.1 here summarizes the eye information with relatively acceptable differences. For this same amount of bits in simulation, the PCE surrogate model only requires 52 seconds but it takes about 89 seconds in a transient simulation. The surrogate model speeds up by 1.7 times faster.

Table 2.1: Eye diagram comparison

Type	EH (mV)	EW (ns)	One Level (mV)	Zero Level (mV)
PCE	295	755	623	17
REF	281	740	622	18
ERROR	4.98%	2.02%	0.16%	5.56%

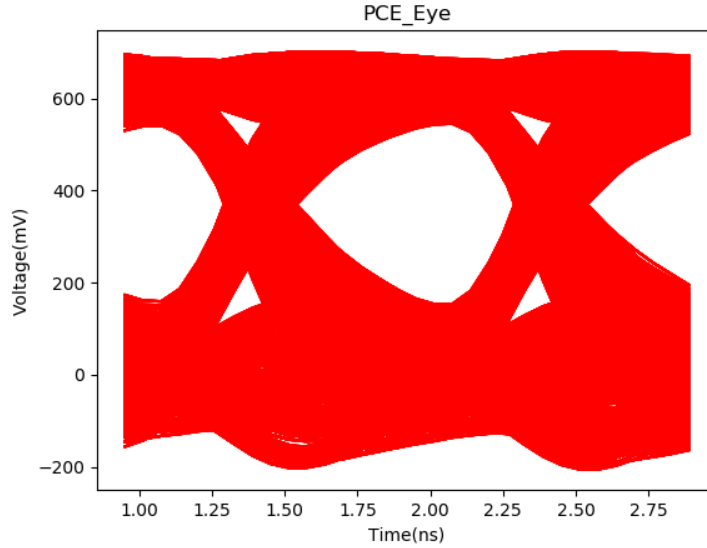


Figure 2.6: Eye diagram from PCE.

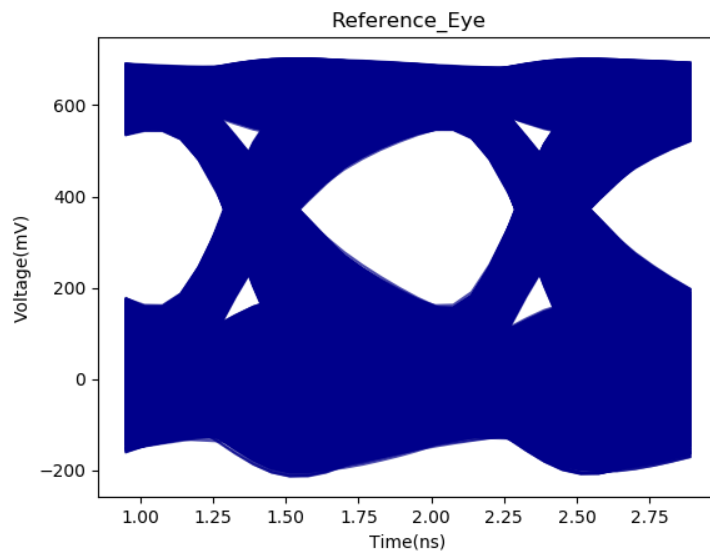


Figure 2.7: Eye diagram from transient simulation.

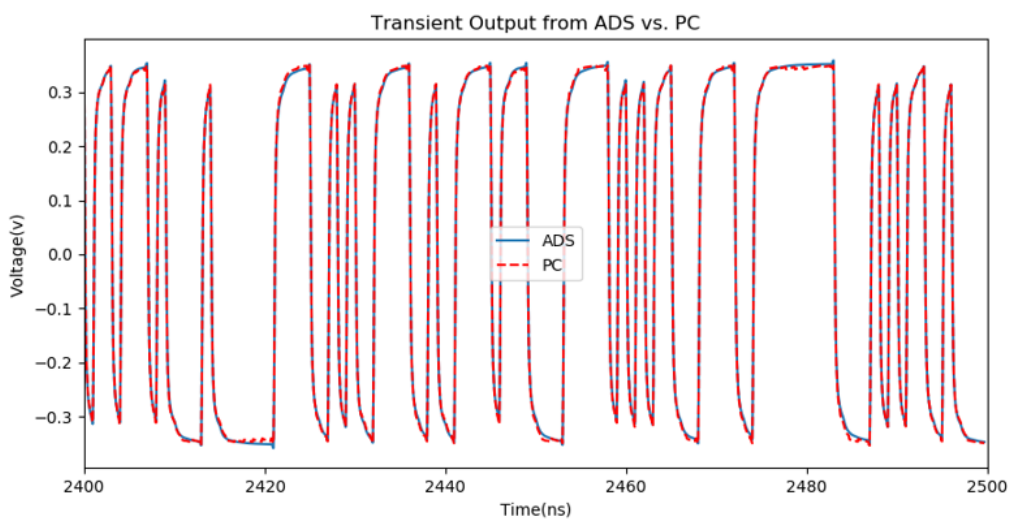


Figure 2.8: Waveform comparison between ADS simulation and PC.

2.4 Decision-Feedback Equalizer (DFE) Taps Estimation with Surrogate Modeling Methods [25]

2.4.1 Introduction

Despite the rapid growth of the electronic circuit industry in recent decades, the demand for high-speed transmission rates remains unabated. Currently, the industry has achieved data rates of up to 32 Gbps and aims to surpass 56 Gbps with the next generation of technologies. When the world enjoys the benefits of fast data rates, electrical engineers have to interact with important signal integrity problems due to the increasing signal transmission speed in electronic devices. The electronic devices are composed of connectors, IC packages, chips, and printed circuit boards. Each component under high-speed circumstances can directly induce unwanted distortions, such as channel loss, crosstalk, intersymbol interference, jitter, and noise, to signal propagating from transmitter to receiver shown in Figure 3.3. To mitigate these distortions, the ability to restore the original signal becomes crucial in addressing signal integrity issues. Equalization emerges as one of the most effective solutions in this regard.

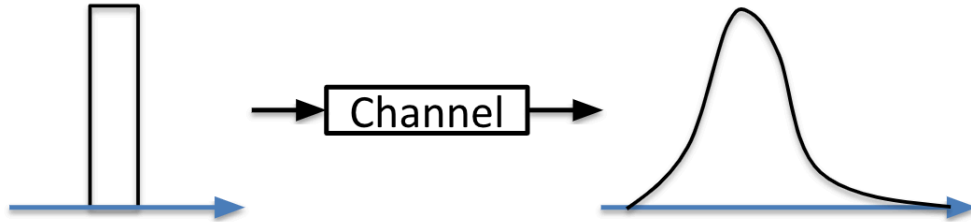


Figure 2.9: Intersymbol interference effect.

Machine learning models based on Decision-Feedback Equalizer (DFE) can be found in [5]. And there are feasible machine learning models predicting eye information from high-speed link physical design specifications [6–8]. However, for system-level device design engineers, understanding the equalization requirements to reduce intersymbol interference and maximize the eye is of greater importance, especially when armed with knowledge of their channel information during the preliminary design phase. For the example of DFE,

it would be advantageous to know the range of tap weights needed for their channel due to variations in channel geometry parameters. Generally, both electromagnetic solver and channel simulation are used to perform equalization analysis with substantial amounts of simulation data and time. A machine learning model can be introduced here to estimate DFE taps requirement from channel geometry. Engineers could foresee how the DFE should behave to restore the compromised signal based on their channel information in a short amount of time. In this study, four different machine learning techniques—polynomial regression (PR), support vector regression (SVR), feed-forward neural network (FNN), and polynomial chaos (PC)—will be explored and compared for DFE tap prediction. PR and SVR can manage various levels of nonlinearity by adjusting the polynomial order and kernel mappings. FNN excels in the regression problem due to its learning capability. However, PC outperforms others with the lowest prediction error rate in this case study.

2.4.2 Regression Methods

2.4.2.1 Polynomial Regression

In the PR, the relationship between the output $\mathbf{Y}$ and the input variable $\mathbf{X}$ is modelled as the n^{th} order polynomial equation. The form is defined as:

$$y = \beta_0 + \sum_{i=1}^m \beta_i \prod_{j=1}^d x_j^{k_j} \quad (2.13)$$

where β_0 is a constant, β_i is polynomial effect parameters, and m is given by $(n + d)! / (n! d!)$. n is the polynomial order, d is the dimension of input variables, and the power $k_j > 0$ must satisfy the condition $\sum_{j=1}^d k_j < n$.

Linear Regression (LR) is a special case of PR when the order number $n = 1$, so the relation between $\mathbf{Y}$ and $\mathbf{X}$ is linear. The least-square will help to derive the β coefficients and find an equation of line or curve which is a close approximation of the actual data points.

2.4.2.2 Feed-forward Neural Network

In this work, the FNN defines a mapping $y = f(x; \theta)$, where x and y represent the system input and output, respectively. The values of the weight parameter θ are learned during the training process to achieve the best function approximation. The FNN architecture used in this study comprises one input layer, three hidden layers, and one output layer. Sigmoid functions serve as activation functions, introducing nonlinearity into the network, while the mean square error (MSE) is minimized through the adoption of stochastic gradient descent (SGD). This approach enables the FNN to effectively capture complex relationships between input and output variables, facilitating accurate predictions and efficient learning from the training data.

2.4.2.3 Support Vector Regression

SVR [29], an important branch of support vector machine (SVM) [30], aims to solve the regression prediction problem by finding a regression plane $h(\mathbf{x}) = \mathbf{w}^T \varphi(\mathbf{x}) + b$, to which all the prediction results of the data set are as close as possible to the corresponding y .

The SVR problem can be regarded as

$$\begin{aligned} \min_{\mathbf{w}, b, \xi, \xi'} \quad & \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{1 \leq i \leq N} (\xi_i + \xi'_i), \\ \text{s.t.} \quad & h(\mathbf{x}^{(i)}) - y^{(i)} \leq \varepsilon + \xi_i, \\ & y^{(i)} - h(\mathbf{x}^{(i)}) \leq \varepsilon + \xi'_i, \\ & \xi_i \geq 0, \xi'_i \geq 0, i = 1, 2, \dots, N \end{aligned} \quad (2.14)$$

where ξ_i and ξ'_i are slack variables, $\mathbf{w} = \{w_1, \dots, w_d\}$ is a normal vector of hyperplane, C is a positive constant and SVR allows a margin of tolerance ε . Lagrange multiplier method is a common method to solve convex optimization problems with constraints. In practice, Kernel method is always added in SVR for non-linear classification. The final SVR predictive function can be calculated via a Lagrange multiplier method:

$$f(\mathbf{X}) = \sum_{i=1}^n (\alpha'_i - \alpha_i) \kappa(\mathbf{X}, \mathbf{X}_i) + d, \quad (2.15)$$

where $\alpha_i \geq 0$ and $\alpha'_i \geq 0$ are introduced as Lagrange multipliers and we select the Gaussian kernel in this problem [?, 10]:

$$\kappa(\mathbf{x}, \mathbf{x}^{(i)}) = \exp\left(\frac{-\|\mathbf{x} - \mathbf{x}^{(i)}\|^2}{2\sigma^2}\right) \quad (2.16)$$

where $\sigma > 0$ is the width of the Gaussian kernel.

SVR algorithm is implemented by MATLAB Statistics and Machine Learning Toolbox, which hyperparameters, e.g. C , σ and ε , are calculated by the Bayesian optimization method in order to minimize the cross-validation error and provide accurate prediction results with robustness.

2.4.3 Example and Results

A differential microstrip-via-stripline discontinuity channel is considered in this paper with 5 design parameters: strip width, channel length, trace space, substrate thickness, and impedance. Firstly, Ansys HFSS is used to obtain the S-parameter of the channel by sweeping geometric parameters. Secondly, transient simulation from Keysight ADS is performed to acquire pulse response. The post-signal processing is then required to collect 3 DFE taps from its post cursors. As shown in Figure 3.4, the post-response distortion gradually dies out after the third unit interval and hence a 3-tap DFE circuit is sufficient to significantly reduce the ISI and avoid over-equalization. When varying the channel geometry parameters, the corresponding pulse response will be generated with the difference in the magnitude of post cursors in Figure 3.4. Therefore, the various 3 taps become the output parameters. There are a total of 13000 sets of geometrical parameters-to-DFE taps samples. When the training samples N are used to train PR, LR, FNN, SVR, and PC models, the rest of the $13000 - N$ unseen samples are used for testing.

To assess the performance of these various models, Figures 3.5, 3.6, and 3.7 depict the prediction average relative error rate of 3 taps on a logarithmic scale to illustrate discrepancies more effectively. The training samples N range from 10 to 200. The average relative error rate decreases from 10% to a minimum of $10^{-6}\%$. As more training samples are utilized, the disparity between predicted and true taps gradually diminishes, as indicated by the figures. LR and PR are utilized to characterize the level of nonlinearity of

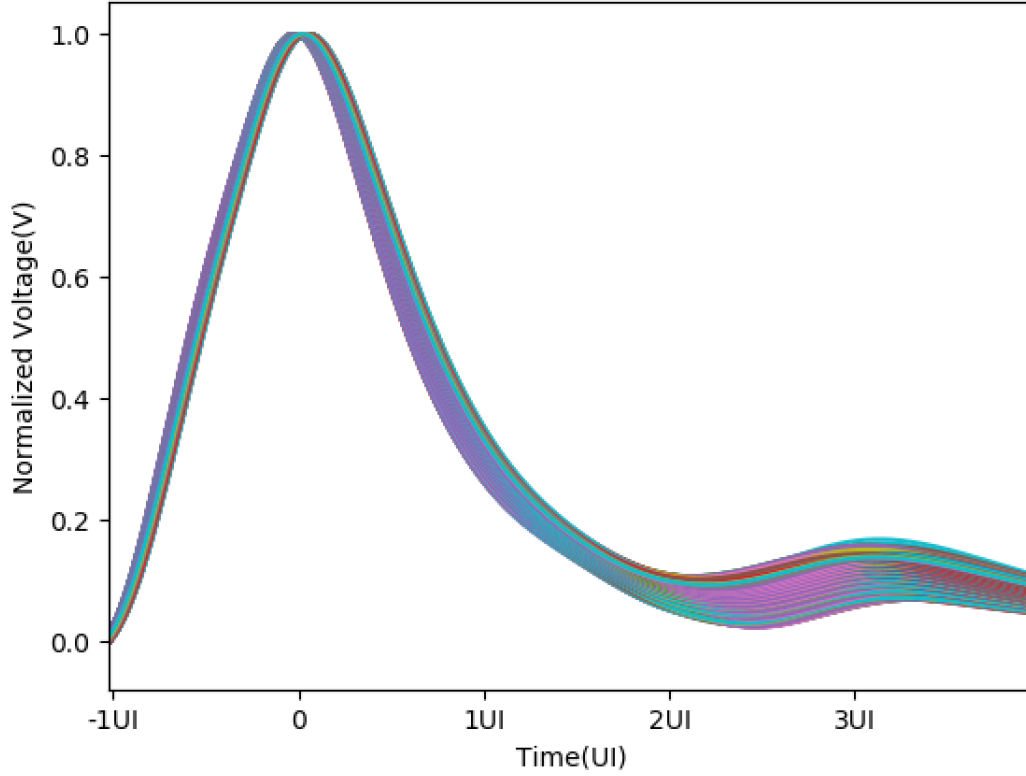


Figure 2.10: Post response variation due to various tap combinations.

the model. While LR, assuming the black-box as a linear model, is capable of predicting results, PR exhibits advancement after 120 samples with an even lower error rate. FNN demonstrates similar performance to LR within the range of 200 training samples but continuously reduces the error rate after 500 samples, unlike LR, which remains constant. In this example, SVR exhibits a stable error rate within this training range and possesses the lowest error rate from 10 to 120 training samples. PC performs the best prediction after 120 samples have been trained.

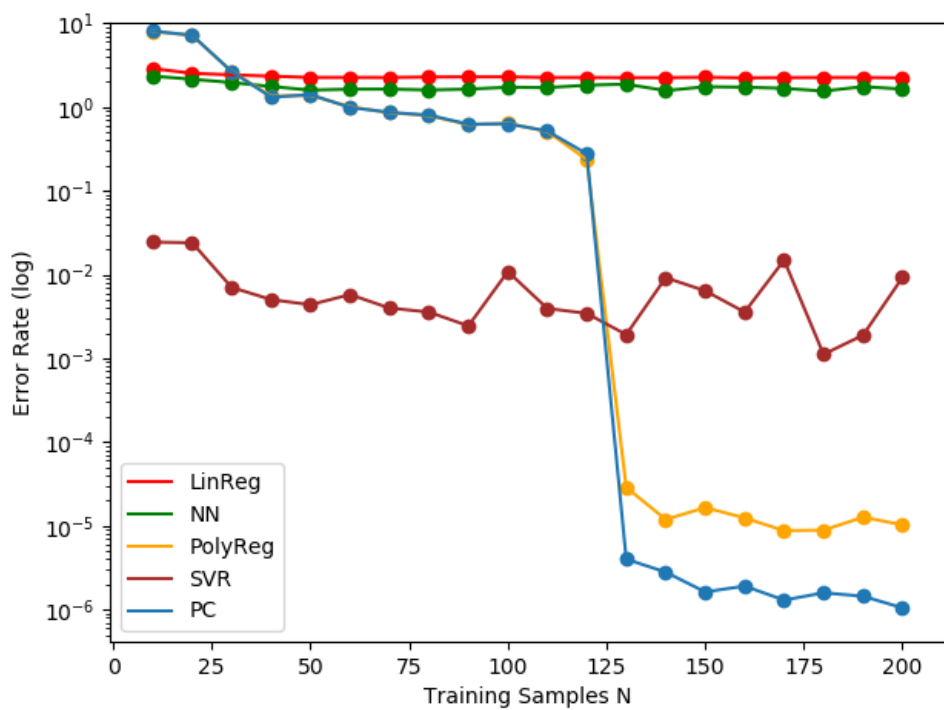


Figure 2.11: DFE tap 1.

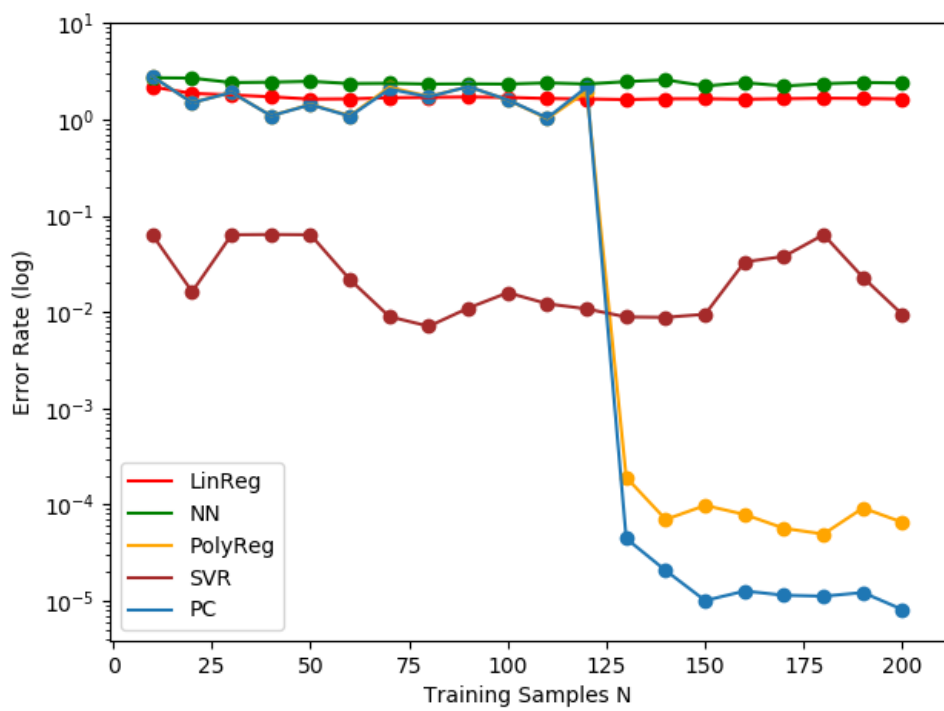


Figure 2.12: DFE tap 2.

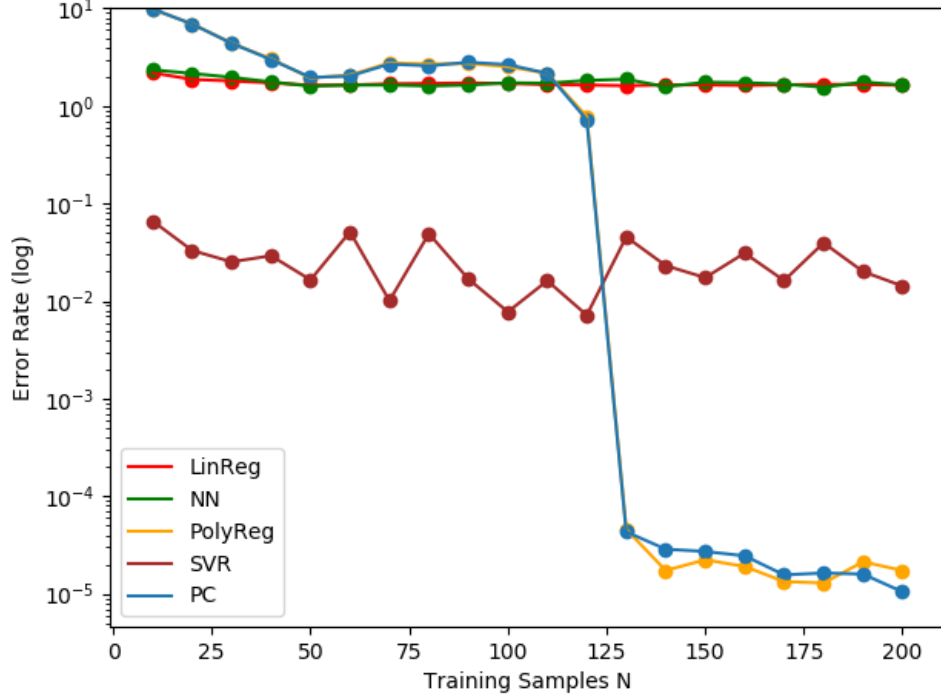


Figure 2.13: DFE tap 3.

2.4.4 Conclusion and Discussion

As a result, a rapid and accurate machine learning model for DFE 3 taps estimation given by 5 geometric channel input parameters is developed with four regression methods, PR, FNN, SVR, and PC, that are presented to show the low prediction error rate. Among them, SVR is the best option to perform prediction when there is a limited number of training samples available. FNN will eventually reach a lower error rate with a slower convergence, but it can be improved for a better nonlinear model prediction with the optimization of hidden layers and activation functions. In general, PC is the more optimal regression method to be used in this example, as after 120 samples the predicted results are almost identical to the true simulated results. Due to its orthonormal polynomial expansion on this problem, PC demonstrates a strong prediction ability and could be used to perform in sophisticated high-speed channel problems in the future.

2.5 Implementation in Microwave Circuit

2.5.1 Introduction

Given today's short product cycles, there is a growing need for rapid and precise modeling methods in the development of next-generation electronic devices. Moreover, the circuitry has become increasingly dense and complex, presenting circuit designers with high-dimensional problems that often require numerous design iterations involving sensitivity analysis and performance optimization. Consequently, the surrogate models are preferred due to their efficiency in optimization or direct MC analysis. This section introduces surrogate models in addition to previous methods, constructed using a non-parametric Gaussian Process Regression (GPR) and Partial Least-Square Regression (PLSR) for comparison. An example involving a millimeter-wave filter is provided to demonstrate the strengths and weaknesses of each method.

2.5.2 Gaussian Process Regression (GPR)

In exact single-output GP regression, given a set of data $\mathcal{D} = \{(\mathbf{x}^{(i)}, y^{(i)}), i = 1, 2, \dots, N\}$ of N pairs of d -dimensional vector-valued input $\mathbf{x}^{(i)} \in \mathbb{R}^d$ and function-valued output $y^{(i)} \in \mathcal{Y} \subset \mathbb{R}$ such that:

$$y = f(\mathbf{x}) + \epsilon \quad (2.17)$$

where $\epsilon \sim \mathcal{N}(0, \sigma^2)$ is the Gaussian observation noise, GP makes prediction, y_* , on a test point $\mathbf{x}_*$ by sampling from the posterior

$$p(y_* | \mathbf{x}_*, \mathcal{D}) = \int_{\boldsymbol{\theta}} p(y_* | \mathbf{x}_*, \boldsymbol{\theta}) p(\boldsymbol{\theta} | \mathcal{D}) d\boldsymbol{\theta} \quad (2.18)$$

where $\boldsymbol{\theta}$ is the hyperparameter vector. The hyperparameter posterior is given by Bayes' rule

$$p(\boldsymbol{\theta} | \mathcal{D}) = \frac{p(\mathcal{D} | \boldsymbol{\theta}) p(\boldsymbol{\theta})}{\int_{\boldsymbol{\theta}} p(\mathcal{D} | \boldsymbol{\theta}) p(\boldsymbol{\theta}) d\boldsymbol{\theta}} \quad (2.19)$$

In implementations, the denominator of Equation (2.19), a.k.a the *evidence*, is the biggest challenge. It is often a high dimensional integral and

intractable. For exact GPs, Equation (2.19) was never calculated, because everything was assumed Gaussian, and the analytical form of $p(\boldsymbol{\theta}|\mathcal{D})$ [31,32] as

$$p(\mathbf{y}_*|\mathbf{x}_*,\mathcal{D}) = \mathcal{N}(\boldsymbol{\mu}_*, \boldsymbol{\Sigma}_*) \quad (2.20)$$

where

$$\boldsymbol{\mu}_* = \mathbf{K}_{tr} (\mathbf{K}_{rr} + \sigma^2 \mathbf{I})^{-1} \mathbf{y} \quad (2.20a)$$

$$\boldsymbol{\Sigma}_* = \mathbf{K}_{tt} - \mathbf{K}_{tr} (\mathbf{K}_{rr} + \sigma^2 \mathbf{I})^{-1} \mathbf{K}_{tr} \quad (2.20b)$$

$\mathbf{K}_{tt}$, $\mathbf{K}_{tr}$ and $\mathbf{K}_{rr}$ are the kernel matrices whose i, j -th element is calculated by evaluating a kernel function, $k(\cdot, \cdot) : \mathbb{R}^d \rightarrow \mathbb{R}$, using 2 data points, $K^{(ij)} = k(\mathbf{x}^{(i)}, \mathbf{x}^{(j)})$. The subscript $\mathbf{t}$ stands for test data while $\mathbf{r}$ stands for training data. The first and second subscript in a kernel matrix indicate which data set the i^{th} and j^{th} data point come from, respectively. Popular kernel functions can be found in [32,33]. The marginal likelihood

$$p(\mathcal{D}|\boldsymbol{\theta}) = \prod_{i=1}^N p(y^{(i)}|\mathbf{x}, \boldsymbol{\theta}) \quad (2.21)$$

is maximized to find the hyper-parameters of the GP.

2.5.3 Partial Least-square (PLS) Regression

Partial Least-square (PLS) regression relies on the idea of principle component analysis. Principle component regression (PCR) involves the principle component analysis (PCA) in which the input space is reduced to the principle component space; then, an interpolation is carried out between a few significant principle components and the output. Assume a multi-input multi-output system, i.e. $\mathbf{y} \in \mathbb{R}^q$. Let

$$\mathbf{X} = \mathbf{V}\mathbf{P}^T \quad (2.22a)$$

$$\mathbf{Y} = \mathbf{U}\mathbf{Q}^T \quad (2.22b)$$

be the principle decomposition of $\mathbf{X} \in \mathbb{R}^{N \times d}$ and $\mathbf{Y} \in \mathbb{R}^{N \times q}$, $\mathbf{V}, \mathbf{P}, \mathbf{U}$ and $\mathbf{Q}$ are of appropriate dimensions. PCR perform regression on $\mathbf{V}$ and $\mathbf{U}$. We can see that though $\mathbf{V}$ best describes inputs and $\mathbf{U}$ best describes outputs as PCA was applied to both input and output, it was applied separately.

PLS fixes this limitation, it iteratively projects input and output onto the most significant components but the projection happens in a leapfrog scheme so that there is cross-information exchange between input and output while doing projections. Formally, starting with j^{th} column of $\mathbf{Y}$, or y_j , set $u_j = y_j$, we start a loop to iteratively update the j^{th} column of $\mathbf{V}, \mathbf{P}, \mathbf{U}$ and $\mathbf{Q}$ until they converge, i.e. stop changing within a specified threshold

$$p_j = \frac{\mathbf{X}^T u_j}{\|\mathbf{X}^T u_j\|} \quad (2.23a)$$

$$v_j = \mathbf{X} p_j \quad (2.23b)$$

$$q_j = \frac{\mathbf{Y}^T v_j}{\|\mathbf{Y}^T v_j\|} \quad (2.23c)$$

$$u_j = \mathbf{Y} q_j \quad (2.23d)$$

where $\|\cdot\|$ is the 2-norm of a vector, then the j^{th} principle component can be projected out, what is left of $\mathbf{X}$ and $\mathbf{Y}$ is used to repeat the same procedure

$$\mathbf{X} \leftarrow \mathbf{X} - v_j p_j^T \quad (2.24a)$$

$$\mathbf{Y} \leftarrow \mathbf{Y} - u_j q_j^T \quad (2.24b)$$

It can be seen that the information about input and output is intertwined thanks to Equation (2.23b) and Equation (2.23d). After L projections, we obtain an L -component decomposition of $\mathbf{X}$ and $\mathbf{Y}$, $\mathbf{V}, \mathbf{U} \in \mathbb{R}^{N \times L}$ and $\mathbf{P} \in \mathbb{R}^{d \times L}$, $\mathbf{Q} \in \mathbb{R}^{q \times L}$. Now a regression model can be created using $\mathbf{U}$ and $\mathbf{V}$

$$\mathbf{U} = \mathbf{V} \boldsymbol{\theta} \quad (2.25)$$

Predictions can be obtained by

$$\mathbf{Y} = \mathbf{U} \mathbf{Q}^T = \mathbf{V} \boldsymbol{\theta} \mathbf{Q}^T = \mathbf{X} \mathbf{P} \boldsymbol{\theta} \mathbf{Q}^T \quad (2.26)$$

2.5.4 Millimeter-Wave Filter

A 12 GHz coupled-line bandpass filter is presented as an example to illustrate the surrogate modeling performance. There are 12 design variables, such as lengths, widths, separations of the coupled lines, dielectric permittivity, and tangent loss. Figure 2.14 shows the insertion loss of the filter as the change

of design variables.

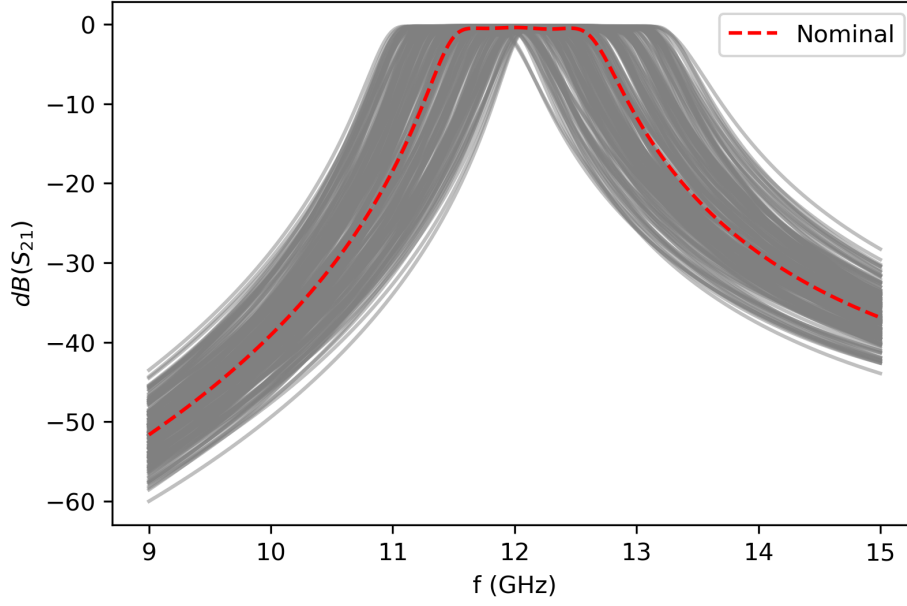


Figure 2.14: Filter insertion loss variations [12].

To quantify the insertion loss of the filter, the center frequency (y_0), bandwidth (y_1), and shape factor (y_2) are computed. Surrogate models were developed to predict these three figures of merit (FOM). Figures 2.15 to 2.17 illustrate the training process for each FOM. Initially, most models converge rapidly, with the exception of the PC model, which requires a larger number of samples to achieve a validation R^2 score of 0.99. Secondly, models focused on single output encounter greater difficulty in learning the shape factor compared to the center frequency and bandwidth, as shown in Figure 2.17. PLS and SVR models require additional training samples to converge compared to MOPLS or MOGP. As previously mentioned, MOPLS, MOGP, and PC are multi-output models, hence a single R^2 score determines their convergence. The quick convergence of LR model suggests that despite the problem's high dimensionality, it is relatively simple and straightforward, as the mapping between the design variables and insertion loss FOMs is predominantly linear.

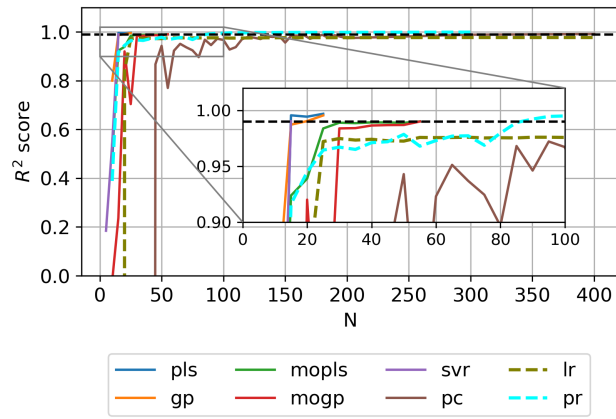


Figure 2.15: Center frequency as output [12].

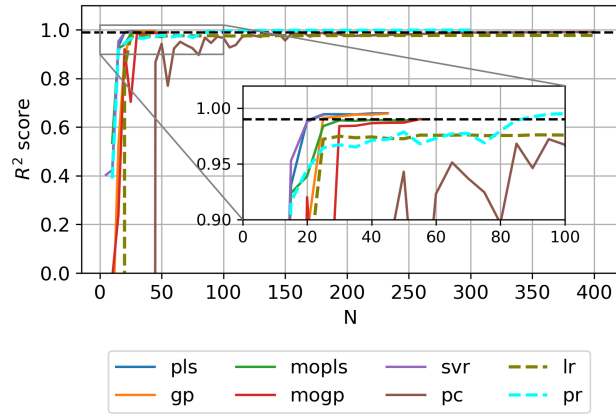


Figure 2.16: Bandwidth as output [12].

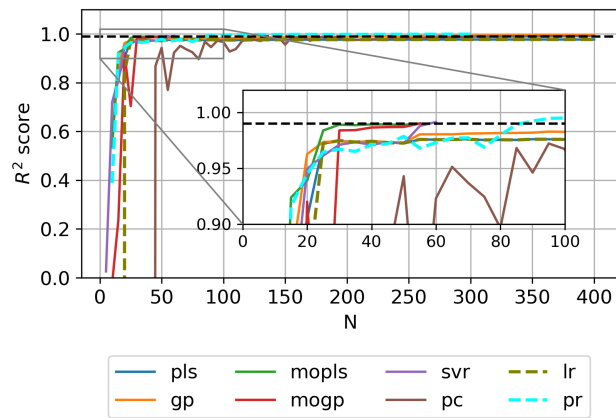


Figure 2.17: Shape factor as output [12].

CHAPTER 3

HIGH-SPEED LINK METHODS IN LINEAR TIME-INVARIANT SYSTEM

3.1 Introduction

The eye diagram is a critical figure of merit for engineers that allows to estimate the performance of high-speed links. Signal integrity (SI) and power integrity (PI) engineers rely heavily on the eye margins of products in low-bit error rate (BER) conditions. Conventional eye diagram estimation methods are based on time-consuming transient simulation methods that use a pseudo-random bit sequence (PRBS) excitation, which provides an accurate and complete output waveform in either linear or nonlinear systems. The larger the bit sequence, the more accurate an eye diagram is obtained at the cost of a longer running time. To remedy lengthy bit sequence simulations, various methods of fast and accurate eye diagram estimations have been proposed [19,34,35], the majority of which are based on superposition and linear time-invariant (LTI) assumptions.

There are two primary categories for characterizing the system performance to high-speed links: the transient simulation and the statistical simulation. In transient simulation, the outcome is the entire output waveform over time, from which the eye diagram is derived by truncating the waveform. However, in statistical simulation, the waveform is bypassed, and the focus is directly on generating the eye diagram. Transient simulation provides precise dynamic voltage and time results over a simulation period, albeit with a lengthy runtime. Consequently, it is not feasible and time-demanding to estimate the worst-case eye performance and anticipate the degree of eye-opening for a low bit error rate (BER) case. For instance, if the system acceptable BER is 10^{-12} , the input sequence of 10^{12} bit is transmitted with the expectation of only 1-bit failure. Simulating millions of bits over time is unacceptable and a challenge for complex circuits. Therefore, people

have been developing fast time-domain simulation methods to overcome the timing issue of the traditional SPICE transient simulation. In [36, 37], the voltage-in-current latency insertion method (VinC LIM) is a fast transient circuit simulation algorithm with superior stability to simulate the waveform in high-speed link systems. Casper et al. [19] introduced the peak distortion analysis (PDA) method to predict the worst-case eye for a high-speed channel, utilizing the single-bit response (SBR) to determine the output for all potential input patterns through superposition. However, the PDA method only identifies the worst inner contour of the eye diagram. To address cases with asymmetric rising and falling edges, the double-edge response (DER) method [34] was proposed. This method decomposes the input data pattern into rising and falling edge transitions, enabling the calculation of the system response by superimposing shifted versions of the rising and falling edge responses. Additionally, the concept of multiple-edge response (MER) was introduced by Oh et al. [38] and Ren et al. [35], wherein multiple rising and falling edges are constructed based on preceding bit patterns. Sanders et al. [39, 40] and Casper et al. [19] introduce a novel methodology based on a statistical method that allows fast and accurate compliance testing of differential channels, in which the eye diagram is estimated directly from the SBR. [18] further implements MER to statistical eye diagram directly for a nonlinear system with advanced speed and accuracy.

This chapter begins by examining the PDA method, which utilizes the SBR to determine the worst-case eye. Following that, it explores the bit-by-bit simulation employing the DER to generate the complete waveform and eye diagram. Lastly, the chapter concludes with an overview of the statistical simulation approach, which directly generates the eye diagram.

3.2 Peak Distortion Analysis

The peak distortion analysis is introduced to determine the worst case received eye-opening from the single-bit response in the LTI system. The single-bit response labeled as $y(t)$ of a system is given by

$$y(t) = x(t) * h(t) \quad (3.1)$$

where $x(t)$ is the single bit input to the LTI system, $h(t)$ is the impulse response of the channel and receiver, and symbol $*$ denotes the convolution process. A typical single-bit response is viewed in Figure 3.1. The intersymbol interference (ISI) due to reflection, channel resonances, and channel loss is represented as oscillations and extended tails in the single-bit response. This interference can lead to distortion within the current bit due to neighboring bits, thus the maximum distortion can be directly assessed by superposing all ISI at multiples of the bit period. Within the cursor concept, it represents the magnitude of a combination of pre-cursors and post-cursors, positioned at integer multiples of the unit interval away from the main cursor. In order to find the worst case eye-opening, the procedure splits into finding the worst case 1, or the worst top half of the eye, and the worst case 0, or the worst bottom half.

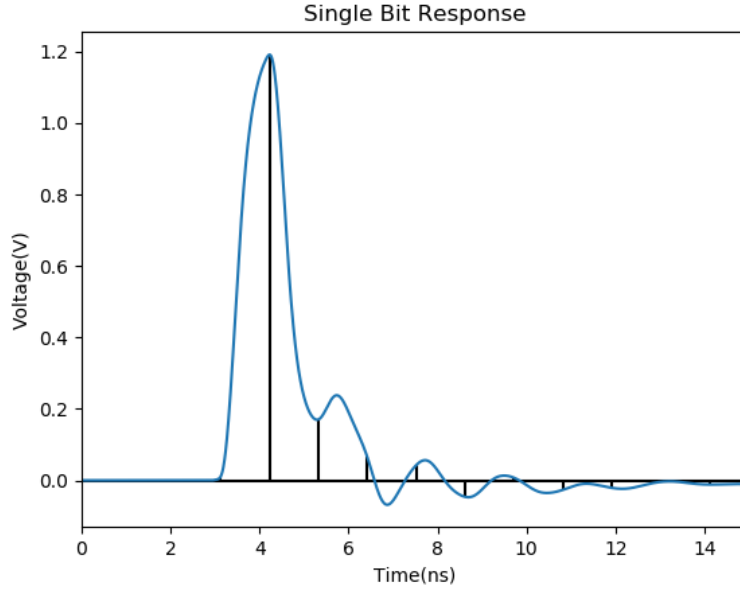


Figure 3.1: Single bit response.

The worst case 1 is the superposition of all negative $k \neq 0$ ISI (negative pre-cursors and post-cursors) with the single-bit response. The equation is shown as

$$S_1(t) = y(t) + \sum_{k=-\infty, k \neq 0}^{\infty} y(t - kT) |_{y(t-kT) < 0} \quad (3.2)$$

where $y(t)$ is the single bit response, T is the bit period and k is the integer multiples. The second term describes the total distortions that all negative

ISI can contribute on the $y(t)$, which has negative value, to reduce or distort the peak response voltage level. Therefore, $S_1(t)$ is the maximum voltage level response after negatively distorted effects. Similarly, the worst case 0 is the superposition of all positive $k \neq 0$ ISI (positive pre-cursors and post-cursors). The equation is shown as

$$S_0(t) = y_0(t) + \sum_{k=-\infty, k \neq 0}^{\infty} y(t - kT)|_{y(t-kT) > 0} \quad (3.3)$$

where $y_0(t)$ is the single 0 bit response, which the input sequence only has one 0 bit. Ideally, $y_0(t)$ is symmetric to $y(t)$ by flipping the amplitude. With same analogy, the second term in Equation (3.3) is positive and contributes on positive effects on level 0, thus shifting the eye bottom upwards. Then, the worst case eye opening $e(t)$ is the difference between the worst case 1 and worst case 0. As shown in Figure 3.2, the blue curve is the result by applying Equation (3.2) on the single bit response in Figure 3.1, which shows the top boundary of the worst eye. The red curve is the result of Equation (3.3) to show the bottom boundary of the worst eye.

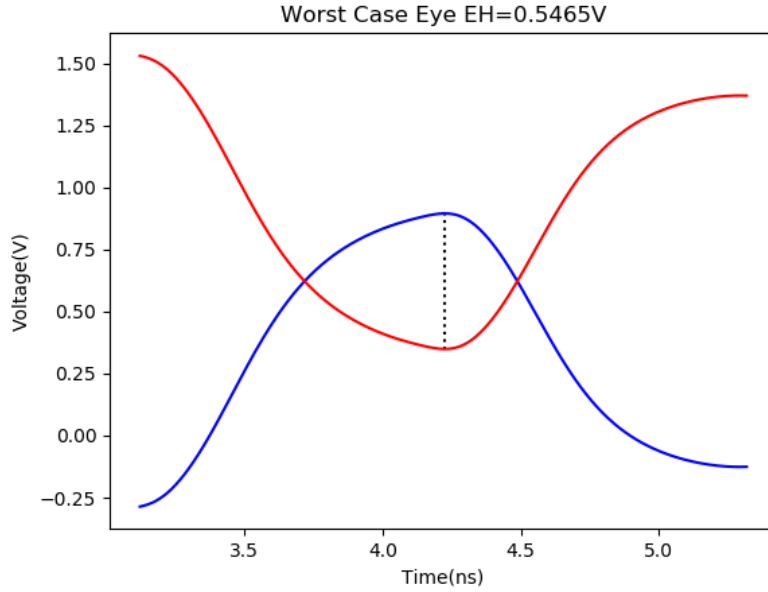


Figure 3.2: The worst eye-opening.

The Equation (3.2) and Equation (3.3) represent the scenario of a jitter-free and single channel system. If there are multiple sources of co-channel

interference and transmitter or receiver jitter in system, more terms like Equation (3.4) can be added into equation, where i is the number of co-channel interference and t_i is the relative sampling point of each co-channel single bit response.

$$\sum_{i=1}^n \sum_{k=-\infty, k \neq 0}^{\infty} y^i(t - kT - t_i) |_{y(t-kT-t_i) > 0} \quad (3.4)$$

3.3 Double-Edge-Based Bit-by-Bit Approach

3.3.1 Double Edge Response

If the transmitted signal in an LTI system stabilizes within a significantly shorter time than a one-bit period, the voltage at the channel output can be accurately estimated by superimposing the edge responses. To elaborate on the concept of edge response, let's consider the example of a rising edge response from a zero to one transition input, as shown in Figure 3.3. When the digital input transitions from zero to one, the output after passing through the channel naturally rises from V_{low} to V_{high} , and this rising edge output is denoted as $V_r(t)$. V_{low} and V_{high} represent the steady-state low and high values, respectively. Furthermore, the channel delay, denoted as D and illustrated in Figure 3.3, refers to the time difference between the initial voltage point of the input and the output just before the rising edge begins during the transition.

Similarly, when the input undergoes a transition from one to zero, the output will eventually decrease from V_{high} to V_{low} . Assuming symmetric rising and falling edges, we define $V_f(t)$ as the falling edge response, which intentionally mirrors $V_r(t)$ but in the opposite direction, ranging from $-V_{low}$ to $-V_{high}$. This falling edge response is plotted in the lower part of Figure 3.4.

3.3.2 Simulation Analysis

Figure 3.4 presents the rising edge and falling edge after removing the channel delay, and these edges will be used to superpose the output waveform based

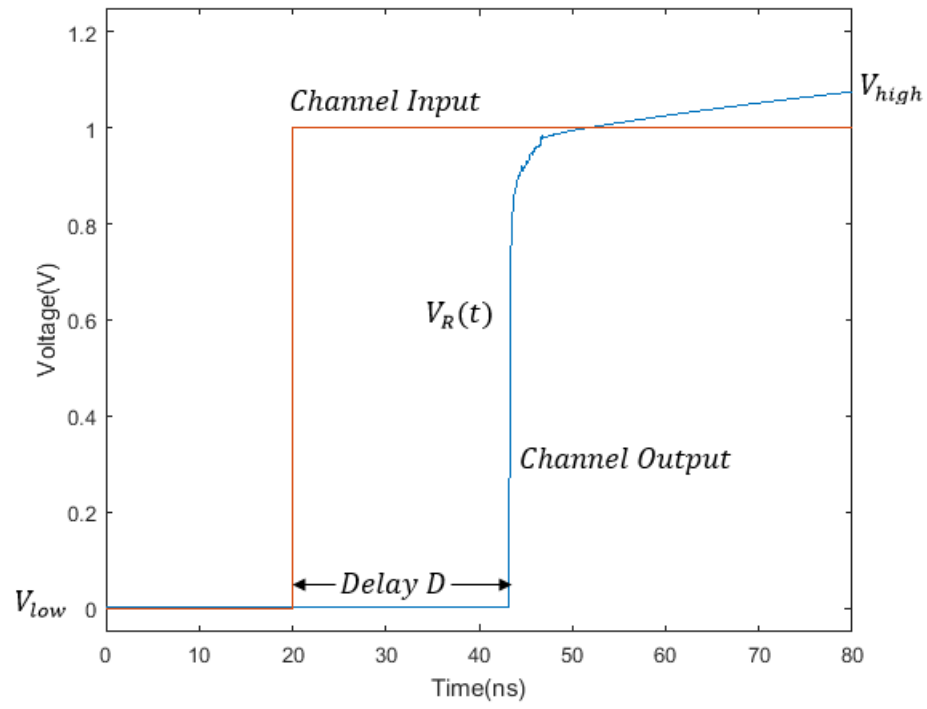


Figure 3.3: Rising edge response with notations.

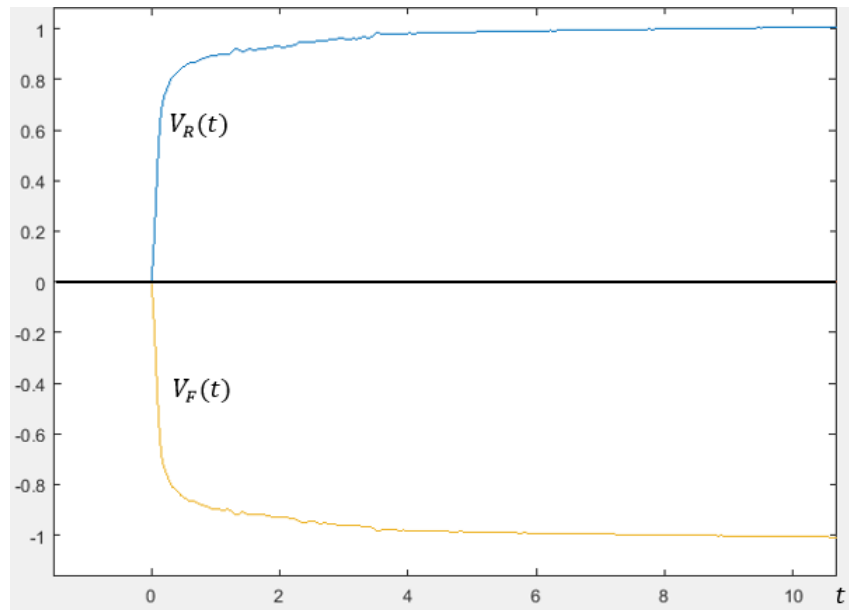


Figure 3.4: Rising and falling edge responses.

on two consecutive bits from the input bit sequence. When the k^{th} bit is 0 and the $(k + 1)^{th}$ bit is 1, the rising response must happen on the output waveform between the corresponding time interval of k^{th} and $(k + 1)^{th}$ bit. The output waveform is defined when only the rising edge presents as:

$$y(t) = \sum_{k=1}^N V_r(t - kT), \quad (3.5)$$

where $y(t)$ is the output waveform, the coefficient k represents the k^{th} bit position, N represents the total number of bits, and T is the time interval of each bit. The shifted rising edge is added after the kT time position of the output waveform. Similarly, when the k^{th} bit is 1 and the $(k + 1)^{th}$ bit is 0, the falling response must happen starting from the time corresponding to the k^{th} bit as

$$y(t) = \sum_{k=1}^N V_f(t - kT) \quad (3.6)$$

Since the $V_f(t)$ is defined in the negative domain, adding this negative edge will bring the previous V_{high} tail from the rising response to a zero-level tail in the output waveform. In the cases of continuous 0 bits and 1 bits, the output waveform will remain in its pattern without adding either a rising response or a falling response.

3.3.3 Example

An example of edge-by-edge superposition is shown below. The input bit pattern is “0100” and each bit takes $T = 1$ ns. k here will be 1 to 5 and the transient time t is from 0 to 5 ns. The first bit has to be considered individually first. If the first bit is 1, the initialization needs to be done to pre-set a constant V_{high} at the first time interval for the output waveform, which means

$$y(t) = \begin{cases} V_{high}, & 0 < t \leq T \\ V_{low}, & T < t < nT \end{cases}$$

Similarly, when the first bit is 0,

$$y(t) = V_{low}, \quad 0 < t < nT$$

Here, in the example, the first bit is 0 so V_{low} is pre-set in the first 1 ns at output waveform, $y(t) = V_{low}, 0 < t \leq 1$ ns.

Following the algorithm description above, the first bit ($k = 1$) is 0, and the second bit is 1, then $y(t) = V_r(t - 1)$, shown as the blue rising pulse in Figure 3.5. The rising edge response is filled to the output waveform starting from 1 ns till 4 ns. The second bit ($k = 2$) is 1 and the third bit is 0, then $y(t) = V_r(t - 1) + V_f(t - 2)$, shown as falling pulse in Figure 3.5. The falling edge response is added in from 2 ns to 4 ns. In the end, the third and fourth bits are 0, and the output waveform from 3 ns to 4 ns is unaltered being V_{low} . When all 4 bits are considered by adding all rising and falling edges in Figure 3.5, the final summation plot is shown as Figure 3.6.

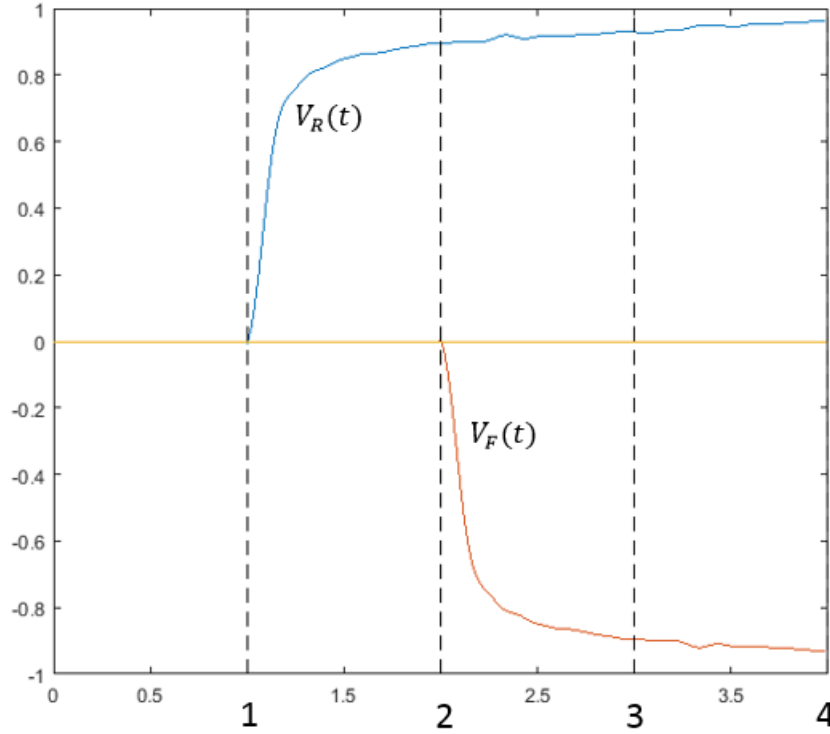


Figure 3.5: Rising edge response at 1 ns and falling edge response at 2 ns.

An additional example of a bit sequence “101001” is illustrated in Figure 3.7. When analyzing this sequence, the output waveform includes $V_f(t - 1)$ upon encountering the first “10” bit sequence, adds $V_r(t - 2)$ upon observing the first “01” bit sequence, includes $V_f(t - 3)$ upon encountering the second “10” bit sequence, and adds $V_r(t - 4)$ upon observing the second “01” bit sequence.

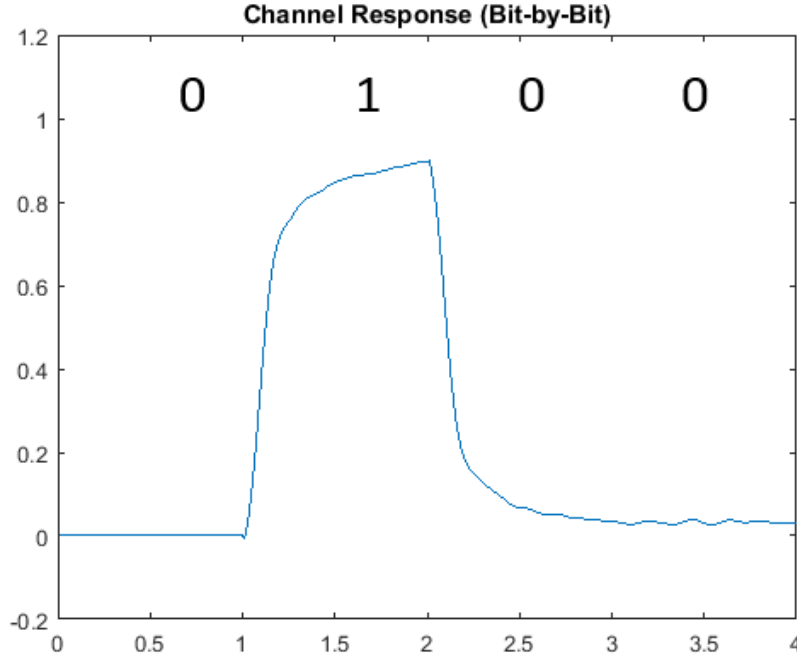


Figure 3.6: The channel response of bit sequence “0100”.

3.3.4 Eye Diagram

Once the output waveform is established, an eye diagram can be presented. An eye diagram is a straightforward tool used in the high-speed link to evaluate the high-speed system performance. It is constructed by slicing the transient signal waveform from the receiver into the size of two UIs and then overlapping every section on top of each other. Consequently, the horizontal axis spans two symbols in time, while the vertical axis depicts the voltage magnitude of the receiver signal. Figure 3.8 and Figure 3.9 illustrate an eye diagram example of a 5 Gbps 200-bit data pattern generated from the traditional transient simulation and the bit-by-bit method. The eye-opening in the center part of the eye diagram indicates the clarity of the signal. A larger eye opening implies more margin for timing and voltage requirements, making it easier for the receiver to distinguish between the digital 1 and 0. The eye height from the transient simulation is 0.324 V and the eye height from the bit-by-bit simulation is 0.312 V. The eye width from the transient simulation is 0.14 ns and from the bit-by-bit simulation is 0.14 ns. The run-time for a 200-bit data pattern in edge-based bit-by-bit mode halves from 3.7 seconds to 1.8 seconds.

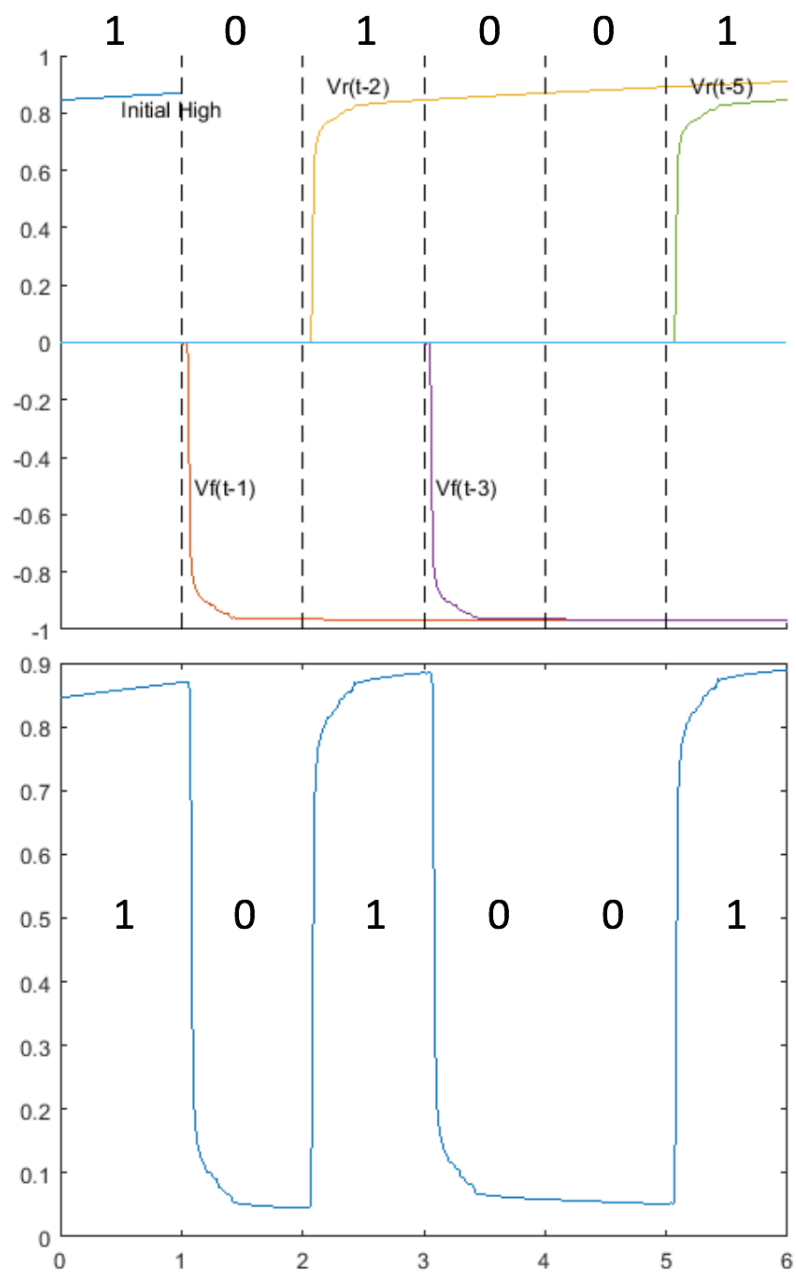


Figure 3.7: The channel response of bit sequence "101001".

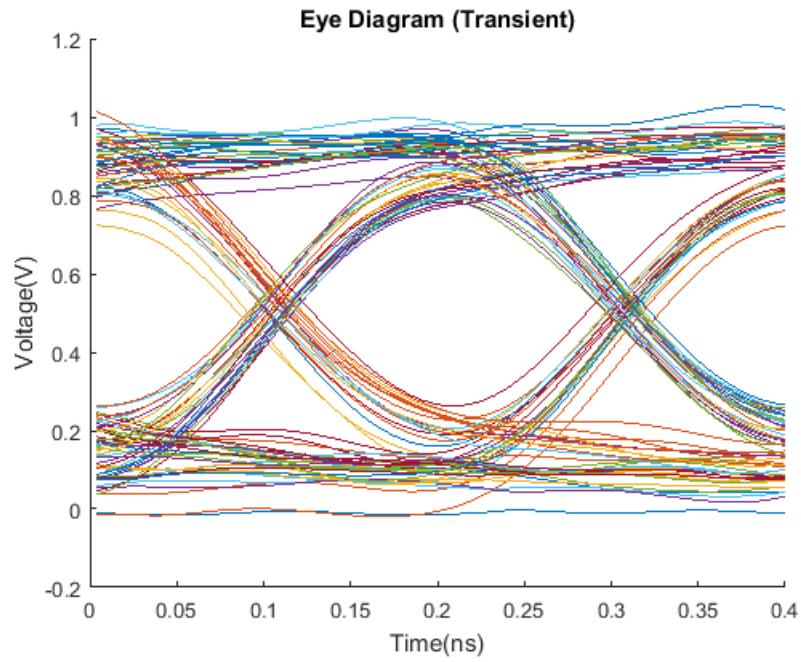


Figure 3.8: Transient eye diagram for 5 Gbps 200-bit data sequence.

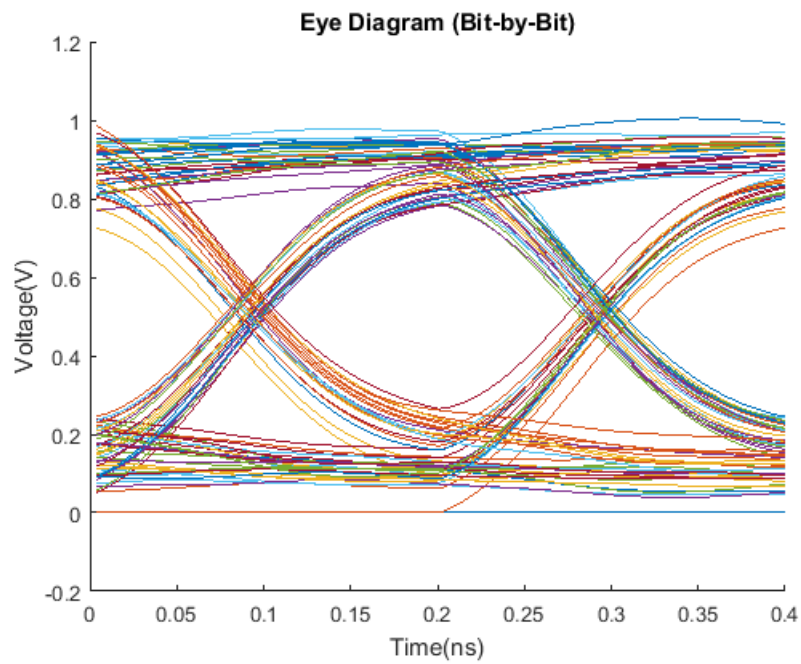


Figure 3.9: Bit-by-bit eye diagram for 5 Gbps 200-bit data sequence.

3.4 Statistical Analysis in the Linear Time-invariant System

The statistical analysis is an advanced peak distortion analysis by taking the probability of ISI occurring into account. In peak distortion analysis, negative ISI values are aggregated to decrease the top of the eye diagram, while positive ISI values are aggregated to raise the bottom of the eye. The underlying assumption is that all potential positive or negative ISI scenarios are accounted for simultaneously, constituting a superposition of all possibilities. The resulting eye diagram, as depicted in Figure 3.2, shows two main regions: one within the eye where errors are unlikely to occur, and another outside the eye where errors are expected to occur. However, the statistical analysis yields a probabilistic representation of the eye diagram, showcasing the probability as a function of voltage and sampling time.

In calculating statistical eye, the superposition of pre-cursors and post-cursors to form ISI is statistical. For the SBR with a finite number of cursors, each cursor has an equal probability of being at level 1 or level 0. The probability of each possible combination to generate ISI is then recursively calculated using convolution. This approach accounts for the probabilistic nature of ISI formation and enables the estimation of the statistical eye diagram. This is given by

$$Z_{k+1}(\tau, t) = \begin{cases} \frac{\delta(\tau) + \delta(\tau - y(t - kT))}{2} * Z_k(\tau, t), & k \neq 0. \\ Z_k(\tau, t), & k = 0. \end{cases} \quad (3.7)$$

where Z is recursively calculated from $k = -N_{post}$ to $k = N_{pre}$ while N_{post} is the amount of post-cursors duration in SBR and N_{pre} the amount of pre-cursors duration in SBR. Initial condition is $Z_{-\infty}(\tau, t) = \delta(\tau)$. At first glance, this 2-dimensional probability equation is obscure to understand, but it will be bright once knowing all the parameters and the convolution process.

First of all, $\delta(\tau)$ is the delta function which represents the probability of voltage level τ at every time step. When k is positive, $y(t - kT)$ is shifted k^{th} unit interval (UI) rightwards of $y(t)$ and adds up the pre-cursors effects to the origin. When k is negative, $y(t - kT)$ takes care of post-cursors ISI effects into account. The first term of convolution in Equation (3.7) indeed is two delta functions with equal probability $1/2$, as there is equal chance of

transmitting bit 0 and bit 1. $\delta(\tau)$ is convolved and no voltage is changed as transmitting bit 0. $\delta(\tau - y(t - kT))$ is convolved and the effect of transmitting bit 1 is statistically considered. After recursive convolution, all pre-cursors and post-cursors effects from transmitted bit 0 and bit 1 are represented by the PDF function.

To be more specific, let us take a simple example in Figure 3.10. The main sampling point is called t_0 and there is one pre-cursor located at t_{-1} and two post-cursors at t_{+1} and t_{+2} . The corresponding δ function for them is illustrated in Figure 3.11, where the x-axis is the voltage level and the y-axis is the probability with equal probability $1/2$. Explicitly, the δ functions at

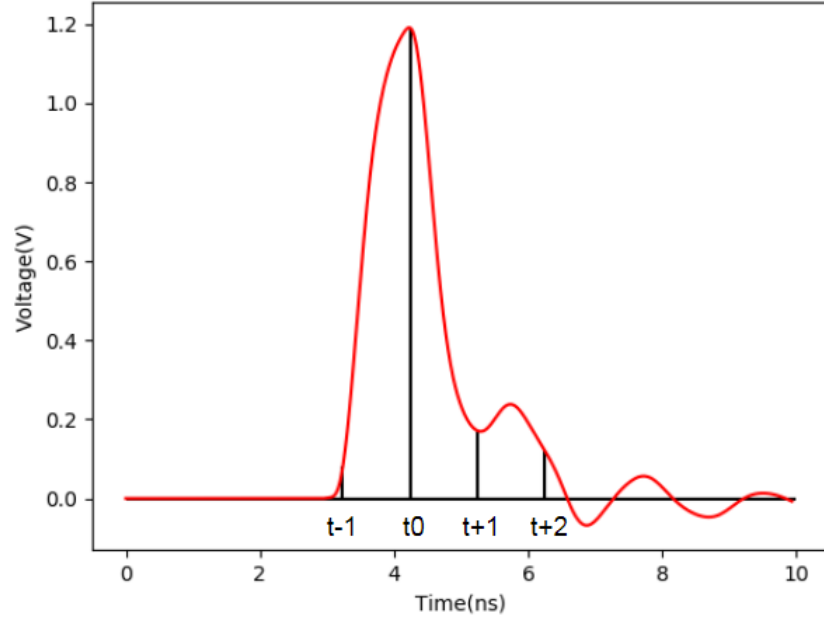


Figure 3.10: Single bit response with 1 pre-cursor and 2 post-cursors.

those 4 time points are $\frac{1}{2}\delta(0) + \frac{1}{2}\delta(0.1)$, $\frac{1}{2}\delta(0) + \frac{1}{2}\delta(1.2)$, $\frac{1}{2}\delta(0) + \frac{1}{2}\delta(0.18)$ and $\frac{1}{2}\delta(0) + \frac{1}{2}\delta(0.15)$. Then, the convolution between the first two delta functions is

$$\begin{aligned} & \left(\frac{1}{2}\delta(0) + \frac{1}{2}\delta(0.1)\right) * \left(\frac{1}{2}\delta(0) + \frac{1}{2}\delta(1.2)\right) \\ &= \frac{1}{4}\delta(0) + \frac{1}{4}\delta(0.1) + \frac{1}{4}\delta(1.2) + \frac{1}{4}\delta(1.3) \quad (3.8) \end{aligned}$$

The second convolution is between Equation (3.8) and the third delta func-

tion:

$$\begin{aligned}
& \text{Equation(3.8)} * \left(\frac{1}{2}\delta(0) + \frac{1}{2}\delta(0.18) \right) \\
&= \frac{1}{8}\delta(0) + \frac{1}{8}\delta(0.1) + \frac{1}{8}\delta(1.2) + \frac{1}{8}\delta(1.3) + \\
& \quad \frac{1}{8}\delta(0.18) + \frac{1}{8}\delta(0.28) + \frac{1}{8}\delta(1.38) + \frac{1}{8}\delta(1.48) \quad (3.9)
\end{aligned}$$

Lastly, the third convolution is performed between Equation (3.9) and the fourth delta function that generates $2^4 = 16$ terms.

$$\begin{aligned}
& \text{Equation(3.9)} * \left(\frac{1}{2}\delta(0) + \frac{1}{2}\delta(0.15) \right) = \frac{1}{16}\delta(0) + \frac{1}{16}\delta(0.1) + \frac{1}{16}\delta(1.2) + \frac{1}{16}\delta(1.3) + \\
& \quad \frac{1}{16}\delta(0.18) + \frac{1}{16}\delta(0.28) + \frac{1}{16}\delta(1.38) + \frac{1}{16}\delta(1.48) + \\
& \quad \frac{1}{16}\delta(0.15) + \frac{1}{16}\delta(0.25) + \frac{1}{16}\delta(1.35) + \frac{1}{16}\delta(1.45) + \\
& \quad \frac{1}{16}\delta(0.33) + \frac{1}{16}\delta(0.43) + \frac{1}{16}\delta(1.53) + \frac{1}{16}\delta(1.63) \quad (3.10)
\end{aligned}$$

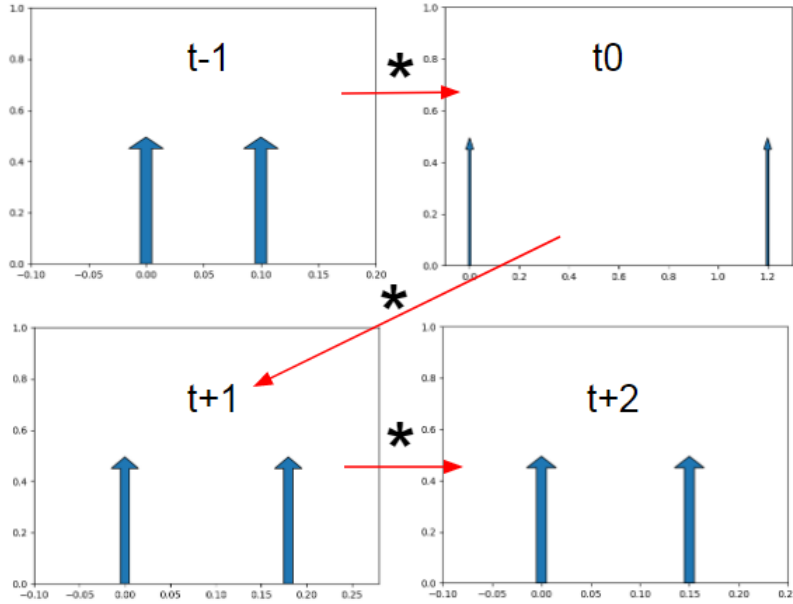


Figure 3.11: Individual PDF for 4 cursors.

After taking all the convolution, the resulting PDF given the main sampling time t_0 is shown in Figure 3.12, described by Equation (3.10). There are 16 delta functions due to 4 cursors in total and each of them contributes 2 voltage levels. Each probability is $1/16$. In the case of N cursors, the total

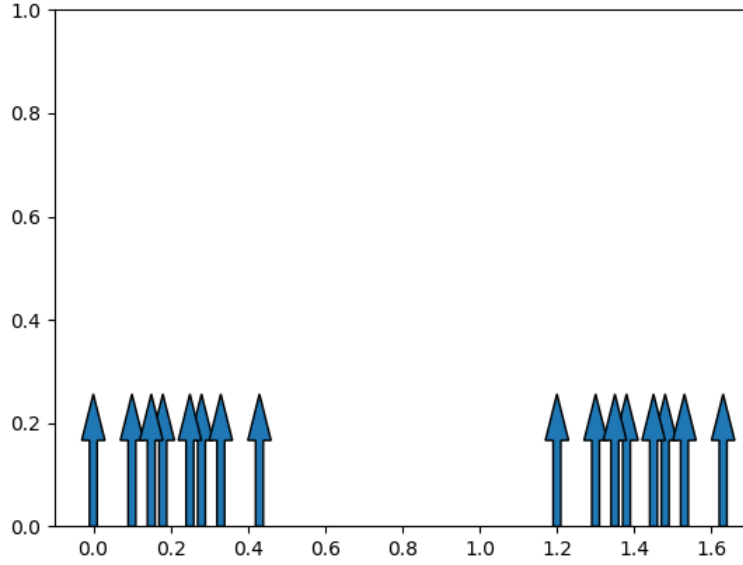


Figure 3.12: Convolution resultant PDF.

delta functions will be 2^N and each δ has a magnitude of $1/N$. The Equation (3.10) shown in Figure 3.12 is $Z(\tau, t = t_0)$ from Equation (3.7) when the initial sampling time is t_0 . Now, transferring the PDF in Figure 3.12 to voltage versus time plot in Figure 3.13, at t_0 , multiple ISI voltage levels (displayed as dots) and the probability of each dot is known.

The next step is to select a different sampling time point to t_1 and then repeat the same delta function convolution procedure. The other group of ISI voltage levels (dots) and its probability density can be identified as shown in Figure 3.14. The complete eye diagram has time sweeping over 2 UIs, so the range of sampling time t_0 to t_n should be 2 UIs. The number n describes the time resolution of the eye diagram. The larger the n , the more time points with higher density would be in the eye diagram horizontal direction. Similarly, the ISI voltage level (dots) describes the voltage resolution of the eye diagram. More dots (2^N) there are, more compact and well-defined in the eye diagram vertical direction. In Figure 3.15, 20 time points are chosen to sweep over the 2 UIs region of the single-bit response. For each time point, there is one pre-cursor and two post-cursors, so the number of ISI superpositions is $2^4 = 16$ including the main cursor itself. The corresponding 16 ISI levels are plotted on top of the response as well. Not only do those dots represent the ISI voltage levels, but each dot also contains the probability density value information. If there is a z-axis out of the page, it would be the

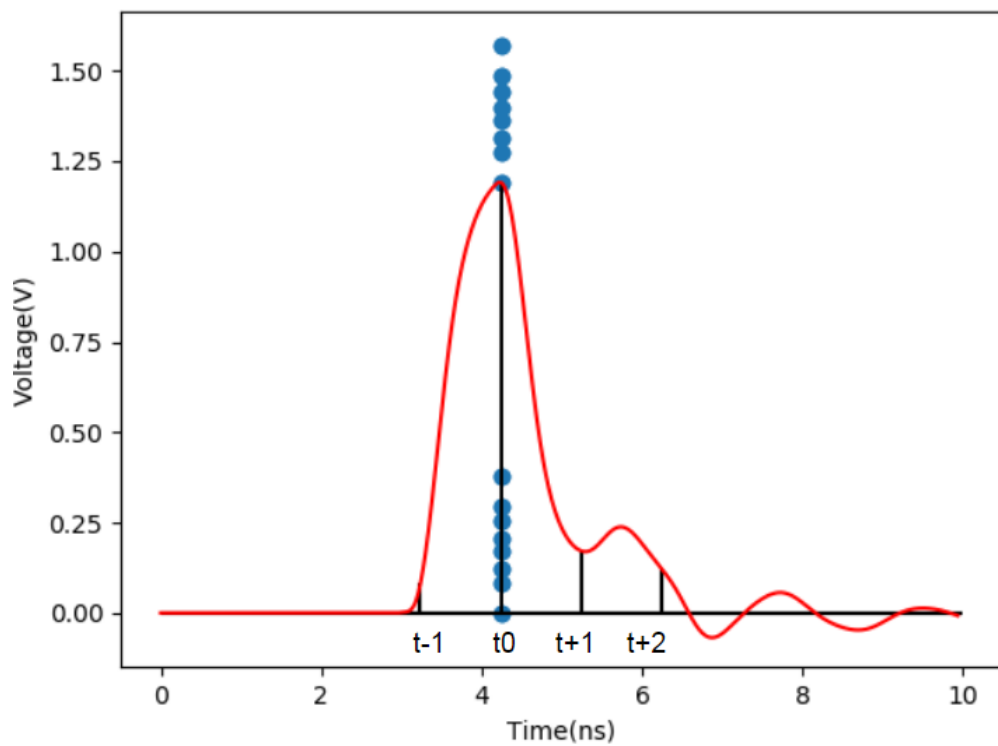


Figure 3.13: ISI combination at t_0 .

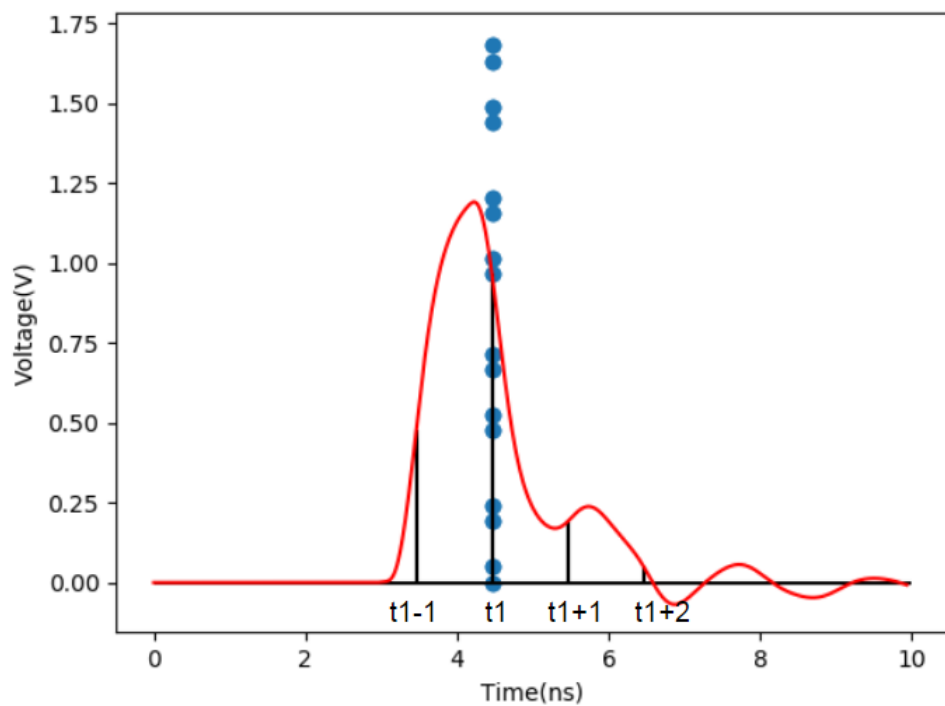


Figure 3.14: ISI combination at t_1 .

magnitude of probability. From this plot, the eye shape is gradually formed as there are more dots surrounding the top and bottom location, leaving the center region spare and open. Figure 3.16, Figure 3.17 and Figure 3.18 show the eye diagram when more cursors are taken into consideration and the eye becomes much denser, as expected.

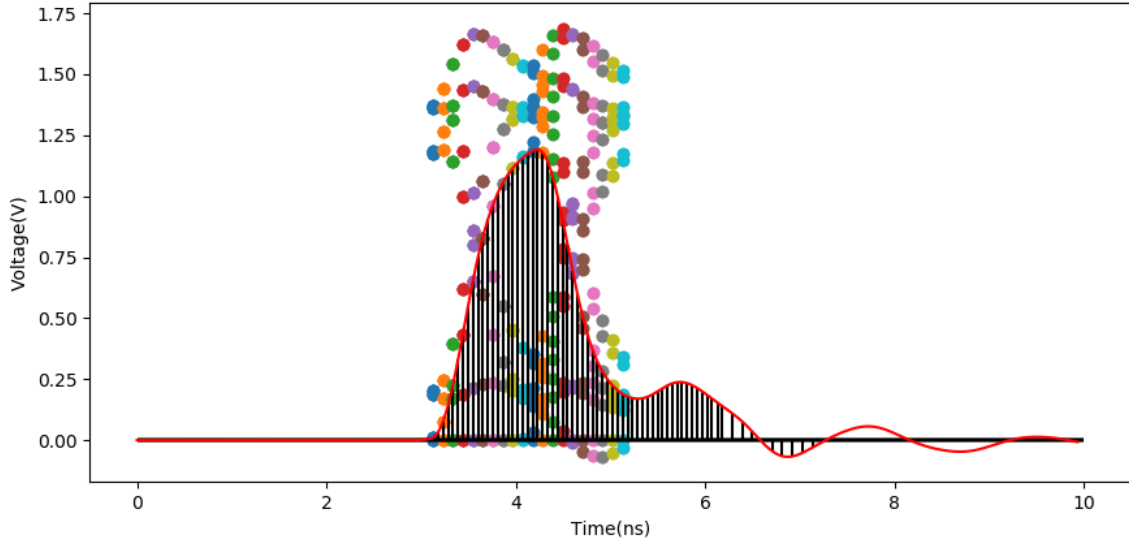


Figure 3.15: ISI combination from t_0 to t_{20} .

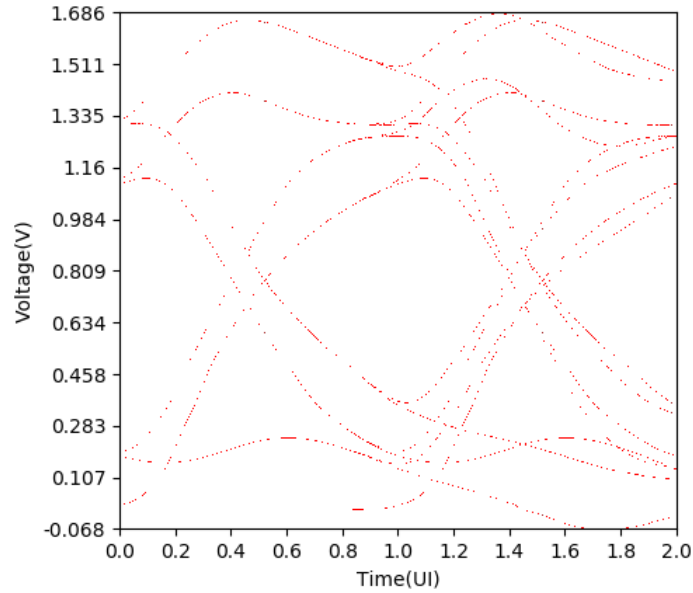


Figure 3.16: Eye diagram from 1 pre-cursor and 2 post-cursors.

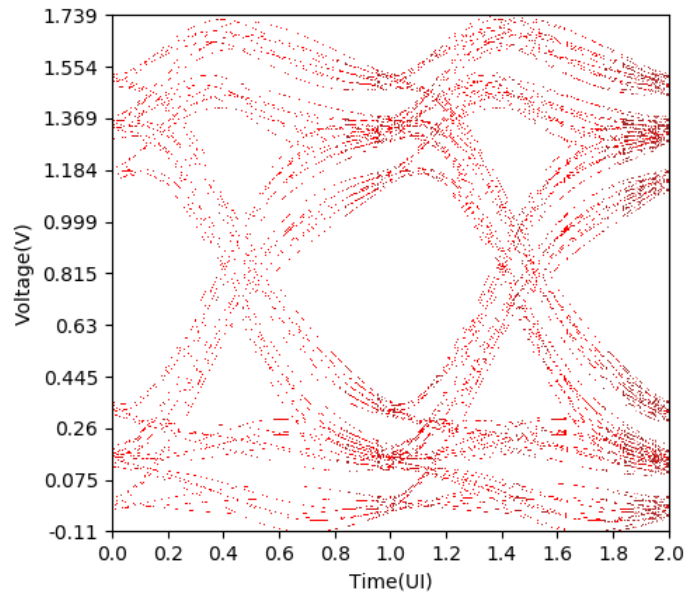


Figure 3.17: Eye diagram from 2 pre-cursors and 5 post-cursors.

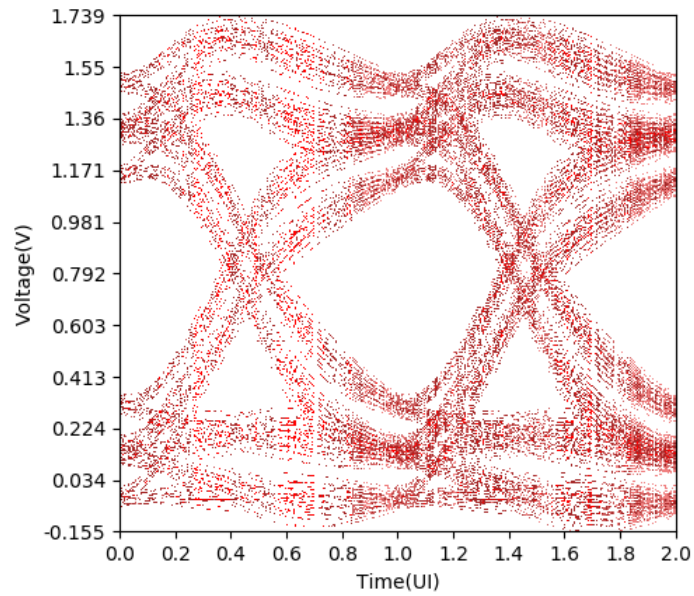


Figure 3.18: Eye diagram from 3 pre-cursors and 7 post-cursors.

In the statistical analysis, as shown above, the eye diagram is related to the number of cursor selections. The number of cursor selections can be also related to the number of bit patterns running in the transient simulation. For example, in the case where the pulse response only has a maximum of 3 post cursors, when all 3 cursors are selected in statistical analysis, it can represent the situation of $2^3 = 8$ bit patterns: 000, 001, 01, 011, 100, 101, 110, 111, send into the transient simulation. However, in reality, the pulse response always experiences multiple post cursors or ISI, therefore the more cursor selections exist the more bit patterns and stressful waveforms are included. Figure 3.19 illustrates that as the number of cursors increases, both the eye width and eye height exhibit a gradual decrease and eventual convergence. This trend persists until the additional cursor's influence on the ISI effect becomes negligible. In the pulse response example in Figure 3.19, the convergence occurs after 12 cursors. This implies that when $2^{12} = 4096$ identical bit patterns are run, the resulting eye diagram from transient simulation (eye width = 0.595 UI, eye height = 165.7 mV) aligns with the statistical eye diagram.

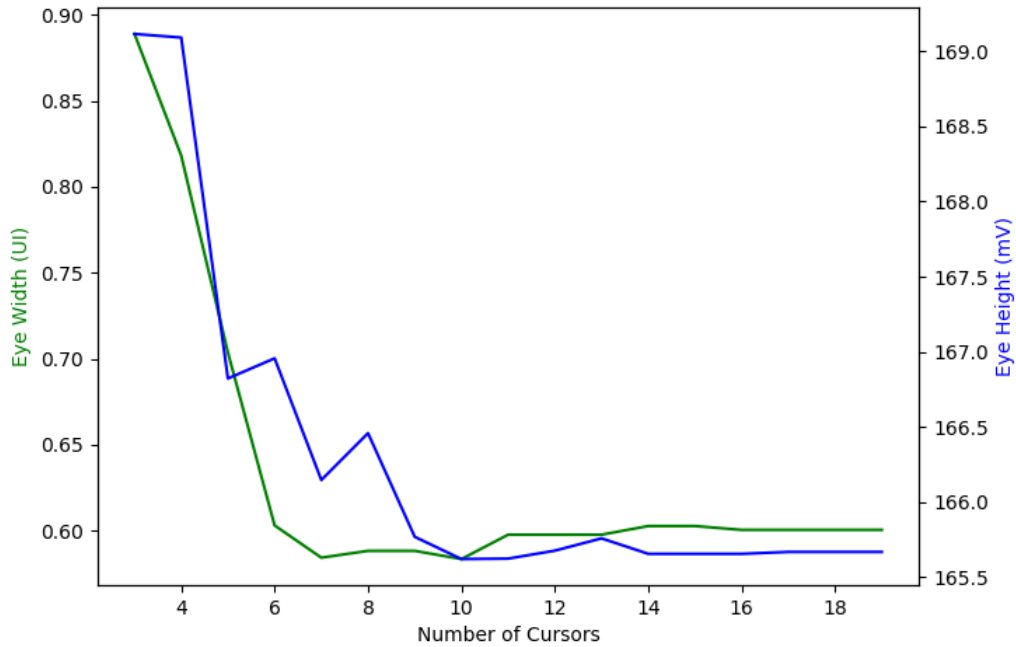


Figure 3.19: Eye height and eye width convergence.

CHAPTER 4

STATISTICAL HIGH-SPEED LINK METHOD IN NONLINEAR SYSTEM

4.1 Introduction

The eye diagram is of paramount importance in high-speed link systems, as it serves as a visual representation of the signal quality and integrity. By analyzing the characteristics of the eye diagram, signal integrity (SI) engineers can identify and diagnose various SI issues such as intersymbol interference (ISI), crosstalk, timing violations, and amplitude distortions. During the design phase of high-speed links, engineers also use the eye diagram to validate and verify the performance of components such as transmitters, receivers, interconnects, and channels. This ensures that the designed system meets the required performance criteria before production. Therefore, accurate eye diagram estimation is the foundation of all next-step evaluations.

The transient time domain simulation approach can achieve accurate simulation results; however, the time required for simulation increases in proportion to the length of the input bit sequence. For a complex system nowadays, a standard low BER like 10^{-12} to 10^{-18} is typically required to evaluate the performance. In the transient simulation method, such BER analysis is computationally heavy with at least 10^{12} to 10^{18} input bits transmitted into the system. To remedy lengthy bit sequence simulations, various methods of fast and accurate eye diagram estimations have been proposed [19, 34, 35], the majority of which are based on superposition and LTI assumptions.

The SBR [19] is widely used to estimate the worst-case eye diagram and perform bit-by-bit convolution simulation based on LTI system assumption. However, due to asymmetric input/output (I/O) buffers or mismatches between pull-up and pull-down drivers, the rising and falling edge responses are different. Therefore, the double-edge response (DER) approach is introduced in [34] with limited nonlinearity consideration. To overcome this limitation,

the multiple-edge response (MER) [35, 38] considers the impact of previous multiple bits on the current bit to completely cover the nonlinear effects on the edge response. The second approach is through statistical perspective simulation to the eye diagram directly without the system output waveform requirement, which in general is even faster than the time-domain approach. In [19], the concept of statistical eye diagram is first used for preliminary estimation of an eye diagram, which does not require extensive input sequence in time-domain simulation, assuming the LTI system with the use of superposition.

However, as previously stated, with the fast growth in high-speed systems and the increasing circuit complexity over the past decades, assuming linearity and using the superposition concept in high-speed signaling systems underestimates the impact of nonlinear effects in circuits. Therefore, the current conventional fast simulation method is no longer valid for analyzing high-speed signaling systems with nonlinear components. To properly predict eye diagrams with nonlinear components in a fast and efficient path, there are several approaches [15–18, 41–46]. One approach uses the Volterra series to decompose a nonlinear network into multiple linear systems and then applies the peak distortion analysis to estimate the worst-case eye margin [15]. Next, the nonlinear network is represented by the Hammerstein or Wiener models, and then the eye diagram is still analyzed through the bit sequence transient simulation [16], which hinders the simulation speed. In [17, 41], a matrix is created from all combinations of nonlinear responses and a low-rank matrix representation is found to reduce the number of nonlinear responses needed to complete the eye diagram for nonlinear circuits and channels. However, the $\mathcal{O}(\cdot)$ time complexity to find a matrix is relatively high. In [42], the nonlinear package system statistical eye diagram is performed by multiple short-bit patterns running in parallel to overcome the intensive runtime, but the amount of short-bit patterns requirement is unknown. The eye diagram accuracy of a nonlinear buffer can also be improved by an analytically modeled closed-form equation in [43] for a specific receiver. The majority uses the advanced MER method to estimate the nonlinear system eye diagram by linear superposition. The method in [18] proposes m th order multiple edge responses beyond the single bit response, following the convolution approach in [19] to calculate the statistical eye diagram. In [44–46], the statistical convolution procedure is separated into linear and nonlinear sections to save

up the simulation time. While these MER methods have significantly enhanced the precision and effectiveness of nonlinear signaling analysis, their applicability can still be constrained by their reliance on the linear superposition concept. In addition to the traditional methods mentioned earlier, machine learning techniques are extensively utilized in modeling high-speed link systems [9, 12, 13] to address the runtime limitations of full-waveform simulations. However, it is important to note that these methods do not directly account for the impacts of nonlinear factors.

In a brief overview of current methods, either the lengthy bit sequence transient simulation is time-demanding or the majority of fast and efficient methods for analyzing eye diagrams are designed for LTI systems with nonlinearity trade-off. To tackle these limitations, a new statistical model of the nonlinear system using the Hammerstein-Wiener model, extended from [47], utilizes probability density functions with random variable transformations to statistical eye diagrams directly. This statistical approach offers eye diagram simulation in nonlinear systems a new dimension characterized by enhanced accuracy and efficiency.

This chapter is organized as follows. Section 4.2 solves the nonlinear system using the Wiener model and the proposed statistical estimation method is introduced in detail. In Section 4.3, the results of the statistical eye diagram prediction in various examples are verified and compared against those obtained from transient simulation. Lastly, it concludes by summarizing the findings and outlining potential future research directions.

4.2 Nonlinearity Modeling

As integrated circuit density and data rates continue to increase in high-speed links, nonlinear behaviors become increasingly prominent [48]. Therefore, accurate and efficient modeling of nonlinear circuits is essential. Despite the preference for linearity in high-speed link systems, transmitters (TX) and receivers (RX) still exhibit nonlinear behaviors, posing significant challenges for simulation. In the simplest approach, the TX and RX system can be treated as an LTI system. This involves applying LTI concepts directly to the nonlinear circuitry, often by assuming weak or negligible nonlinearity. However, the error will occur and the result will differ from the expected

measurement result. The alternate way is through multilinear theory [49]. The nonlinear network is decomposed into multiple linear networks. More popular approaches are done by block-oriented nonlinear models [50], which are created using two basic building blocks: an LTI block and a static nonlinear block. There are various ways to construct the model in Figure 4.1. The model of nonlinearity proposed here is the Wiener model derived from the Volterra series [51]. It is chosen due to its capability to simplify the nonlinear system into a cascaded combination of linear dynamic sections followed by nonlinear static sections. This flexibility meets the requirements for statistical analysis and aids in capturing the complex behavior of the system.

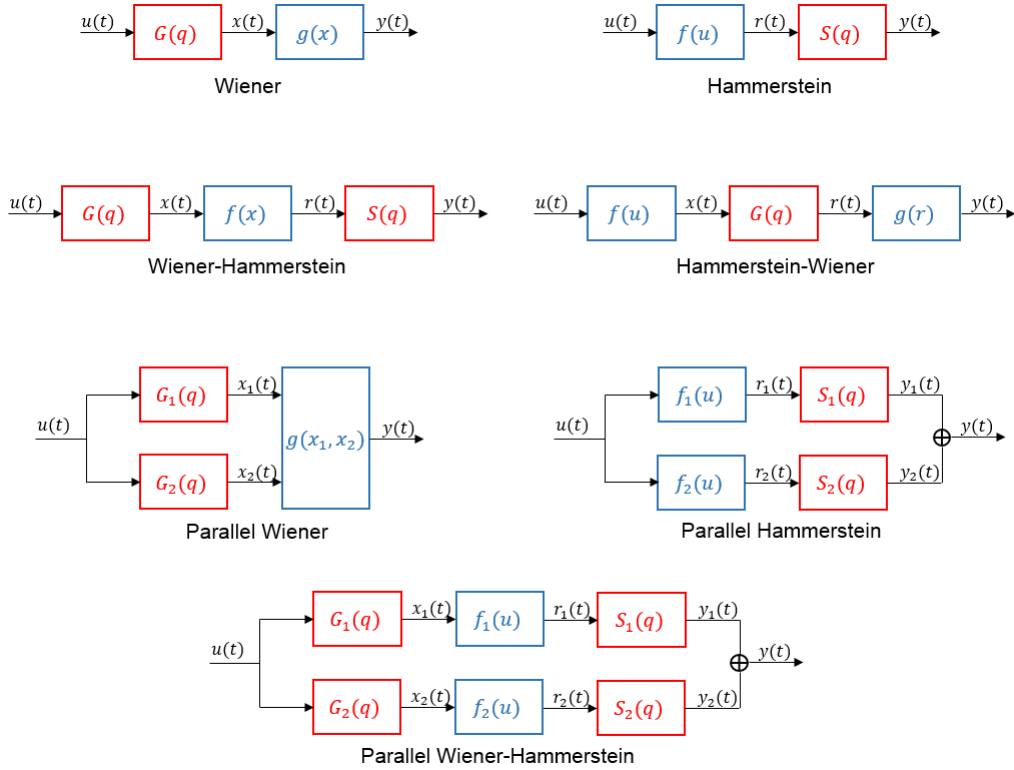


Figure 4.1: The different block-oriented structures, connected by LTI blocks $G(q)$ and $S(q)$ and static nonlinear blocks $f(\cdot)$ and $g(\cdot)$.

4.2.1 Volterra-Wiener Model System

The Volterra-Wiener model is a mathematical framework used to describe and analyze the behavior of nonlinear systems [52]. It extends the classical linear Wiener system model by incorporating higher-order nonlinearities

through the use of the Volterra series. In the Wiener model, the output of a system is expressed as a convolution of the input signal with a linear impulse response function. However, in the Volterra-Wiener model, the system's output is described as a series of convolutions of the input signal with multiple nonlinear impulse response functions, known as Volterra kernels. Wiener considered representing the output, $y(t)$, of a nonlinear system as a functional series expansion of its input, $u(t)$ [53]. Mathematically, the Volterra series expression of the output $y(t)$ is:

$$y(t) = h_0 + \sum_{n=1}^m H_n[u(t)] \quad (4.1)$$

in which

$$H_n[u(t)] = \int_1^m \dots \int_1^m h_n(\tau_1, \dots, \tau_n) u(t - \tau_1) \dots u(t - \tau_n) d\tau_1 \dots d\tau_n. \quad (4.2)$$

In these equations, H_n is a Volterra operator and $h_n(\tau_1, \dots, \tau_n)$ is called the Volterra kernel of the n th order model approximation. The Volterra series expression can be also viewed as a polynomial series with memory by introducing the input gain factor, α , so that the new system output expression, as the Volterra-Wiener system, is then seen to be

$$y(t) = \sum_{n=0}^m \alpha_n x^n(t) = g(x(t)) \quad (4.3)$$

which now is a polynomial system of degree m , represented by $g(\cdot)$. The system is assumed to be stationary linear, and $x(t)$ is the output result of a convolution by an LTI system, represented by a transfer function

$$G(q) = \frac{b_m(q - z_1)(q - z_2) \dots (q - z_m)}{a_n(q - p_1)(q - p_2) \dots (q - p_n)} \quad (4.4)$$

where z_i are the zeros and p_i are the poles. Therefore, for any nonlinear system, the goal is to recover the system representation parameters in $G(q)$ and the polynomial coefficients α_n through the Wiener system identification using the maximum likelihood method.

In order to estimate the unknowns, the Wiener model is defined in Figure

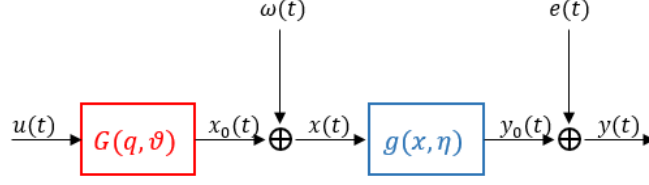


Figure 4.2: The Wiener model, where the input $u(t)$ and the output $y(t)$ are measurable, but not the intermediate signal $x(t)$.

4.2. The input $u(t)$ and the output $y(t)$ pairs are the only known signal and $x(t)$ denotes the unmeasurable intermediate signal. $w(t)$ and $e(t)$ are mutually independent process noise terms that are in use of estimating the unknown parameters ϑ in the LTI block $G(\cdot)$ and the unknown parameters η in the static nonlinear block $g(\cdot)$. For convenience, a joint parameter vector θ is defined as

$$\theta = [\vartheta^T, \eta^T]^T \quad (4.5)$$

to describe the unknown intended to be found.

The Maximum Likelihood Method (MLM) is a statistical technique used to estimate the parameters θ of a model by maximizing the likelihood function, based on the observation data set $Y_n = \{y_1, y_2, \dots, y_n\}$ and the input data set $U_n = \{u_1, u_2, \dots, u_n\}$. The likelihood is denoted by $p_\theta(Y_N)$:

$$p_\theta(Y_N) = \left(\frac{1}{2\pi\sqrt{\lambda_e\lambda_w}}\right)^N \prod_{t=1}^N \int_{-\infty}^{\infty} e^{-\frac{1}{2}\epsilon(t,\theta)} dx(t), \quad (4.6)$$

where

$$\epsilon(t, \theta) = \frac{1}{\lambda_e} (y(t) - f(x(t), \eta))^2 + \frac{1}{\lambda_w} (x(t) - G(q, \vartheta)u(t))^2, \quad (4.7)$$

in which the process noise $\omega(t)$ and $e(t)$ are assumed in Gaussian distribution with zero means and variances λ_w and λ_e . The Maximum-Likelihood (ML) estimate is obtained by

$$\hat{\theta} = \arg \max_{\theta} p_\theta(Y_N). \quad (4.8)$$

Therefore, the likelihood and its gradient are defined and the targeted θ parameters can be calculated by the either direct gradient-based search approach or the Expectation-Maximization (EM) approach.

4.2.2 Random Variable Transformation

Through the Volterra-Wiener model construction of the nonlinear system, the statistical eye diagram can be identified efficiently, as the flowchart displayed in Figure 4.3, in which $u(t)$ is a single-bit input, $y(t)$ is a single-bit output of the nonlinear system, and $f_Y(y, t)$ is the nonlinear statistical eye diagram to be found. Once the LTI block has been identified, the SBR statistical analysis is applied to first find the intermediate statistical eye diagram $f_X(x, t)$. A polynomial function represents the static nonlinear block as

$$g(x) = \sum_{n=0}^m \alpha_n x^n, \quad (4.9)$$

where t in $x(\cdot)$ is dropped due to the stationary system. This nonlinear polynomial function defines the statistical relationship or stochastic transformation between the input and the output of the static nonlinear block so that the nonlinear statistical eye diagram can be calculated as $f_Y(y, t)$. Suppose X , being a random variable with the entries distributed accord-

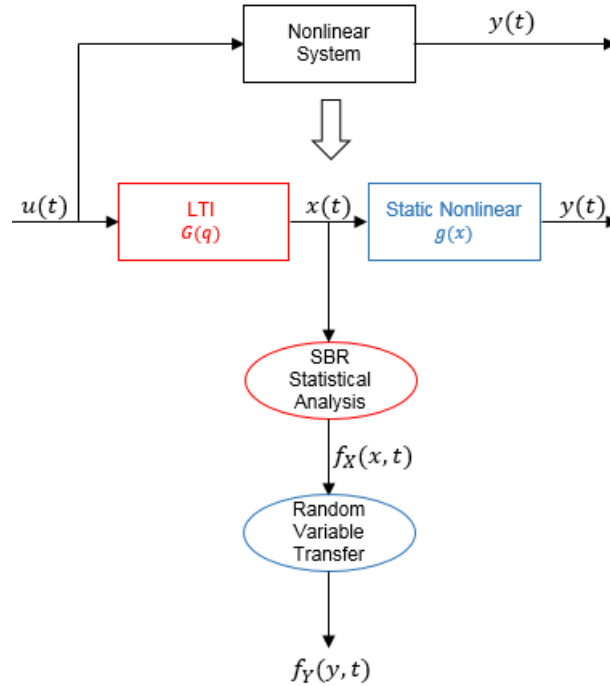


Figure 4.3: The flowchart of the proposed method through the random variable transformation.

ing to $f_X(x, t)$, represents the SBR output of a comprehensive LTI system

and the input of the nonlinear system equivalently. Then, $y = g(x)$, where $g : \mathbb{R} \rightarrow \mathbb{R}$, is the output of a nonlinear mapping from the input X . From the transformation of the random variable [54], the PDF of y can be calculated as

$$f_Y(y, t) = \sum_i \frac{f_X(x_i, t)}{|g'(x_i)|}, \quad (4.10)$$

where x_i is the i th root of $y = g(x)$ and $g'(x)$ denotes the derivative of $g(x)$. For a monotonic function $g(x)$, which is usually the case in the high-speed link, the Equation (4.10) can be simplified as

$$f_Y(y, t) = \frac{f_X(x, t)}{|g'(x)|}. \quad (4.11)$$

Consequently, the statistical eye diagram following the nonlinear system can be represented as $f_Y(y, t)$, achieved through a one-to-one function mapping $g(x)$ that is either monotonic increasing or decreasing. Figure 4.4 shows how the voltage density $f_X(x)$ of the intermediate signal x transfers to the output voltage density $f_Y(y)$ at one instant time step. After mapping all input densities within the interval $t \in [0, 2T]$ through the static nonlinear block, the resulting 2D PDF statistical eye diagram is obtained.

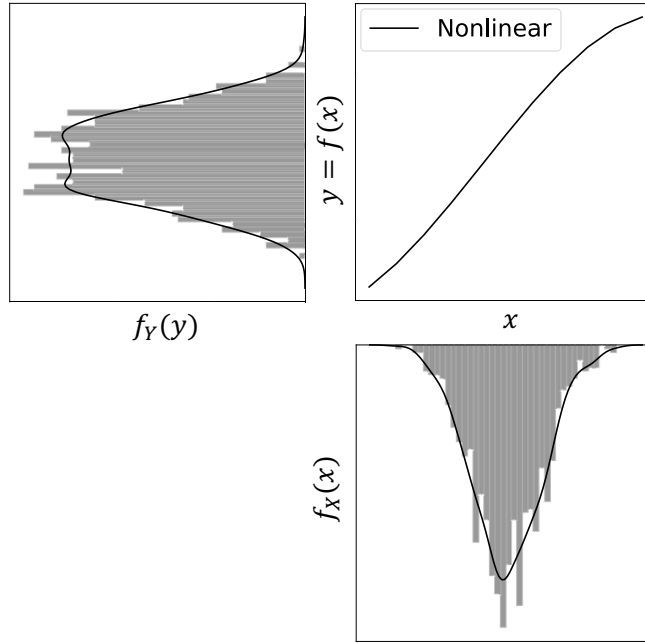


Figure 4.4: Nonlinear transformation of a standard normal random variable.

4.3 Validation of the Proposed Method

To validate the proposed method, three examples are given. The first one is a theoretical Wiener model system, constructed by a high-speed channel as the LTI block and a voltage-controlled-voltage-source (VCVS) as the static nonlinear block. The second example is a 10nm technology Fin Field-Effect Transistor (FinFET) differential buffer for a pair of Bunch of Wires (BoW) physical standard [55, 56] at 32 Gbps and 64 Gbps. The last example is a FinFET high-speed link system with a buffer receiver and a receiver with continuous time linear equalization (CTLE) and decision feedback equalization (DFE) at 10 Gbps.

4.3.1 Mathematical Model

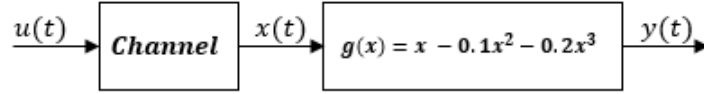


Figure 4.5: The mathematical model of a Wiener system.

The ideal Wiener model system is proposed in Figure 4.5 with an LTI channel combining a nonlinear receiver. The static nonlinear part of the receiver, as representing by a nonlinear VCVS in the design, is mathematically expressed as:

$$y = g(x) = x - 0.1x^2 - 0.2x^3 \quad (4.12)$$

where x is the input of a nonlinear component in the range of $[-1, 1]$ and y is the output of a nonlinear component with nonlinearity in Figure 4.6.

Suppose $x(t)$ is the channel pulse response of a single-ended input bit with $V_{low} = 0V$ and $V_{high} = 0.75V$, and the intermediate pulse in the Wiener model, shown in Figure 4.7 (a), and through the $g(x)$, the output $y(t)$ as nonlinear pulse in Figure 4.7 (b) is obtained. If the direct statistical analysis method discussed in Chapter 3.4 with the selection of 5 pre-cursors and 50 post-cursors is applied regardless of the nonlinearity, the corresponding statistical eye diagrams from $x(t)$ and $y(t)$ are displayed in Figure 4.7 (c) and (d). Figure 4.7 (d) is the statistical eye diagram by assuming the $y(t)$ is the output of an LTI system by given input $x(t)$. Nonetheless, the expected eye

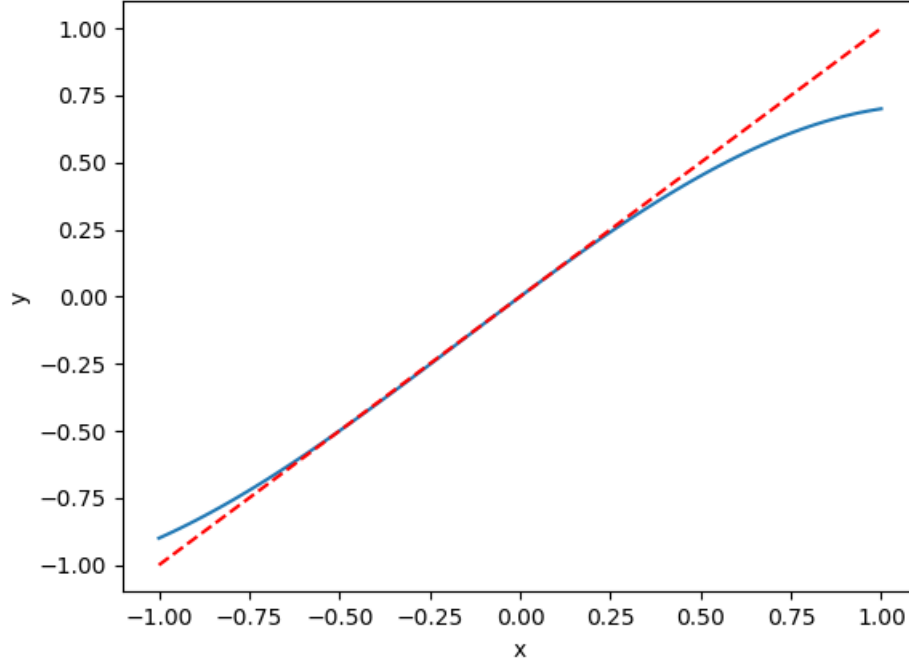


Figure 4.6: Receiver nonlinearity.

diagram, derived from a pseudo-random bit sequence (PRBS) of 10^4 bits and overlaying the 2-bit waveform obtained from the output of the nonlinear block $g(x)$, is depicted in Figure 4.8 (a). This transient eye diagram is significantly different from the eye calculated from the direct statistical analysis applied on $y(t)$ nonlinear pulse in Figure 4.7 (d).

To obtain the correct statistical eye diagram or PDF of output y , the Equation (4.11) is applied:

$$g'(x) = 1 - 0.2x - 0.6x^2$$

$$f_Y(y, t) = \frac{f_X(x, t)}{|1 - 0.2x - 0.6x^2|} \quad (4.13)$$

The input 2D PDF or statistical eye diagram in Figure 4.7 (c) is weighted by $g'(x)$ to the resulting output PDF $f_Y(y, t)$, which is shown in Figure 4.8 (b). The statistical eye diagram generated by the proposed method closely aligns with the transient eye diagram, as observed in Figure 4.8. The eye diagram information like eye height (EH) and eye width (EW) can be calculated directly from the statistical information of the eye diagram. In Figure 4.8 (c), the voltage PDF is presented by central 20% mean values of one level and zero level, and the EH is calculated from the difference between the

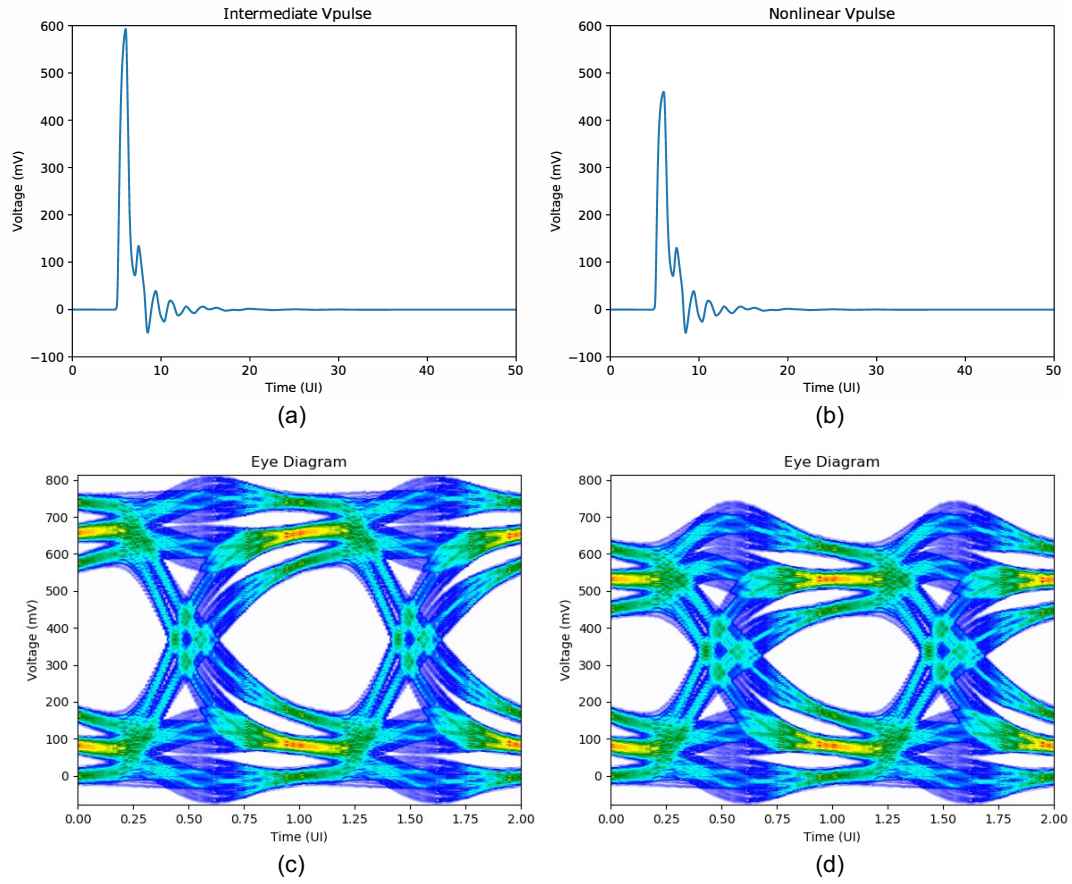


Figure 4.7: The single-ended signal eye diagrams through the direct statistical analysis.

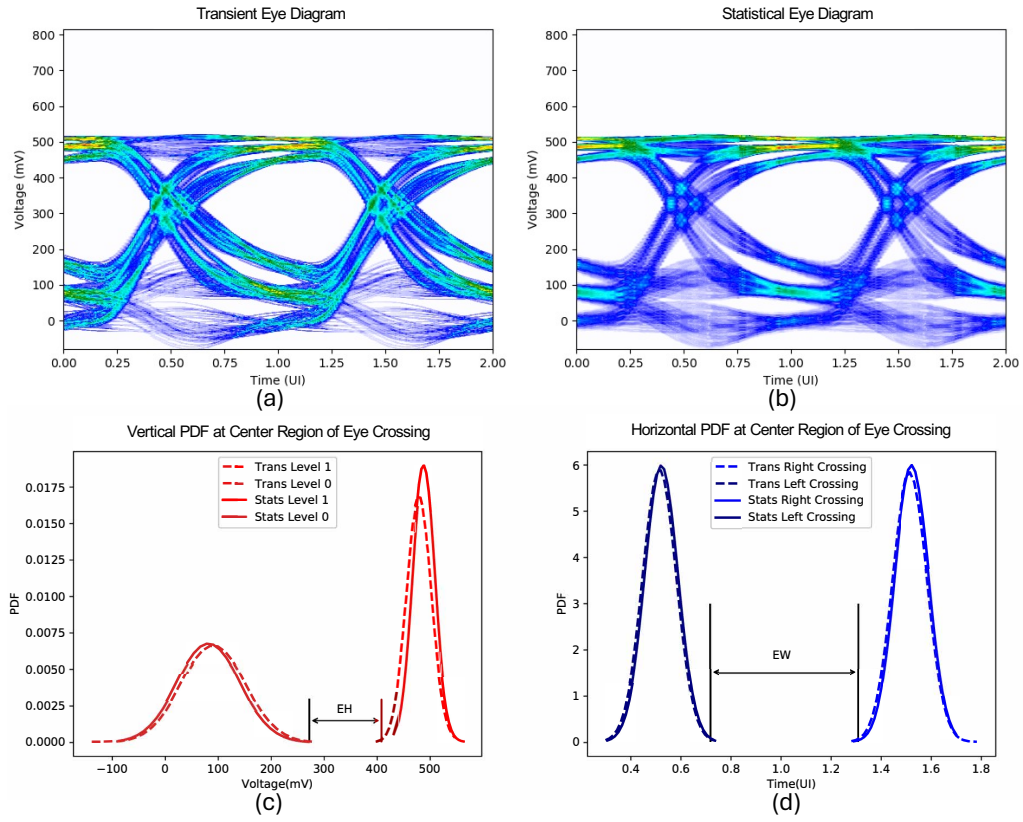


Figure 4.8: The comparison between the transient simulation and proposed nonlinear statistical analysis in single-ended signal.

inner 3σ points of the level 1 and level 0 PDF. Likewise, the EW from time PDF, calculated by averaging the central 10% of the left crossing and right crossing time points, in Figure 4.8 (d) is determined by measuring the interval between the inner 3σ points.

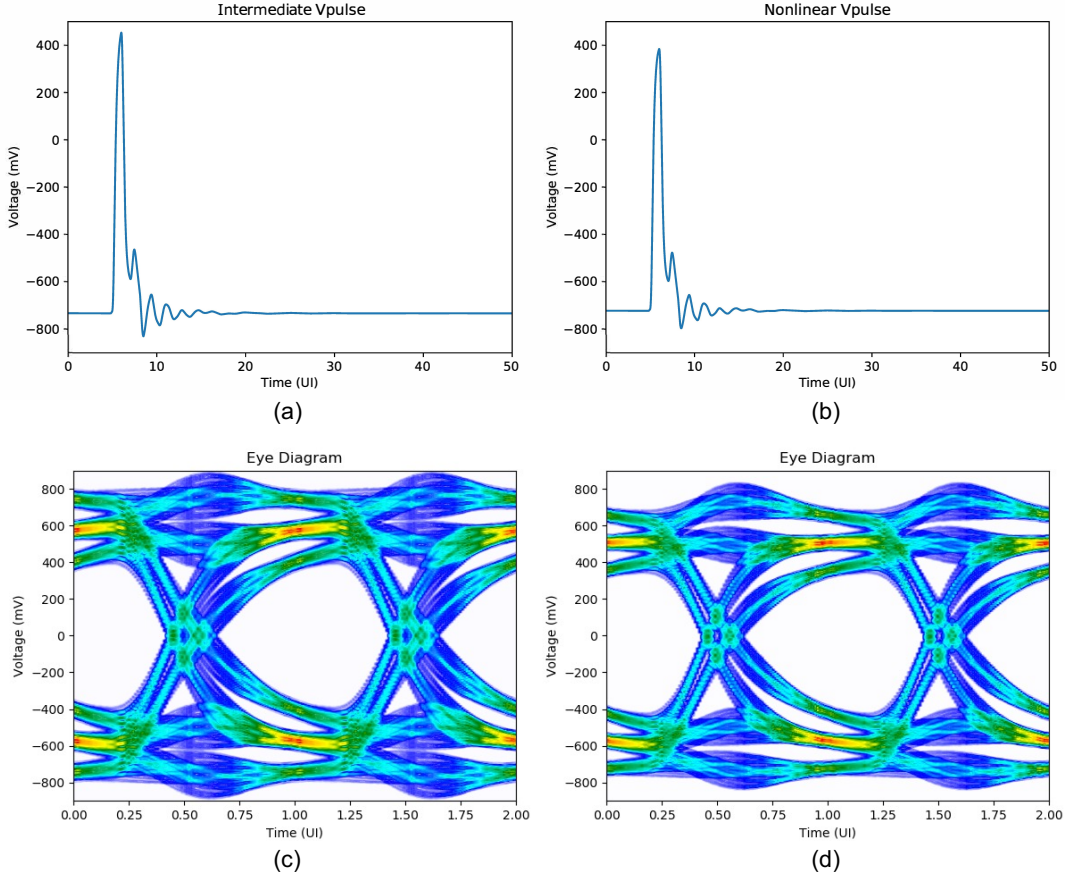


Figure 4.9: The differential signal eye diagrams through the direct statistical analysis.

Now, suppose $x(t)$ is the channel pulse response of a differential input bit with $V_{low} = -0.75V$ and $V_{high} = 0.75V$ shown in Figure 4.9 (a), and through the same nonlinear system $g(x)$, the output $y(t)$ as nonlinear pulse in Figure 4.9 (b) is obtained. By looking at the nonlinearity effects of different voltage levels in Figure 4.6, the higher voltage end experiences more nonlinearity than the lower end. However, the eye in 4.9 (b) does not represent such distortion if applying the statistical analysis directly. The eye diagram comparison in Figure 4.10 (a) and (b) reveals greater distortion at high voltage levels and less distortion at low levels, which are the expected and correct eye diagram simulation results. The corresponding EH and EW are also calculated in

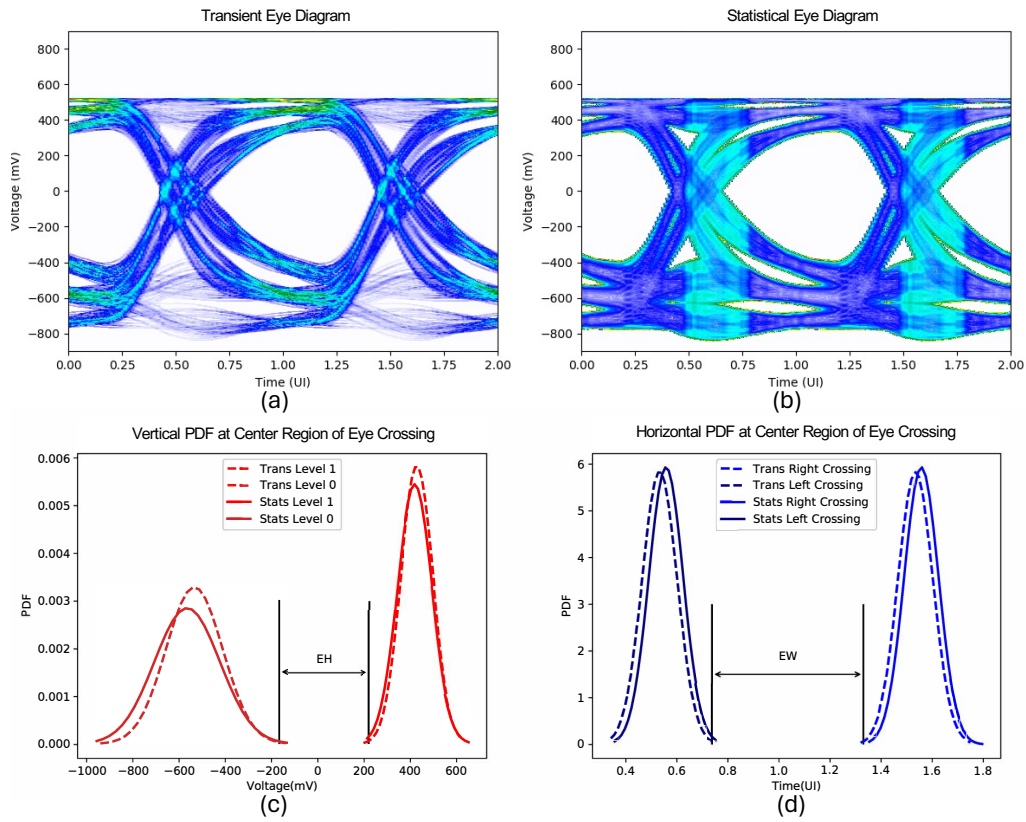


Figure 4.10: The comparison between the transient simulation and proposed nonlinear statistical analysis in the differential signal.

Figure 4.10 (c) and (d). The eye diagram information of the mathematical Wiener model is summarized in Table 4.1, demonstrating a close estimation result of the statistical eye with a small deviation from the transient eye. In terms of simulation runtime, the CPU time for statistical analysis only costs roughly 40% for a single-ended signal and 60% for a differential signal of the CPU runtime in SPICE transient simulation. The runtime efficiency in statistical analysis is directly proportional to the amount of identical voltage levels. In this instance, the single-ended signal comprises 895 voltage levels within the range of $[-100, 500]$ mV. Conversely, with differential signaling, there are 1800 distinct voltage levels spanning $[-800, 800]$ mV, doubling the range of the single-ended signal. Consequently, the runtime is also doubled from 2.75 seconds to 4.82 seconds.

Table 4.1: Comparison between the transient and nonlinear statistical simulation methods in numerical Wiener model example.

	Single-Ended			Differential		
	EH (mV)	EW (UI)	Time (s)	EH (mV)	EW (UI)	Time (s)
Transient	169.3159	0.6005	7.42	354.7546	0.5966	7.86
Statistical	165.6665	0.5922	2.75	344.2349	0.5928	4.82
Error	-2.37%	-1.38%	–	-2.82%	-0.64%	–

4.3.2 Differential FinFET Buffer

Initially, nonlinearity modeling is conducted by decomposing the buffer system in Figure 4.11 into the Wiener model, consisting of one LTI part and one nonlinear part. A single-bit input pulse $u(t)$ is then applied to the buffer system to acquire the corresponding channel output response $y(t)$, which helps to find unknowns in the LTI block and polynomial coefficients through the Maximum Likelihood statistical technique. Once the LTI system is characterized, the intermediate pulse $x(t)$ is readily obtained through the convolution of $u(t)$ and the transfer function, alongside the intermediate statistical eye diagram $f_X(x, t)$ via direct statistical analysis. Subsequently, utilizing the known polynomial function $g(x)$ and applying the random variable transformation, $f_Y(y, t)$ is naturally calculated.

For this test, a differential signal is transmitted to the 10 nm FinFET

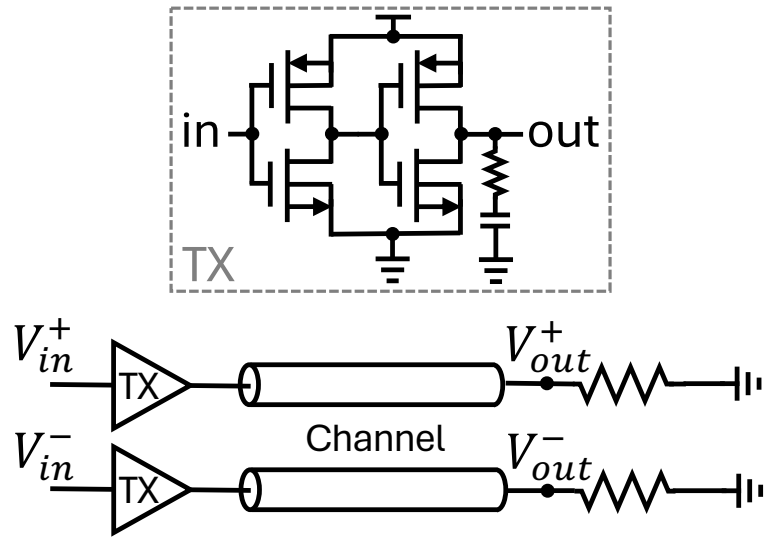


Figure 4.11: The schematic of differential FinFET buffer system.

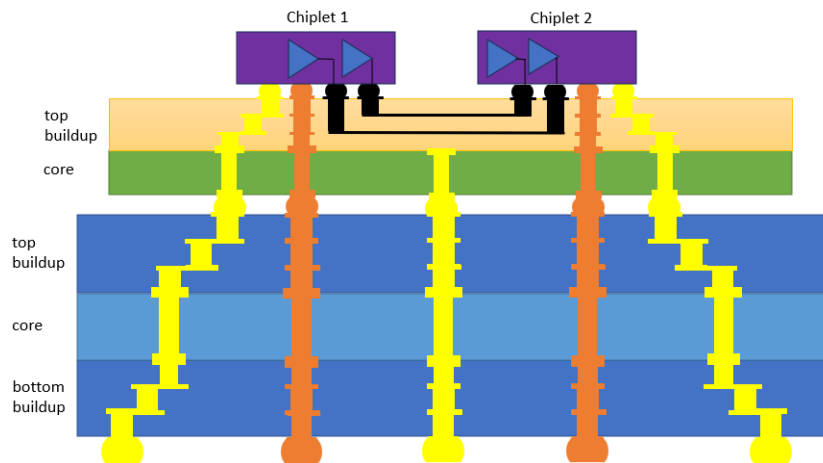


Figure 4.12: 2.5D package model with two chiplets.

buffer in 32 Gbps and 64 Gbps to validate the eye diagram in the proposed method. This buffer system consists of the high-speed chiplet-to-chiplet communication links BoW standard shown in Figure 4.12 with $50\ \Omega$ characteristic impedance and the inverters-based FinFET buffer with $50\ \Omega$ load impedance. For 10 nm FinFET, the voltage supply is 0.75 V, so the input differential signal has a voltage range from -0.75 V to 0.75 V. At a typical BoW transmission rate of 32 Gbps, a differential single-bit pulse is sent to the TX driver as input, and the output response of the buffer system is received after one BoW differential channel. This input-output pair is used for the Wiener system identification. Then, the intermediate pulse and the polynomial expression of the nonlinearity are identified. Through direct statistical analysis with the choice of $N_{pre} = 5$ and $N_{post} = 50$, the intermediate statistical eye diagram is derived, and the random variable transformation is applied to get the nonlinear statistical eye diagram. In comparison, a 10^4 PRBS waveform transient simulation of the same setup is also performed. The transient eye diagram is constructed from the resulting waveform to compare with the statistical eye diagram as shown in Figure 4.13. Similarly, at the data rate of 64 Gbps, the eye diagrams of the two methods are illustrated in Figure 4.14 with a smaller eye-opening and constrained eye shape due to channel loss at a higher data rate. The quantitative measurement of the eye diagram information is summarized in Table 4.2 with a small discrepancy between the transient analysis and the proposed method. The CPU time also has a 70% reduction while maintaining a high accuracy.

Table 4.2: Comparison between the transient and nonlinear statistical simulation methods in the differential FinFET buffer example.

	32 Gbps			64 Gbps		
	EH (mV)	EW (UI)	Time (s)	EH (mV)	EW (UI)	Time (s)
Transient	415.0030	0.8871	14.7	201.7932	0.7768	17.5
Statistical	413.8299	0.8616	4.07	199.0208	0.7723	5.31
Error	-0.28%	-2.87%	—	-1.37%	-0.58%	—

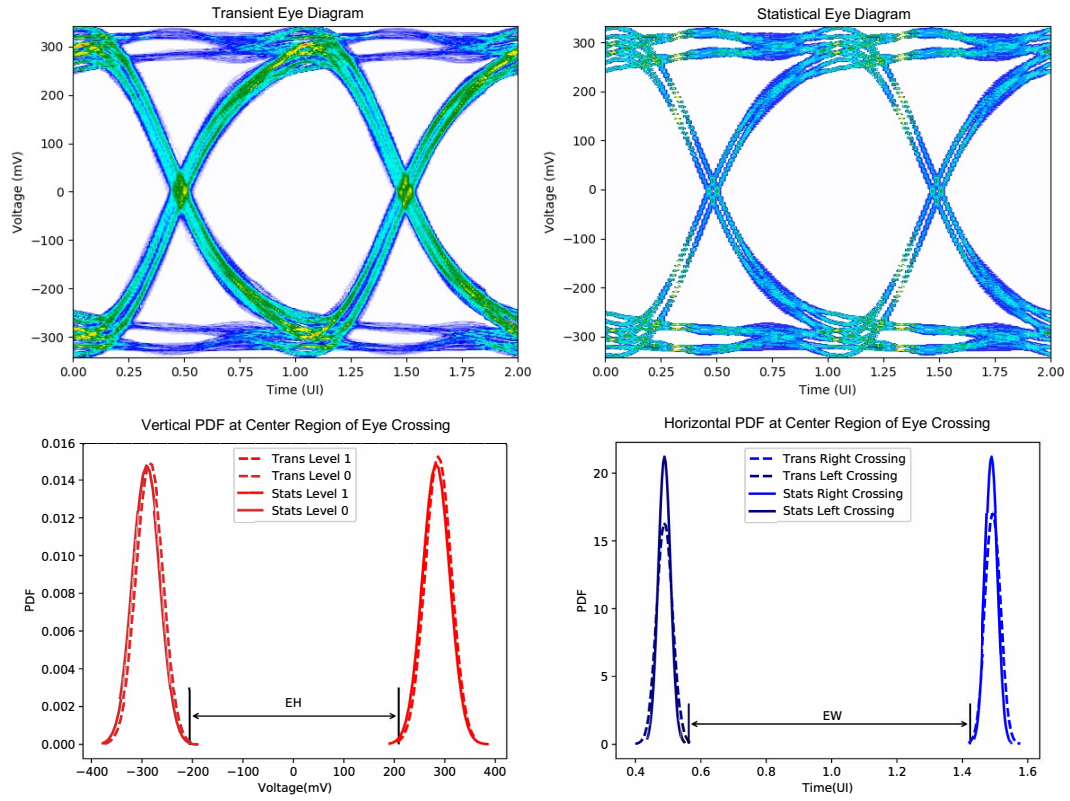


Figure 4.13: The comparison between the transient simulation and proposed nonlinear statistical analysis in 32 Gbps differential FinFET buffer.

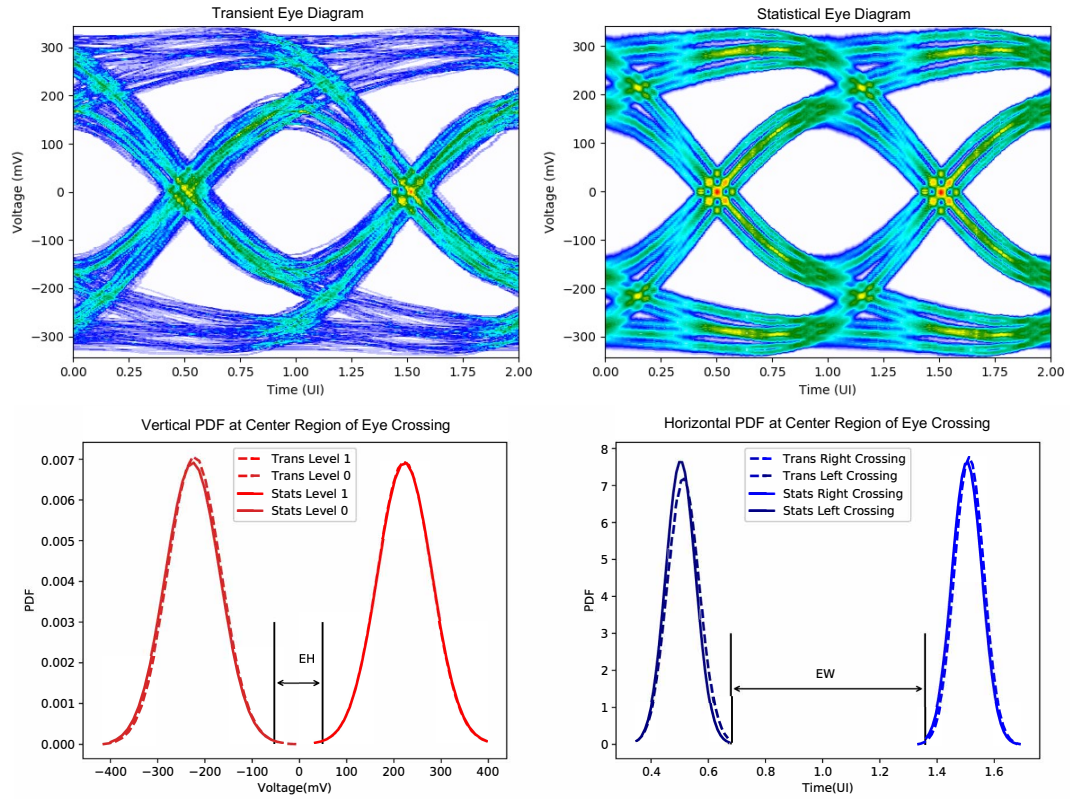


Figure 4.14: The comparison between the transient simulation and proposed nonlinear statistical analysis in 64 Gbps differential FinFET buffer.

4.3.3 High-speed Link System with Rx CTLE and DFE

Equalization is crucial in high-speed link systems to mitigate the effects of signal distortion caused by various factors such as attenuation, noise, and frequency-dependent losses. The typical equalization used in Rx is the combination of Continuous-Time Linear Equalization (CTLE) and Decision Feedback Equalization (DFE), a nonlinear system, primarily due to its feedback mechanism [57]. Figure 4.15 shows the topology of this high-speed link system with Rx CTLE and DFE, where CTLE is a differential amplifier with RC source degeneration and CTLE is a summing amplifier with a nonlinear finite impulse response (FIR) feedback filter. This equalization is to restore the signal distortion of a USB 3.1 channel at 10 Gbps. There are three eye diagrams of interest to be detected in this high-speed link system: V_{CH} the eye diagram at channel output, V_{CTLE} the eye diagram after the CTLE system, and V_{OUT} the eye diagram after the DFE network. Therefore, 3 Wiener models are required to be identified. The differential single-bit pulse V_{in} is defined with $V_{low} = -750mV$, $V_{high} = 750mV$, $T_{rise} = T_{fall} = 35ps$, and $T_{pulse} = 100ps$ with period of $10ns$. The corresponding output pulse responses are collected at V_{CH} , V_{CTLE} , and V_{OUT} to form three input-output pairs for Wiener system identifications.

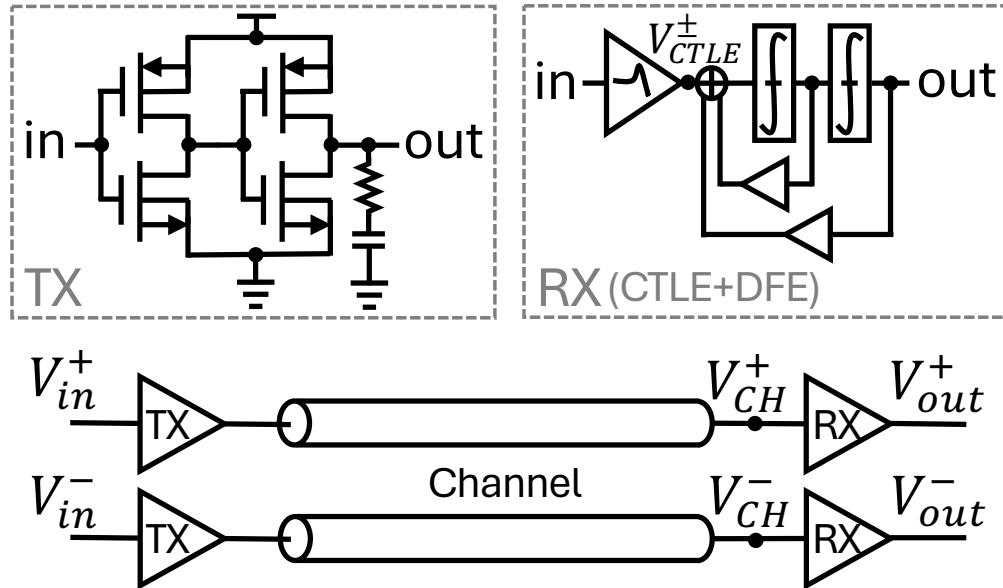


Figure 4.15: The schematic of the high-speed link system with Rx CTLE and DFE.

Following the proposed analysis and Wiener model decomposition, in the first Wiener system between V_{in} and V_{CH} , the intermediate pulse $x(t)$ is derived as well as the polynomial function that can map $x(t)$ to $y(t)$ at V_{CH} . In this case, 5 pre-cursors and 50 post-cursors are used to implement the statistical analysis from $x(t)$ to statistical eye diagram $f_X(x, t)$. Then, the nonlinear statistical eye $f_Y(y, t)$ is mapped from $f_X(x, t)$. To assess accuracy, the transient simulation probes the output waveform at V_{CH} using a 10^4 bits PRBS. Subsequently, a transient eye diagram is generated for analysis. The eye diagrams of the two methods and the voltage and time PDFs are shown in Figure 4.16. The density difference along the voltage leads to color variations between two eye diagrams. Similarly, the other two Wiener models for $\{V_{in}, V_{CTLE}\}$ and $\{V_{in}, V_{OUT}\}$ are identified and their eye diagrams comparisons are presented in Figure 4.17 and Figure 4.18. Table 4.3 summarized the eye-opening information. The eye-opening results from 10^4 bits transient simulation are noted as Transient 1. By comparing this transient result with the statistical result, the statistical eye height is significantly smaller in the order of 10%. This occurs because the transmission of bits may not adequately represent various distorted waveforms. Certain bit patterns can exert greater stress on the high-speed link system, resulting in more distortion in the representation of the eye diagram. In this example, 10^4 bits are not sufficient to cover most stressful cases. Therefore, 10^5 bits are sent to the system to collect the eye diagram again. In Table 4.3, the revised transient eye diagram outcome is labeled as Transient 2, indicating narrower eye-opening and fewer errors when compared to the statistical eye diagram findings. The preference for statistical analysis becomes more evident when considering simulation runtime comparisons.

In Table 4.4, although transient analysis involves only one simulation to obtain three results, the total CPU time required for running the proposed statistical analysis three times remains significantly lower. The runtime increase between Transient 1 and Transient 2 is exponential due to the SPICE transient analysis nature. The CPU runtime at V_{CH} is longer due to the larger size in the 2D PDF voltage distribution compared to other cases. This larger size results from the presence of more identical voltage levels between -300 mV and 300 mV. The thickness observed in level one and level zero of the 2D PDF serves as evidence of multiple voltage columns, in contrast to the thinner level one and level zero observed in the eye diagram. In this

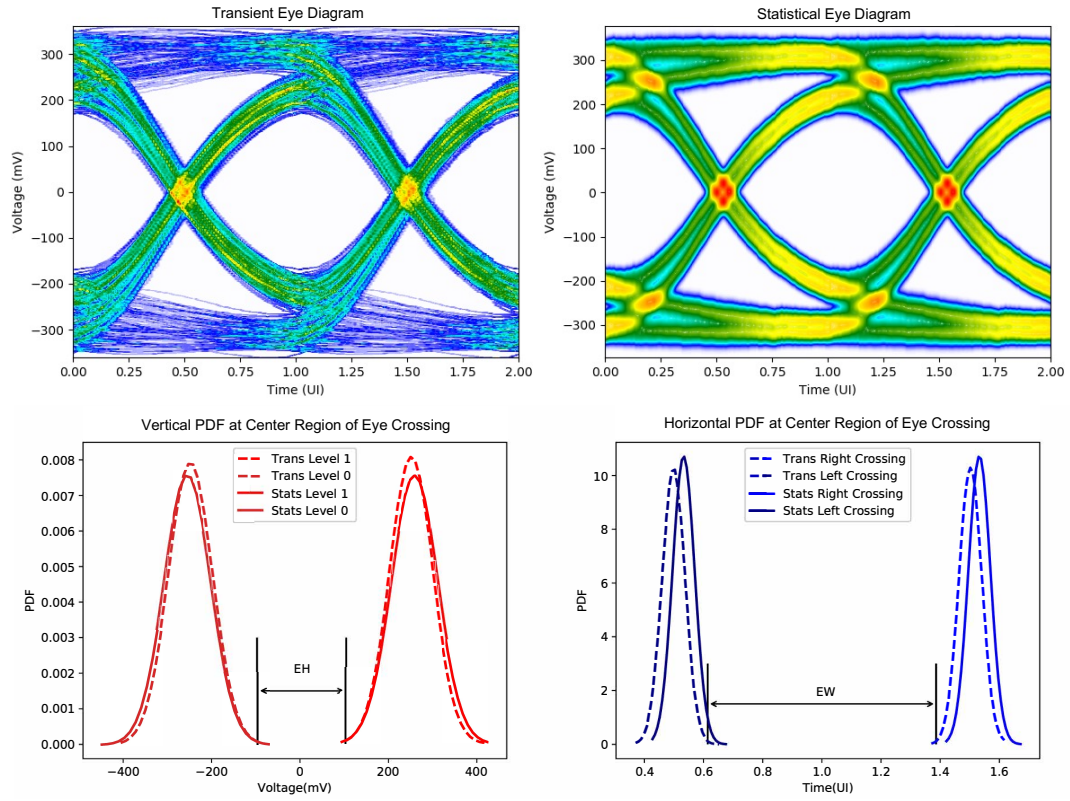


Figure 4.16: The comparison between the transient simulation and proposed nonlinear statistical analysis of the eye diagram at V_{CH} .

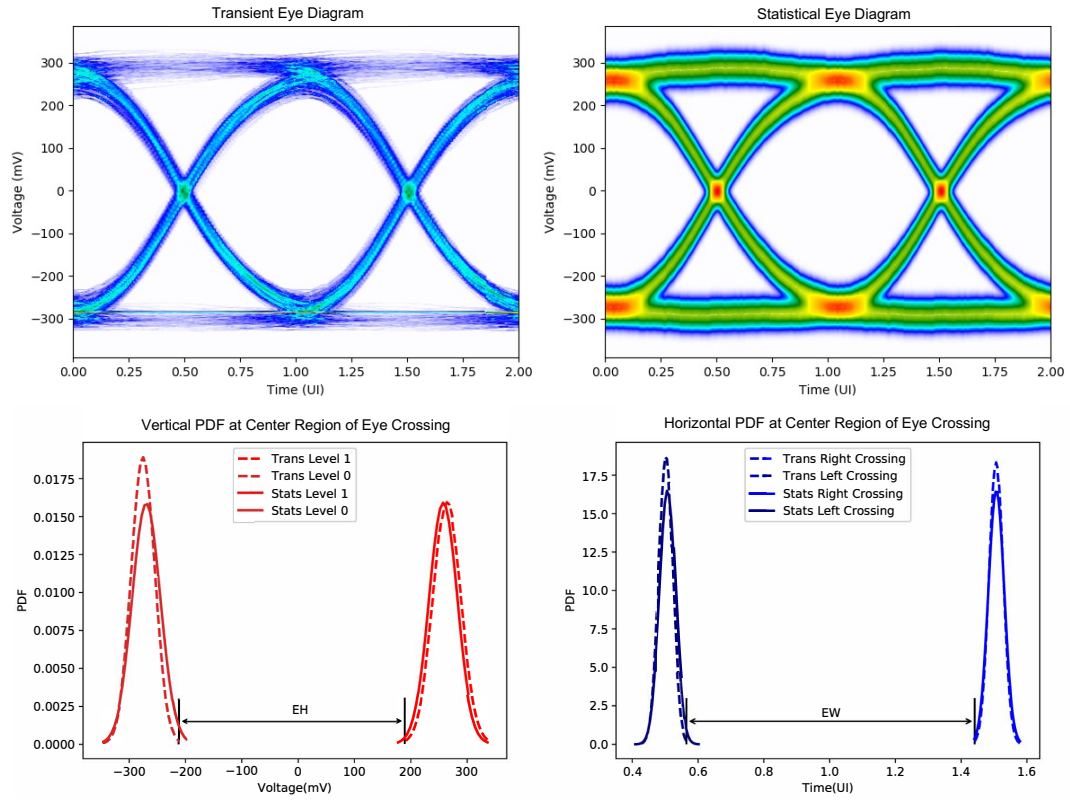


Figure 4.17: The comparison between the transient simulation and proposed nonlinear statistical analysis of the eye diagram at V_{CTLE} .

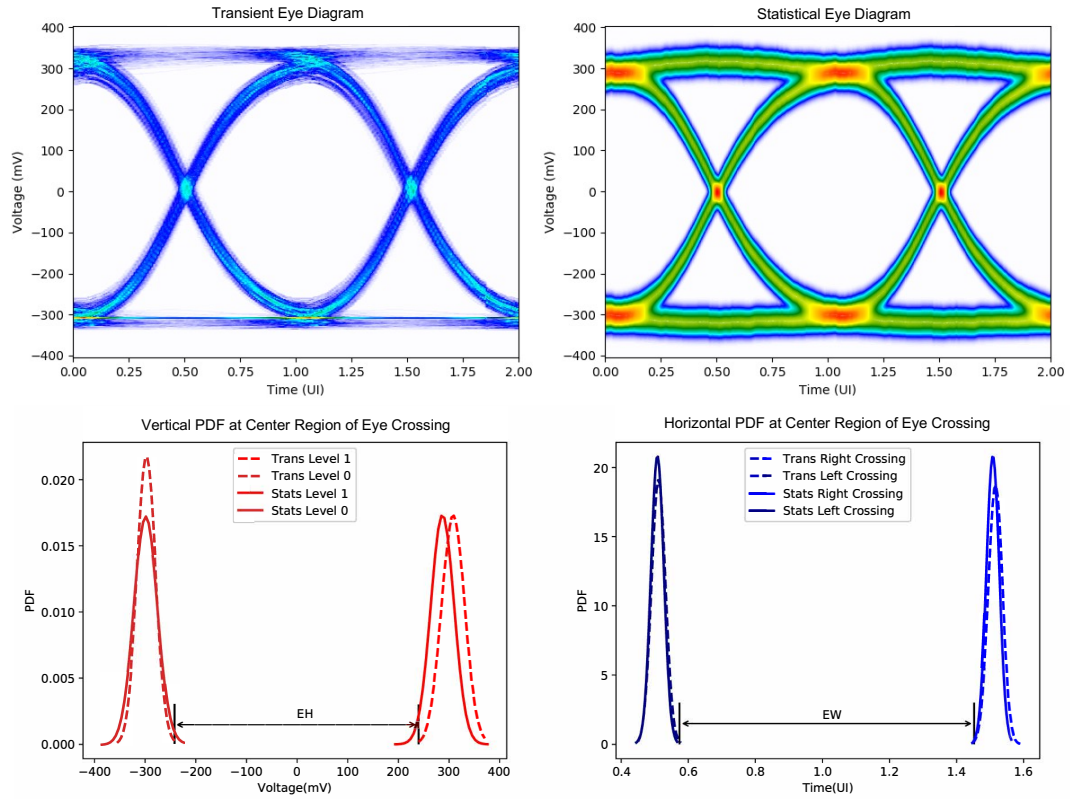


Figure 4.18: The comparison between the transient simulation and proposed nonlinear statistical analysis of the eye diagram at V_{OUT} .

example, there are around 9000 voltage levels in the statistical eye diagram at V_{CH} , compared to about 1500 voltage levels at V_{CTLE} and V_{OUT} . The CPU runtime in statistical analysis depends on the number of cursors to the convolution and the number of identical voltage levels. In this nonlinear high-speed link system example, the proposed method results in a maximum of 4.41% in error in eye-opening but reduces the CPU runtime from 43 minutes to 28 seconds, which verifies the efficiency and accuracy of this algorithm.

Table 4.3: Comparison between the transient and nonlinear statistical simulation methods in the high-speed link example.

	V_{CH}		V_{CTLE}		V_{OUT}	
	EH (mV)	EW (UI)	EH (mV)	EW (UI)	EH (mV)	EW (UI)
Transient 1	483.1616	0.8851	165.6665	0.6005	394.2349	0.5966
Transient 2	454.6452	0.8898	143.6442	0.5999	351.4231	0.5912
Statistical	447.9806	0.8782	137.3159	0.5922	344.7546	0.5928
Error 1	-7.28%	-0.78 %	-17.11%	-1.38%	-12.55%	-0.64%
Error 2	-1.47%	-1.30 %	-4.41%	-1.28%	-1.90%	-0.27%

Table 4.4: Runtime comparison between the transient and nonlinear statistical simulation methods in the high-speed link example.

	V_{CH}	V_{CTLE}	V_{OUT}
Transient 1	144 s		
Transient 2	43 m 16.9 s		
Statistical	16.58 s	5.53 s	5.96 s

4.4 Conclusion

This chapter introduces a novel approach for estimating statistical eye diagrams in nonlinear high-speed links through the Volterra-Wiener model system identification. The method involves approximating the nonlinearity into the LTI systems and polynomial nonlinear systems and leveraging the knowledge of density transformation within nonlinear systems to estimate the statistical eye diagram of the expected signal. The examples of the ideal

nonlinearity Wiener model, the differential FinFET buffer, and the high-speed link with nonlinear equalization are analyzed by the proposed method to provide feasible and accurate eye diagram results. Additionally, the article conducts transient simulations in different lengths of bits to validate the accuracy of this approach as the comparison.

CHAPTER 5

CONCLUSION AND FUTURE WORK

5.1 Conclusion

In this thesis, the surrogate modeling of high-speed links serves as a pivotal method for expediting the estimation of eye diagrams, particularly employing the polynomial chaos expansion technique. Unlike some surrogate modeling techniques which predict eye height and eye width only [7,8], the polynomial chaos expansion generates the whole transient waveform containing more information. Rather than training the complete bit pattern at the input as a single surrogate model, the approach involves categorizing the input into four groups of bits based on the last 2-bit transition states, which subsequently serve as training sets for four surrogate models. This classification strategy significantly reduces the time required for solving the unknown coefficients within a polynomial chaos expansion matrix, owing to the reduced matrix dimension. Subsequently, the eye diagram is reconstructed through the superposition of the outputs from these four surrogate models.

In addition to its application in estimating waveforms and eye diagrams for high-speed links, the polynomial chaos expansion technique provides importance in estimating tap values for decision-feedback equalization across various channel geometries. This means that engineers can anticipate how to set up the DFE to effectively restore compromised signals based on their knowledge of the channel, all without running extensive electromagnetic solvers and channel simulators. Similarly, in microwave filter design, this technique offers a quicker path to estimating critical parameters like center frequency, bandwidth, and shape factor directly, from multiple design variables such as lengths, widths, and permittivity. The fact that these estimations converge quickly and achieve a validation R^2 score of 0.99 highlights the effectiveness of this modeling approach in Chapter 2.

Moreover, the necessity for periodic retraining of surrogate models highlights the dynamic nature of high-speed link systems. Changes or updates within the system properties or operating conditions require re-calculation of the surrogate models to maintain accuracy and reliability in prediction. This iterative process underscores the importance of adaptability and responsiveness in modeling techniques, ensuring that the surrogate models remain reflective of the evolving system dynamics.

In parallel, this necessity leads to the exploration of a second approach for eye diagram estimation, employing statistical analysis. While statistical analysis of eye diagrams has been conventionally applied to linear systems, the incorporation of proper nonlinearity modeling can enhance the analysis for nonlinear systems. In Chapter 3, the bit-by-bit method and direct statistical method for the linear time-invariant system are shown. In Chapter 4, by leveraging the Wiener model and random variable transformation, the statistical approach provides insights into the probabilistic distribution of eye diagrams, accounting for the inherent variability and uncertainties associated with nonlinear systems. This methodological diversity enriches the analytical toolkit available to engineers, enabling a comprehensive understanding of system behavior and performance.

The validation of statistical analysis results against transient analysis demonstrates the robustness and accuracy of the proposed methodology. Three examples of high-speed link systems with nonlinear conditions are presented in Chapter 4. The close agreement between statistical and transient analysis confirms the efficacy of the proposed statistical analysis in capturing the nonlinearity of high-speed link systems. Furthermore, the acceptable error rates observed highlight the reliability of statistical analysis as a viable alternative or complement to traditional transient simulation methods.

In summary, the surrogate modeling technique addresses the runtime challenges associated with transient simulation, while the nonlinear statistical analysis method tackles nonlinearities in high-speed link systems during statistical eye diagram estimation. Both approaches have demonstrated effectiveness and accuracy in this dissertation. In future work, combining the computational efficiency of surrogate models with the probabilistic insights provided by statistical analysis holds promise for electrical engineers. Engineers could be equipped with this versatile tool to address the multifaceted challenges posed by nonlinearities and dynamic system environments. This

synergistic approach could not only enhance the efficiency of eye diagram estimation but also enrich the analytical capabilities required for advancing high-speed link technologies.

5.2 Limitations and Future Work

With all the advantages discussed above, there are still some limitations remaining with future development. Surrogate models, while efficient, are often constructed based on a subset of data or a simplified representation of the original model. This can result in inaccuracies or incomplete representations, especially when faced with system changes. In the high-speed link example, the load impedance is not one of the design parameters, therefore whenever the change in it can significantly impact the accuracy of predictions made by surrogate models. Furthermore, Polynomial Chaos Expansion models may not fully capture all the essential nonlinear features of the original model, leading to potential discrepancies between predictions and actual system behavior. To mitigate this limitation, techniques such as Hyperbolic or Sparse Polynomial Chaos Expansion [58–60] could be explored, offering enhanced accuracy and robustness.

Similarly, the proposed statistical analysis, while promising, presents its own challenges. In the discussed example involving high-speed links with DFE and CTLE, the identification of multiple Wiener models for different voltage nodes is necessary, which means statistical eyes cannot cascade with each other. This is because, in the dynamic system either linear or nonlinear, the density transformation relationship is difficult to define. Therefore, this process can be computationally intensive and time-consuming. The possible solution is through the reverse process described in Figure 5.1. Here, the statistical eye is deconstructed back into an artificial pulse response, which is then convolved with the LTI channel system to yield the channel-responded pulse. Subsequently, the statistical eye at V_{CH} can be calculated through direct statistical analysis. By leveraging this technique, researchers can navigate the complexities of nonlinear systems more effectively, paving the way for improved modeling and analysis in various domains.

In addition to the algorithm improvement, more comparisons could be done to prove the advantage of the proposed analysis methods. The IBIS model,

known for its nonlinear and dynamic characteristics, has been extensively applied in current simulations for decades. A more comprehensive analysis would involve comparing circuit-level SPICE simulations, IBIS-AMI models, and the proposed methodologies to show the computational efficiency and accuracy of the proposed simulation methods. In future work concerning nonlinearity diagnosis, once the factors contributing to nonlinear effects are learned and integrated into the training process, the estimated eye diagram is able to reflect the corresponding nonlinear behaviors like an asymmetrical rising and falling time.

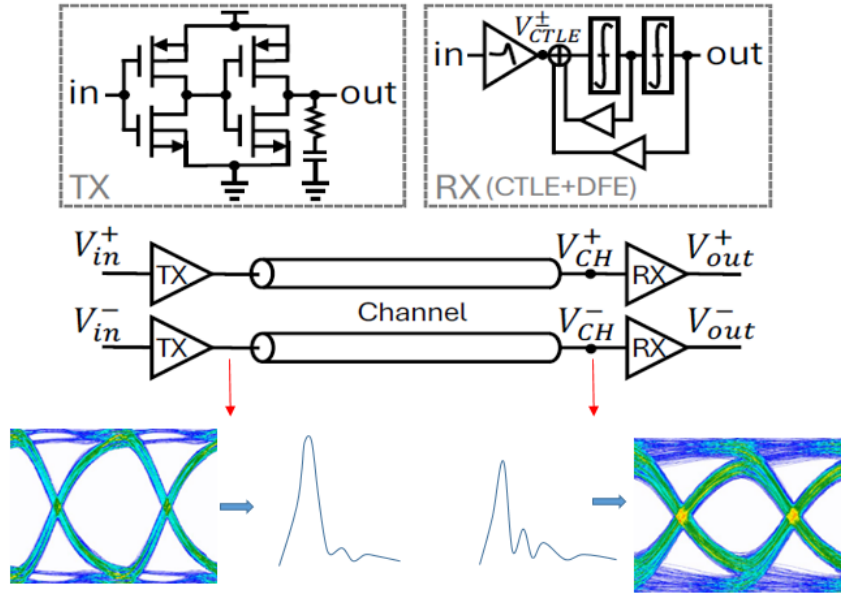


Figure 5.1: The illustration of reversing the statistical eye diagram to an artificial pulse response.

REFERENCES

- [1] O. Sahlen, “Active dbr filters for 2.5 gb/s operation: linewidth, crosstalk, noise, and saturation properties,” *Journal of Lightwave Technology*, vol. 10, no. 11, pp. 1631–1643, 1992.
- [2] B. Mutnury, M. Swaminathan, and J. Libous, “Macromodeling of non-linear digital i/o drivers,” *IEEE Transactions on Advanced Packaging*, vol. 29, no. 1, pp. 102–113, 2006.
- [3] I. Stievano, I. Maio, and F. Canavero, “Parametric macromodels of digital i/o ports,” *IEEE Transactions on Advanced Packaging*, vol. 25, no. 2, pp. 255–264, 2002.
- [4] L. W. Nagel and D. Pederson, “Spice (simulation program with integrated circuit emphasis),” EECS Department, University of California, Berkeley, Tech. Rep. UCB/ERL M382, Apr 1973. [Online]. Available: <http://www2.eecs.berkeley.edu/Pubs/TechRpts/1973/22871.html>
- [5] T. Nguyen and J. Schutt-Aine, “A tunable neural network based decision feed-back equalizer model for high-speed link simulation,” in *2020 IEEE 29th Conference on Electrical Performance of Electronic Packaging and Systems (EPEPS)*, 2020, pp. 1–3.
- [6] T. Lu, J. Sun, K. Wu, and Z. Yang, “High-speed channel modeling with machine learning methods for signal integrity analysis,” *IEEE Transactions on Electromagnetic Compatibility*, vol. 60, no. 6, pp. 1957–1964, 2018.
- [7] H. Ma, E.-P. Li, A. C. Cangellaris, and X. Chen, “Comparison of machine learning techniques for predictive modeling of high-speed links,” in *2019 IEEE 28th Conference on Electrical Performance of Electronic Packaging and Systems (EPEPS)*, 2019, pp. 1–3.
- [8] R. Trinchero and F. G. Canavero, “Modeling of eye diagram height in high-speed links via support vector machine,” in *2018 IEEE 22nd Workshop on Signal and Power Integrity (SPI)*, 2018, pp. 1–4.

- [9] Y. Zhao, T. Nguyen, H. Ma, E.-P. Li, A. C. Cangellaris, and J. E. Schutt-Ainé, “Modular neural network-based models of high-speed link transceivers,” *IEEE Transactions on Components, Packaging and Manufacturing Technology*, vol. 13, no. 10, pp. 1603–1612, 2023.
- [10] H. Ma, E.-P. Li, A. C. Cangellaris, and X. Chen, “Support vector regression-based active subspace (SVR-AS) modeling of high-speed links for fast and accurate sensitivity analysis,” *IEEE Access*, vol. 8, pp. 74 339–74 348, 2020.
- [11] H. Ma, E.-P. Li, A. C. Cangellaris, and X. Chen, “High-speed link design optimization using machine learning SVR-AS method,” in *2020 IEEE 28th Conference on Electrical Performance of Electronic Packaging and Systems (EPEPS)*, Oct. 2020.
- [12] T. Nguyen, B. Shi, H. Ma, E.-P. Li, X. Chen, A. C. Cangellaris, and J. Schutt-Ainé, “Comparative study of surrogate modeling methods for signal integrity and microwave circuit applications,” *IEEE Transactions on Components, Packaging and Manufacturing Technology*, vol. 11, no. 9, pp. 1369–1379, 2021.
- [13] M. Ahadi Dolatsara, J. A. Hejase, W. D. Becker, and M. Swaminathan, “A hybrid methodology for jitter and eye estimation in high-speed serial channels using polynomial chaos surrogate models,” *IEEE Access*, vol. 7, pp. 53 629–53 640, 2019.
- [14] J. Slim, F. Rathmann, A. Nass, H. Soltner, R. Gebel, J. Pretz, and D. Heberling, “Polynomial chaos expansion method as a tool to evaluate and quantify field homogeneities of a novel waveguide rf wien filter,” *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 859, pp. 52–62, 2017. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0168900217303807>
- [15] W. T. Beyene, A. Zargaran-Yazd, and X. Ma, “Statistical simulation of high-speed links with transmitter and receiver nonlinearities,” in *Proc. IEEE Electr. Performance Electron. Packag.*, 2014, pp. 35–38.
- [16] W. T. Beyene, “Peak distortion analysis of nonlinear links,” in *2013 IEEE 22nd Conference on Electrical Performance of Electronic Packaging and Systems*, 2013, pp. 169–172.
- [17] D. Jiao and J. Zhu, “Fast method for an accurate and efficient nonlinear signaling analysis,” *IEEE Trans. Electromagn. Compat.*, vol. 59, no. 4, pp. 1312–1319, 2017.

- [18] C.-C. Chou, S.-Y. Hsu, and T.-L. Wu, “Estimation method for statistical eye diagram in a nonlinear digital channel,” *IEEE Trans. Electromagn. Compat.*, vol. 57, no. 6, pp. 1655–1664, 2015.
- [19] B. Casper, M. Haycock, and R. Mooney, “An accurate and efficient analysis method for multi-gb/s chip-to-chip signaling schemes,” in *Proc. VLSI Circuits Dig. Tech. Papers Symp*, 2002, pp. 54–57.
- [20] M. V. Dunga, C.-H. Lin, D. D. Lu, W. Xiong, C. R. Cleavelin, P. Patruno, J.-R. Hwang, F.-L. Yang, A. M. Niknejad, and C. Hu, “Bsim-mg: A versatile multi-gate fet model for mixed-signal design,” in *2007 IEEE Symposium on VLSI Technology*, 2007, pp. 60–61.
- [21] E. Matthews, “The use of scattering matrices in microwave circuits,” *IRE Transactions on Microwave Theory and Techniques*, vol. 3, no. 3, pp. 21–26, 1955.
- [22] D. Vye, “X-parameters,” *Microwave Journal*, 2010.
- [23] B. Ross, “Ibis evolution ver.7.1,” 2021. [Online]. Available: <https://ibis.org/ver7.1/>
- [24] N. Kapre and A. DeHon, “Parallelizing sparse matrix solve for spice circuit simulation using fpgas,” *International Conference on Field-Programmable Technology*, pp. 190–198, 2009.
- [25] B. Shi, Y. Zhao, H. Ma, T. Nguyen, E.-P. Li, A. C. Cangellaris, and J. Schutt-Aine, “Decision feedback equalizer (dfe) taps estimation with machine learning methods,” in *2021 IEEE Electrical Design of Advanced Packaging and Systems (EDAPS)*, 2021, pp. 1–3.
- [26] T. Nguyen, B. Shi, and J. Schutt-Aine, “Comparative study of machine learning methods for variability analysis in high-speed link,” in *2021 IEEE 25th Workshop on Signal and Power Integrity (SPI)*, 2021, pp. 1–3.
- [27] D. Xiu, “Fast numerical methods for stochastic computations: a review,” in *Communications in computational physics*, vol. 5, no. 2-4, 2009, pp. 242–272.
- [28] D. Xiu and G. E. Karniadakis, “The wiener–askey polynomial chaos for stochastic differential equations,” *SIAM Journal on Scientific Computing*, vol. 24, no. 2, pp. 619–644, 2002. [Online]. Available: <https://doi.org/10.1137/S1064827501387826>
- [29] A. J. Smola and B. Schölkopf, “A tutorial on support vector regression,” *Statistics and Computing*, vol. 14, no. 3, pp. 199–222, Aug. 2004.

- [30] V. N. Vapnik, *The nature of statistical learning theory*. Springer science & business media, 1995.
- [31] T. Nguyen and J. Schutt-Aine, “Gaussian process surrogate model for variability analysis of RF circuits,” in *2020 IEEE Electrical Design of Advanced Packaging and Systems (EDAPS)*, 2020, pp. 1–3.
- [32] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press, 2005.
- [33] M. A. Álvarez, L. Rosasco, and N. D. Lawrence, “Kernels for vector-valued functions: A review,” *Foundations and Trends in Machine Learning*, vol. 4, no. 3, pp. 195–266, 2012.
- [34] R. Shi, W. Yu, Y. Zhu, C.-K. Cheng, and E. S. Kuh, “Efficient and accurate eye diagram prediction for high speed signaling,” in *Proc. IEEE/ACM Int. Conf. Comput.-Aided Design*, 2008, pp. 655–661.
- [35] J. Ren and K. S. Oh, “Multiple edge responses for fast and accurate system simulations,” *IEEE Transactions on Advanced Packaging*, vol. 31, no. 4, pp. 741–748, 2008.
- [36] Y. Zhou, B. Shi, Y. Zhao, and J. E. Schutt-Ainé, “Fast eye diagram simulation based on latency insertion method,” in *2022 IEEE Electrical Design of Advanced Packaging and Systems (EDAPS)*, 2022, pp. 1–3.
- [37] Y. Zhou, B. Shi, and J. E. Schutt-Ainé, “Fast latency-insertion-method-based eye diagram simulation incorporating crosstalk,” in *2023 IEEE 27th Workshop on Signal and Power Integrity (SPI)*, 2023, pp. 1–4.
- [38] D. Oh, “Multiple edge responses for fast and accurate system simulations,” in *Proc. IEEE Electr. Performance Electron. Packag.*, 2006, pp. 163–166.
- [39] A. Sanders, “Statistical simulation of physical transmission media,” *IEEE Transactions on Advanced Packaging*, vol. 32, no. 2, pp. 260–267, 2009.
- [40] A. Sanders, M. Resso, and J. D’ Ambrosia, “Channel compliance testing utilizing novel statistical eye methodology,” in *DesignCon*, Feb. 2004.
- [41] Y. Dou, D. Jiao, J. Yan, and J. Zhu, “Method for analyzing bit error rates (bers) of nonlinear circuits and systems for high-performance signaling,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 70, no. 1, pp. 732–743, 2022.

- [42] Z. Chen, W. D. Becker, and G. Katopis, "A new approach to deriving packaging system statistical eye diagram based on parallel non-linear transient simulations using multiple short signal bit patterns," in *2012 IEEE 62nd Electronic Components and Technology Conference*, 2012, pp. 160–167.
- [43] H. Kim, K. Kim, S. Choi, H. Lee, H. Kim, Y. Kim, and J. Kim, "A fast and accurate statistical eye-diagram estimation method for high-speed channel including non-linear receiver buffer circuit," in *2015 Asia-Pacific Symposium on Electromagnetic Compatibility (APEMC)*, 2015, pp. 94–97.
- [44] J. Wang, Y. Luo, W. Guo, F. Wu, and X. Chu, "Fast estimation of a statistical eye diagram for nonlinear high-speed links based on the minimum required order of the multiple edge response method," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 31, no. 2, pp. 210–218, 2023.
- [45] X. Chu, W. Guo, J. Wang, F. Wu, Y. Luo, and Y. Li, "Fast and accurate estimation of statistical eye diagram for nonlinear high-speed links," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 29, no. 7, pp. 1370–1378, 2021.
- [46] X. Chu, J. Wang, F. Wu, Y. Luo, W. Wang, and Y. Li, "Simulink-based modeling and pca-based nonlinearity evaluation for high-speed links," *IEEE Transactions on Electromagnetic Compatibility*, vol. 62, no. 4, pp. 1386–1397, 2020.
- [47] B. Shi, Y. Zhou, T. Nguyen, and J. Schutt-Aine, "Statistical method for eye diagram simulation in a high-speed link nonlinear system," in *2022 IEEE Electrical Design of Advanced Packaging and Systems (EDAPS)*, 2022, pp. 1–3.
- [48] G. Balamurugan, B. Casper, J. E. Jaussi, M. Mansuri, F. O'Mahony, and J. Kennedy, "Modeling and analysis of high-speed i/o links," *IEEE Transactions on Advanced Packaging*, vol. 32, no. 2, pp. 237–247, 2009.
- [49] J. Savoj, A.-A. Abbasfar, A. Amirkhany, B. W. Garlepp, and M. A. Horowitz, "A new technique for characterization of digital-to-analog converters in high-speed systems," in *2007 Design, Automation Test in Europe Conference Exhibition*, 2007, pp. 1–6.
- [50] M. Schoukens and K. Tiels, "Identification of block-oriented nonlinear systems starting from linear approximations: A survey," *Automatica*, vol. 85, pp. 272–292, 2017. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0005109817303990>

- [51] V. Volterra, *Theory of Functionals and of Integral and Integro-differential Equations*, 1959.
- [52] M. Schetzen, *Linear time-invariant systems*. Hoboken, NJ: John Wiley, 2003.
- [53] F. Giri and E.-W. Bai, *Block-oriented Nonlinear System Identification*. Springer Verlag, 2010.
- [54] A. Papoulis and S. U. Pillai, *Probability, Random Variables, and Stochastic Processes*, 4th ed. Boston: McGraw Hill, 2002.
- [55] H. Sun, B. Shi, T. Nguyen, and J. E. Schutt-Ainé, “Modeling and analysis of heterogeneously integrated chiplet-to-chiplet communication link in 2.5d advanced packaging,” in *2024 IEEE 66th Electronic Components and Technology Conference (ECTC)*, May 2024.
- [56] S. Ardalan, R. Farjadrad, M. Kuemerle, K. Poulton, S. Subramaniam, and B. Vinnakota, “An open inter-chiplet communication link: Bunch of wires (BoW),” *IEEE Micro*, vol. 41, pp. 51–60, Jan. 2021.
- [57] P. Hanumolu, G. Wei, and Y. Moon, “Equalizers for high-speed serial links,” *International Journal of High Speed Electronics and Systems*, vol. 15, no. 2, pp. 429–458, June 2005.
- [58] M. Ahadi, A. K. Prasad, and S. Roy, “Hyperbolic polynomial chaos expansion (hpce) and its application to statistical analysis of nonlinear circuits,” in *2016 IEEE 20th Workshop on Signal and Power Integrity (SPI)*, 2016, pp. 1–4.
- [59] F. Ni, P. H. Nguyen, and J. F. G. Cobben, “Basis-adaptive sparse polynomial chaos expansion for probabilistic power flow,” *IEEE Transactions on Power Systems*, vol. 32, no. 1, pp. 694–704, 2017.
- [60] P. Manfredi, D. Vande Ginste, D. De Zutter, and F. G. Canavero, “Generalized decoupled polynomial chaos for nonlinear circuits with many random parameters,” *IEEE Microwave and Wireless Components Letters*, vol. 25, no. 8, pp. 505–507, 2015.