Preprints.org

Article

Not peer-reviewed version

Custom Adaptive Kernel Strategies for Gaussian Process Regression in Wafer-Level Modeling and FPGA Delay Analysis

Riaz-ul-haque Mian * , Foisal Ahmed , Yoshito Hagihara , Yamane Souma

Posted Date: 4 March 2025

doi: 10.20944/preprints202503.0260.v1

Keywords: Multi-site RF IC Testing; Gaussian Process Regression; Wafer-level Variation Modeling; Kernel Function; Die Selection; FPGA; Adaptive Kernel

Preprints.org is a free multidisciplinary platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This open access article is published under a Creative Commons CC BY 4.0 license, which permit the free download, distribution, and reuse, provided that the author and preprint are cited in any reuse.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 4 March 2025 doi:10.20944/preprints202503.0260.v1

Disclaimer/Publisher's Note: The statements, opinions, and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions, or products referred to in the content.

Article

Custom Adaptive Kernel Strategies for Gaussian Process Regression in Wafer-Level Modeling and FPGA Delay Analysis

Riaz-ul-Haque Mian [1,2,*], Foisal Ahmed [3], Yoshito Hagihara [1] and Yamane Souma [1]

[1] Interdisciplinary Faculty of Science and Engineering, Department of Information Systems Design and Data Science, Shimane University, Matsue, Japan
[2] Estuary Research Center, Shimane University Matsue, Japan
[3] Stonridge Electronics, Inc. Valdmäe tn 5, Tänassilma, 76406 Harju maakond, Tallinn, Estonia
* Correspondence: riaz@cis.shimane-u.ac.jp

Abstract: The Gaussian Process Regression (GPR) has emerged as a state-of-the-art machine learning approach, offering significant potential for reducing testing costs in large-scale integrated circuits (LSIs) while maintaining high-quality standards. This study focuses on the wafer-level characteristic estimation, which measures a small subset of LSI circuits and estimates the characteristics of unmeasured ones. Additionally, it investigates the delay information estimation of field-programmable gate arrays (FPGAs) using ring oscillators (ROs) in look-up tables (LUTs). A key novelty of this work lies in addressing the critical challenge of kernel function selection, an essential factor in applying GPR to LSI testing and FPGA delay prediction. Through experimental analysis on mass-produced LSI industrial production data and actual measured silicon data, this research evaluates proposed adaptive custom kernel functions to identify optimal configurations. The findings reveal that, although hybrid and composite kernel architectures integrating multiple high-accuracy kernels outperform individual kernels in terms of accuracy, they are not consistent across different platforms. The proposed adaptive kernel consistently delivers improved prediction accuracy across multiple platforms, as demonstrated using industrial production data and actual measured silicon data.

Keywords: multi-site RF IC testing; gaussian process regression; wafer-level variation modeling; kernel function; die selection; FPGA; adaptive kernel

---

1. Introduction

Large-scale integrated circuits and field-programmable gate arrays are receiving significant attention in modern technology. Many sectors, from automotive to healthcare, rely heavily on these technologies. While LSIs offer high-density integration and optimized performance, FPGAs provide flexibility and reconfigurability, making them complementary technologies in many applications [1–4]. Given their critical role, ensuring device reliability through comprehensive testing has become paramount, as faulty devices can significantly impact system operations and societal functionality. This necessitates extensive testing across various manufacturing stages under diverse conditions [5]. As these semiconductor devices grow in complexity and functionality, the volume of required tests has surged, resulting in rising test costs. This escalating expense is a major concern, as testing now constitutes a substantial portion of the overall manufacturing cost for both LSIs and FPGAs [6–10]. The challenge of maintaining thorough testing while managing costs is particularly pronounced in FPGAs, where the reconfigurable nature of the device adds an extra layer of complexity to the testing process. Various test cost reduction methods have been proposed that apply data analytics, machine learning algorithms, and statistical methods [11–14,14]. In particular, the wafer-level characteristic modeling method based on the statistical algorithm is the most promising candidate that reduces the test cost, that is, measurement cost, without impairing the test quality [14–21]. On the other hand, the state-of-the-art FPGA estimation proposed in [8–10] shows that the estimated measurements can be

© 2025 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 4 March 2025 doi:10.20944/preprints202503.0260.v1

drastically reduced through a compressed sensing-based *virtual probe* technique [8,9,22–24]. However, as experimentally shown in [13,25,26], the experiments are conducted using wafer measurement data from a mass industry production, the *Gaussian process* GP-based estimation method surpasses the techniques presented in [8,9,16–19] which is based on compressed sensing [23,24]. Because the estimation eliminates the need for measurement, it not only reduces the cost of measurement but also can be used to reduce the number of test items and/or change the test limits, which is expected to improve the efficiency of adaptive testing [27–30]. In [15], the expectation-maximization (EM) algorithm [31] was used to predict the measurement. The *Gaussian process*-based method provides more accurate prediction results [13,14,20,21,25,26]. The use of GP modeling has another side benefit. As it calculates the confidence of a prediction, the user can confirm whether the number and location of measurement samples are sufficient, and suitable or not, being a great advantage from a practical viewpoint. In addition, since the GP can calculate the uncertainty of the estimation result, the user can confirm whether the number and location of measurement samples are sufficient and suitable or not.

Figure 1 presents an experimental analysis in the form of a heat map, showing the full measurement results for a wafer of a 28 nm analog/RF device with a 10% estimation. The figure includes three heat maps (a, b, c), where (a) represents the actual measured data, while (b) and (c) show the predicted data using a virtual probe and Gaussian process regression (GPR), respectively. In both cases, 10% of the samples are used to estimate the remaining 90% of the unmeasured samples. It is evident from the figure that both VP and GPR effectively address and accurately predict the unknown measured data with high accuracy and minimal error.

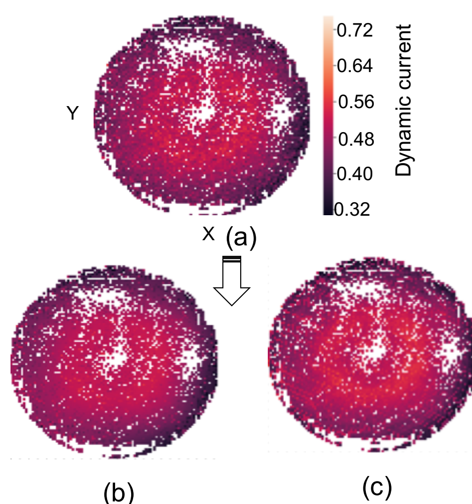


Figure 1. Comparison of heat maps for predictions by different methods for wafer data with 10 % training data from industry measured values: (a) Original, (b) VP, (c) Gaussian Process Regression(GPR [32]).

Figure 2 shows a similar experiment with delay information in look-up tables (LUTs) using ring oscillators (ROs). The figure includes three heat maps (a, b, c), where (a) represents the actual measured data, while (b) and (c) show the predicted data using a 10% virtual probe and 10% training data for the Gaussian process regression (GPR), respectively, The figure shows the similar trend on Figure 1 and in both cases VP and GPR effectively address as well as accurately predict the unknown measured data with high accuracy and minimal error.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 4 March 2025 doi:10.20944/preprints202503.0260.v1

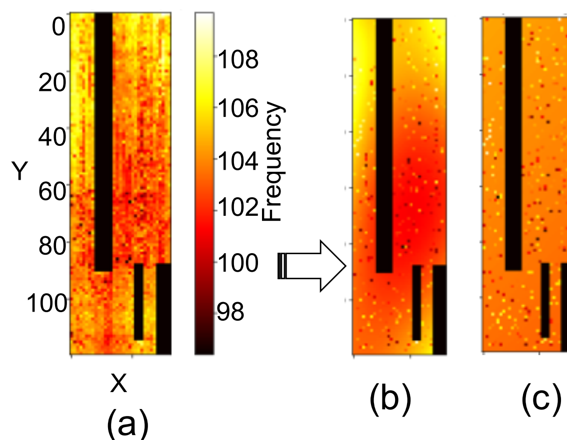


Figure 2. Comparison of heat maps for predictions by different methods for FPGA delay measurement of ring oscillator using 10 % training data: (a) Original, (b) VP, (c) Gaussian Process Regression (GPR [32]).

The VP technique exploits the sparsity of frequency-domain components in spatial process variation for prediction. Due to the gradual nature of process variation in FPGAs and wafers, high-frequency components approach zero, allowing seamless integration with fingerprinting techniques. The method's effectiveness is validated through silicon measurements on commercial FPGAs.

Unlike VP's direct frequency-domain approach, GPR models spatial process variation as a probabilistic distribution using kernel-based correlation analysis. This technique effectively handles gradual variations through Gaussian process smoothing, making it ideal for probabilistic predictions. Experimental validation using commercial FPGAs and wafer analog/RF device data demonstrates the method's effectiveness [33–37].

However, when applying GPR, it is most important to select the best kernel function, but the appropriate kernel function for LSI testing is not well understood. Since there is no effective method for investigating kernel function selection, this study applies GPR to mass production LSI test data, experimentally compares a large number of kernel functions, and identifies the best kernel function. Furthermore, based on this evaluation, we demonstrate that estimation accuracy can be further improved by using multiple kernel functions with high estimation accuracy. The hybrid or mixture kernels are proposed in [35–38] where different modes of mixing and combining are explained. In 3, sum of two product kernels $(k_{\mathrm{RBF}} \cdot k_{\mathrm{Lin}} + k_{\mathrm{RBF}} \cdot k_{\mathrm{RQ}}) \cdot k_{\mathrm{Multiplication}}$ achieves the best result than any individual kernel.

Most importantly, even when one single kernel or composite kernel function works well, it may not outperform others in different datasets with similar performance. Additionally, composite or adaptive kernels with too many parameters may suffer from overfitting issues. This study aims to determine a custom adaptation based on estimation prediction confidence to select the top two candidates for adaptation with dynamic weight adjustment based on the estimation confidence, instead of using fixed-weight adaptive kernels. Unlike typical hybrid or mixture kernels proposed in [35–38], this approach determines the weight of each kernel based on their difference in confidence levels. After an exhaustive analysis of various kernel functions across all possible combinations, using actual industry production data and real silicon data, it was observed that the proposed custom adaptive kernel consistently outperformed across two different datasets.

The remainder of this paper is organized as follows: Section 2 provides an overview of GP-based modeling for wafer and FPGA process variations. Section 3 details the kernel function and proposed custom adaptive kernel strategies. Section **??** presents a quantitative evaluation of the proposed custom kernel approach compared to conventional approaches. Finally, Section 5 offers the conclusion.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 4 March 2025 doi:10.20944/preprints202503.0260.v1

2. Gaussian Process Regression for Semiconductor Reliability

A Gaussian distribution is a type of probability distribution for continuous variables, also known as a normal distribution. In this distribution, the mean, mode, and median are identical, and the distribution is symmetric around the mean. Since it is continuous rather than discrete, the distribution is represented as a smooth curve rather than a histogram. The probability density function of the Gaussian distribution is given by:

$$\frac{1}{\sqrt{2\pi\sigma}} \cdot e^{-\frac{(x-\mu)^2}{2\sigma^2}}, \tag{1}$$

where μ represents the expected value (mean) and σ denotes the standard deviation.

The probability distribution treats the function $f(x)$ as a random variable, creating a probabilistic model that predicts y for a given input x. When $f(x)$ is defined, the outputs $f(x_1), f(x_2), \ldots, f(x_N)$ for inputs $x_1, x_2, \ldots, x_N$ follow a Gaussian process. The joint distribution of $y(x)$ at any set of points $x_1, x_2, \ldots, x_N$ is Gaussian. This property holds true even as N approaches infinity, making a Gaussian process an infinite-dimensional Gaussian distribution.

As illustrated in Figure 3, Gaussian process methods are used to model the relationship between input X and output Y. This relationship is expressed mathematically as:

$$Y = f([X, Y]) + \varepsilon, \tag{2}$$

where ε represents the model's noise or uncertainty. Data collected from the Wafer and FPGA are used to establish this relationship. Using this model, we can estimate the new output $Y(n+1)$ for both Wafer and FPGA is the new input data $X(n+1)$. The model is constructed as a function of the probability distribution using Bayesian inference.

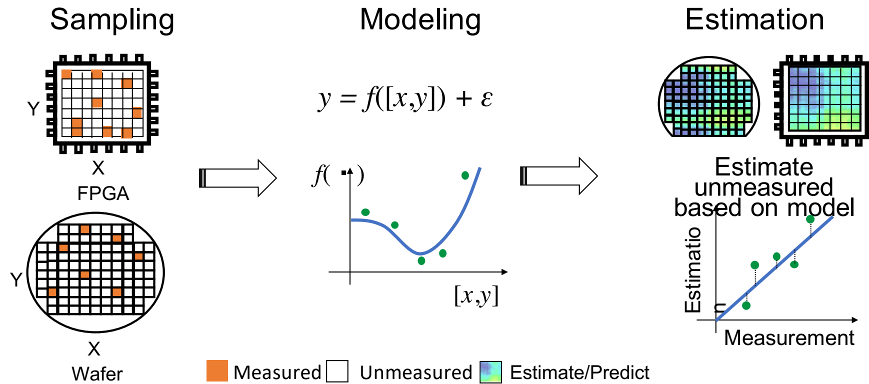


Figure 3. Model of the kernel matrix derivation process.

As mentioned earlier, the semiconductor industry is pursuing methods to improve quality without increasing test costs, or to reduce costs while maintaining quality. Particularly for the latter, wafer-level characteristic modeling methods based on statistical algorithms yield good results. This model uses Gaussian process regression for prediction, creating a prediction model based on limited sample measurements on the wafer surface or FPGA lookup table to predict the entire wafer surface and entire lookup tables.

Figure 4a–c illustrate how the model the relationship between input x and output y. Data D is utilized to model the relationship. The new output y(n+ 1) for the new data x (n+ 1) is estimated based on the model. The model is calculated as a function of the probability distribution with the Bayesian manner.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 4 March 2025 doi:10.20944/preprints202503.0260.v1

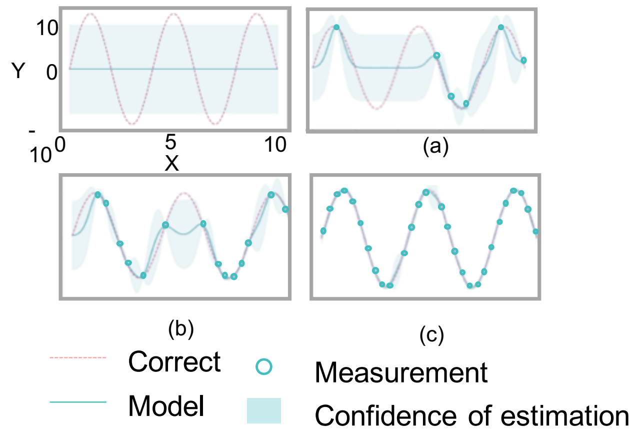


Figure 4. Gaussian process regression model to predict y from input x.

3. Proposed Custom Adaptive Kernel Approach

The GP-based regression employs kernel functions from the GPy [32] framework to model and predict the characteristics of LSIs and FPGAs.

This method learns from the measurement results of a small number of LSIs on the wafer and FPGA delay information. After the learning phase, it determines two functions: $f(\cdot)$, which relates to the wafer LSI coordinates, and $g(\cdot)$, which relates to FPGA delay information in look-up tables (LUTs) using ring oscillators (ROs). These functions estimate the characteristics of unmeasured LSIs and FPGAs, significantly reducing measurement time and costs.

Let $\mathbf{x}_{\text{wafer}}$ represent the wafer surface coordinates, and $\mathbf{x}_{\text{fpga}}$ represent the FPGA LUT delay information. Similarly, define y_{wafer} and y_{fpga} as the respective measurement values to be predicted.

The training and test datasets are defined as follows:

For wafer data:

$$(\mathbf{X}_{\text{wafer,train}}, \mathbf{y}_{\text{wafer,train}}) = \{(\mathbf{x}_{\text{wafer,i}}, y_{\text{wafer,i}})\}_{i=1}^{N},$$

$$\mathbf{X}_{\text{wafer,test}} = (\mathbf{x}^*_{\text{wafer,1}}, \dots, \mathbf{x}^*_{\text{wafer,M}}),$$

For FPGA data:

$$(\mathbf{X}_{\text{fpga,train}}, \mathbf{y}_{\text{fpga,train}}) = \{(\mathbf{x}_{\text{fpga,i}}, y_{\text{fpga,i}})\}_{i=1}^{P},$$

$$\mathbf{X}_{\text{fpga,test}} = (\mathbf{x}^*_{\text{fpga,1}}, \dots, \mathbf{x}^*_{\text{fpga,Q}}),$$

where $M \gg N$ and $Q \gg P$.

Using the predictive models $f(\cdot)$ and $g(\cdot)$ obtained from GPR, the predicted values for the test datasets are:

$$\mathbf{y}^*_{\text{wafer}} = (y^*_{\text{wafer,1}}, \dots, y^*_{\text{wafer,M}}),$$

$$\mathbf{y}^*_{\text{fpga}} = (y^*_{\text{fpga,1}}, \dots, y^*_{\text{fpga,Q}}).$$

3.1. Hybrid or Mixture Kernel Strategy

The Hybrid or Mixture Kernel approach combines multiple kernel functions, each tailored to specific data characteristics, into a composite kernel. This strategy traditionally uses fixed weights for individual kernels, expressed as:

$$K_{\text{composite}} = \sum_{i=1}^{n} w_i K_i,$$

where K_i are the individual kernels, and w_i are their static weights with $\sum w_i = 1$. However, such static weighting lacks adaptability to varying data properties, limiting robustness and flexibility.

To address this, an **adaptive hybrid kernel strategy** where weights w_i are dynamically adjusted based on kernel performance metrics.

3.1.1. Adaptive Kernel Strategy

The adaptive kernel strategy is designed to enhance GPR's robustness and adaptability for semiconductor data, such as wafer variations and FPGA delays. In this case, we introduce two of the adaptive strategies first we adapt three kernel with jitter in the 4th while the second approach consider top two candidates with adaptive weights. The data

$$D_1, D_2, \ldots, D_n$$

works with the kernel function

$$f(k_1), f(k_2), \ldots, f(k_n)$$

to determine the confidence of the error

$$\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_r,$$

to identify the candidates. This strategy dynamically selects or combines kernel functions based on the data's statistical properties, enhancing model accuracy across different production environments. The kernel combination involves forming a composite kernel by weighting multiple kernels:

$$K_{\text{adaptive}}(x, x') = w_1 \cdot K_{\text{Exp}}(x, x') + w_2 \cdot K_{\text{RQ}}(x, x') + w_3 \cdot K_{\text{MLP}}(x, x'), \quad (3)$$

where w_1, w_2, and w_3 are dynamically adjusted. It improves predictive accuracy by adjusting kernel selection and weighting to match data characteristics, enhancing model robustness across varied datasets, and reducing costs in semiconductor testing by minimizing error rates. A composite kernel is then formed by weighting multiple kernels according to:

$$K_{\text{adaptive}}(x, x') = w_1 \cdot K_{\text{Exp}}(x, x') + w_2 \cdot K_{\text{RQ}}(x, x') + w_3 \cdot K_{\text{MLP}}(x, x'), \quad (4)$$

where w_1, w_2, and w_3 are dynamically adjusted weights.

The composite kernel $K_{composite}$ is defined as a weighted sum of individual kernels:

$$K_{composite} = \sum_{i=1}^{4} w_i K_i \quad (5)$$

where K_i represents individual kernels (Exponential, MLP, RBF, and Matérn32) and w_i are adaptive weights. Let a_i represent their respective performance metrics, and $\delta_{ij} = |a_i - a_j|$ denote the performance difference between kernels i and j. Given a threshold τ for significant performance difference, the weights are determined as:

$$w_i = \begin{cases} 0.6 & \text{if } \max(\delta_{12}, \delta_{23}) > \tau \text{ and } i = \arg\max_k(a_k) \\ 0.35 & \text{if } \max(\delta_{12}, \delta_{23}) > \tau \text{ and } i = \arg\max_{k \neq \arg\max_k(a_k)}(a_k) \\ 0.4 & \text{if } \max(\delta_{12}, \delta_{23}) \leq \tau \text{ and } i = \arg\max_k(a_k) \\ 0.3 & \text{if } \max(\delta_{12}, \delta_{23}) \leq \tau \text{ and } i = \arg\max_{k \neq \arg\max_k(a_k)}(a_k) \\ 0.25 & \text{if } \max(\delta_{12}, \delta_{23}) \leq \tau \text{ and } i = \arg\max_{k \neq \{i_1, i_2\}}(a_k) \\ 0.05 & \text{if } i = 4 \text{ (jitter)} \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

where i_1 and i_2 are indices of kernels with highest and second-highest performance metrics. The final composite kernel is expressed as:

$$K_{composite} = w_1 K_{exp} + w_2 K_{MLP} + w_3 K_{RBF} + w_4 K_{M32} \quad (7)$$

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 4 March 2025 doi:10.20944/preprints202503.0260.v1

subject to the constraints:

$$\sum_{i=1}^{4} w_i = 1 \tag{8}$$

$$0 \leq w_i \leq 1 \tag{9}$$

On the other hand, a more simple adaption with only two kernels can also be defined where the kernel combination involves forming a composite kernel by weighting the top two kernels and including a jitter kernel as the third component:

$$K_{\text{adaptive}}(x, x') = w_1 \cdot K_{\text{Exp}}(x, x') + w_2 \cdot K_{\text{RQ}}(x, x') + w_3 \cdot K_{\text{jitter}}(x, x'), \tag{10}$$

where w_1, w_2, and w_3 are dynamically adjusted weights. Here, w_3 (jitter) ensures numerical stability. In this case he composite kernel $K_{composite}$ is defined as a weighted sum of individual kernels:

$$K_{composite} = \sum_{i=1}^{3} w_i K_i, \tag{11}$$

where K_i represents individual kernels (Exponential, MLP, and jitter) and w_i are adaptive weights. Let a_i represent their respective performance metrics, and $\delta_{ij} = |a_i - a_j|$ denote the performance difference between kernels i and j. Given a threshold τ for significant performance difference, the weights are determined as:

$$w_i = \begin{cases} 0.7 & \text{if } i = \arg\max_k(a_k) \\ 0.25 & \text{if } i = \arg\max_{k \neq \arg\max_k(a_k)}(a_k) \\ 0.05 & \text{if } i = 3 \text{ (jitter)} \\ 0 & \text{otherwise} \end{cases} \tag{12}$$

subject to the constraints:

$$\sum_{i=1}^{3} w_i = 1, \quad 0 \leq w_i \leq 1. \tag{13}$$

The final composite kernel is expressed as:

$$K_{composite} = w_1 K_{\text{Exp}} + w_2 K_{\text{RQ}} + w_3 K_{\text{jitter}}. \tag{14}$$

3.2. Propose Custom Adaptive Model

Before discussing the proposed custom adaptive strategy, we first analyze the correlation between variance and average error, as illustrated in Figures 5 and 6 where variance vs. average error scatter plot and cumulative mean of avg. error vs variance are illustrated for Matern52 and Linear Kernel respectively. In both figures, Gaussian Process Regression (GPR) is utilized to predict the input data using randomly selected 10% of the data for training, with error calculations as defined in Section 4. The "variance vs. avg. error scatter plot" and "cumulative mean of average error vs. variance" highlight this relationship in different scenarios. In the case of the Matern-52 kernel, the error remains consistently low, with the cumulative trend exhibiting a slight increase as variance grows. The scatter plot demonstrates a weak positive trend where higher variance corresponds to slightly lower errors. For the linear kernel, where the prediction uncertainty is higher, the correlation exhibits a stronger negative trend. In both cases, regardless of the kernel choice, the negative correlation establishes that lower variance is associated with lower error. A similar trend is also examined in wafer data. We examined a negative correlation between the cumulative average error and the predictive variance. Although in some cases, such as for certain wafer and FPGA paths, we observed a positive correlation

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 4 March 2025 doi:10.20944/preprints202503.0260.v1

for some kernels, the overall correlation averaged across the datasets was found to be negative in both cases.

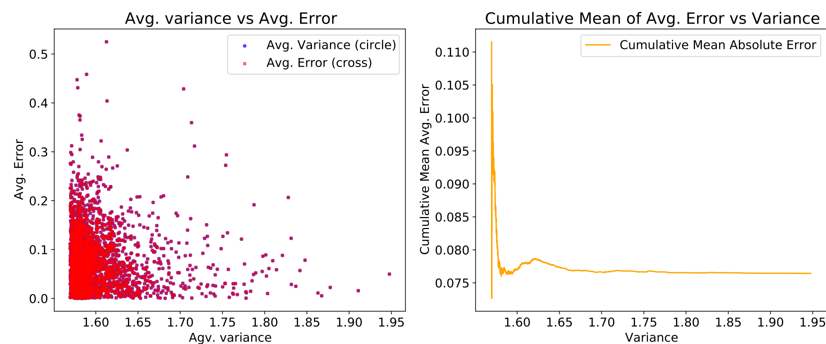


Figure 5. Variance vs. avg. error and cumulative mean of avg. error for Matern52 kernel.

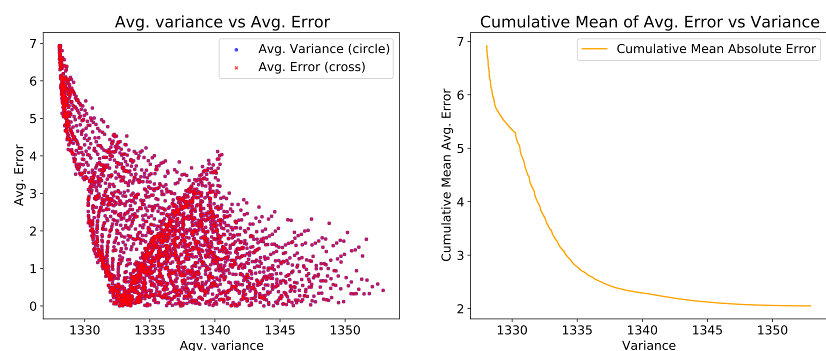


Figure 6. Variance vs. avg. error and cumulative mean of avg. error for linear kernel.

Considering this relationship, the proposed custom adaptive strategy is depicted in Figure 7. The figure illustrates how kernel weights are adapted based on confidence measurements derived from the variance of predictions. In this approach, the adaptive kernel weights are calculated based on predicted variance instead of during the estimation process.

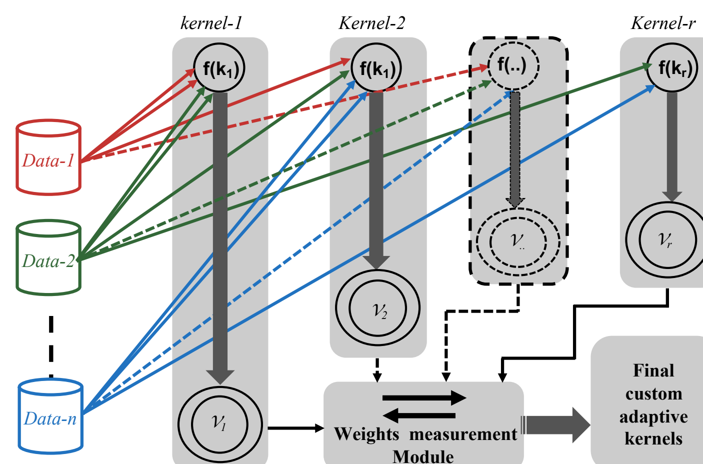


Figure 7. Proposed adaptive kernel methodology.

Figure 7 presents the different datasets $Data_1, Data_2, \ldots, Data_n$ with different kernel functions $fk_1, fk_2, \ldots, fk_r$. The Gaussian Process Regression (GPR) estimates the variance $V_1, V_2, \ldots, V_r$. for each kernel function. From these calculations, the top two candidates fk_{top1} and fk_{top2} are selected based on the lowest average variance among all kernels.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 4 March 2025 doi:10.20944/preprints202503.0260.v1

The weights w_1 and w_2 are defined to satisfy the following condition:

$$w_1 + w_2 = 1, \tag{15}$$

where the weights are assigned based on the difference in the variances fk_{top1} and fk_{top2}:

$$\Delta f = |fk_{top1} - fk_{top2}|. \tag{16}$$

The weight w_1 is calculated as:

$$w_1 = \begin{cases} 0.5, & \text{if } fk_{top1} = fk_{top2} \\ 0.5 + \beta \cdot \Delta f, & \text{if } fk_{top1} > fk_{top2} \\ 0.5 - \beta \cdot \Delta f, & \text{if } f_{ktop1} < fk_{top2} \end{cases} \tag{17}$$

where β is a scaling factor to control the impact of the variance difference on the weight.

Similarly, w_2 is given by:

$$w_2 = 1 - w_1. \tag{18}$$

Here if $fk_{top1} = fk_{top2}$, then $w_1 = w_2 = 0.5$ and as the variance difference Δf increases, w_1 increases if $fk_{top1} > fk_{top2}$ and decreases if $fk_{top1} < fk_{top2}$. The smaller the variance of fk_{top1} compared to fk_{top2}, the more w_1 dominates.

This approach ensures that the kernel with higher confidence receives greater weight, reflecting its importance in the combined kernel function. The proportional adjustment of w_1 and w_2 accommodates scenarios where one kernel significantly outperforms the other in terms of variance.

Figure 8 illustrates the relationship between w_1, w_2, and the variance difference Δf. It presents a general graph and another part demonstrating how the values are calculated and estimated with two variances, 10 and 5. According to the example in Figure 8, the weights w_1 and w_2 are calculated as 0.75 and 0.25, respectively.

The proposed weighting mechanism adjusts w_1 and w_2 based on the variance difference between two kernels. The hypothesis provides a robust framework for assigning weights, ensuring that kernels with higher variance contribute more to the combined kernel function.

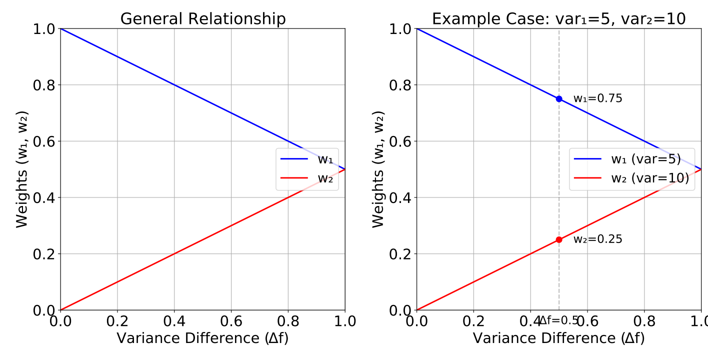


Figure 8. Adaptive kernel generator module : relationship between w_1, w_2, and the variance difference Δf.

4. Experimental Result

In this study, we conducted experiments in two different environments for two data types. A Linux workstation with Debian on an Intel Core i7 processor and 32 GB memory is used for the wafer data. On the other hand, a Windows 11 workstation with AMD architecture and 16 GB memory is used for the FPGA data. Table 1 shows the list of kernels used in each method and the proposed method.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 4 March 2025 doi:10.20944/preprints202503.0260.v1

Table 1. List of kernels for wafer and FPGA data.

Method	Wafer	FPGA
Adaptive	MLP, Exponential, RBF, OU	Matern52, RatQuad, Matern32, Exponential
Hybrid	MLP, Exponential	RatQuad, Matern52
Proposed	Exponential, RBF	Matern52, Exponential
Exhaustive	All supported kernels	All supported kernels

4.1. Experimental Setup

The detailed estimation methodology of the existing work and the proposed environment setup, including datasets for the FPGA and wafer data, the kernel list used, the framework, and the calculation of estimation errors, are explored in this section. The details about the all above are as follows:

4.1.1. Wafer Data

For wafer data, we conducted experiments using an industrial production test dataset of a 28 nm analog/RF device. Our dataset contains 150 wafers from six lots, each lot contains 25 wafers and a single wafer featuring approximately 6,000 devices under test (DUTs). In this experiment, we utilized a measured characteristic for an item of the dynamic current test, A heat map of the full measurement results for the first wafer of the sixth lot is displayed in Figures 1(a) and 9. For ease of experimentation, the faulty dies were removed from the dataset. In the data set, there are 16 sites in a single touchdown. However, in this experiment, the site information is not considered separately during the training phase. Although the data comes from a multisite environment, all 16 sites are treated as a single entity because the source data does not show any discontinuous changes for each site as shown in Figure 1(a) and 9

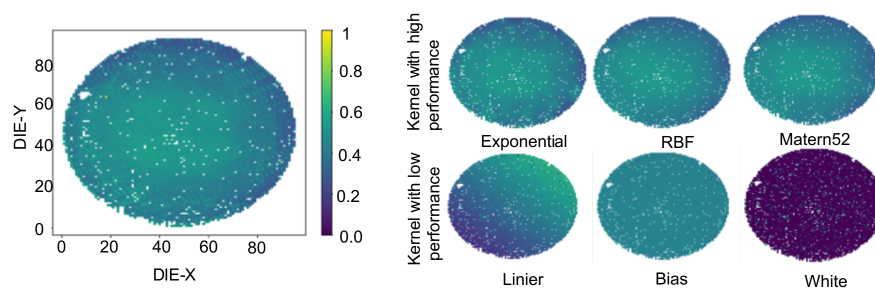


Figure 9. Comparison of heat maps for the measured value between the original and other kernels.

4.1.2. FPGA Data

For FPGA data, we conducted experiments using measurements using the Xilinx Artix-7 FPGA. In this experiment, 10 FPGAs (FPGA-01 to FPGA-10) were used. An on-chip measurement circuit has been implemented using 7-stages (i.e., 7 LUTs) in a Configurable Logic Block (CLB). The Ring Oscillator (RO) is based on XNOR or XOR logic gates. By keeping the same internal routing, logic resources, and structure for each RO placed in a CLB, the frequency variation caused by internal routing differences can be minimized. A total of **3,173** ROs were placed on a geometrical grid of 33×120 (excluding the empty space of the layout) through hardware macro modeling using the Xilinx CAD tool, Vivado [8,9]. Figure 2(a) **??** shows a single path's measured frequency heat-map. This style is followed to cover the total layout of the FPGA. The Artix-7 FPGA has 6-input LUTs, which create a total of 32 ($2^6 - 1$) possible paths for XNOR- and XOR-based RO configurations (path-01 to path-32). In total, to complete the exhaustive fingerprint (X-FP) measurement, 32 fingerprint measurements were conducted for all the 32 paths.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 4 March 2025 doi:10.20944/preprints202503.0260.v1

4.1.3. GPR Kernel

We use all supported kernels by the GPy framework, along with the proposed hybrid and adaptive kernels, for both wafer-level and FPGA-based data.

4.1.4. Estimated Prediction Error

To quantitatively evaluate the modeling accuracy, we defined the error (δ) between the correct (y_{true}) and the predicted mean (μ) normalized using the maximum and minimum values of $\mathbf{y}_{true}$ as follows:

$$\delta = \frac{\mu - y_{true}}{d_{spec}}, \tag{19}$$

where d_{spec} indicates the range between the minimum and maximum values of the fully measured characteristics.

4.1.5. Experimental analysis

Tables 2 and 3 shows the detailed evaluation of 150 wafers of six lots and 32 paths of 10 FPGAs respectively. Actual industry production data and actual silicon data are used in this evaluation. From the table it shows that wafer data's smooth patterns make it suitable for both Exponential and MLP kernels, while FPGA delay's complexity favors the Exponential kernel. MLP kernel's performance on FPGA data may be hindered by high sensitivity to hyperparameters, especially in higher dimensions.

Table 2. Accuracy of different kernels for wafer data.

Kernel	Lot1	Lot2	Lot3	Lot4	Lot5	Lot6	Average
Bias	0.1708	0.1842	0.1471	0.1596	0.1726	0.1996	0.1726
ExpQuad	0.1395	0.1132	0.0881	0.1158	0.1293	0.1343	0.1203
Linear	0.2399	0.2709	0.2393	0.2082	0.2587	0.2538	0.2451
Poly	0.1392	0.1201	0.0908	0.1157	0.1318	0.1372	0.1227
vRBF	0.1395	0.1132	0.0881	0.1158	0.1293	0.1343	0.1203
Exponential	0.1386	0.1038	0.0867	0.1183	0.1262	0.1249	0.1166
GridRBF	0.1395	0.1132	0.0881	0.1158	0.1293	0.1343	0.1203
Matern32	0.1397	0.1062	0.0862	0.1146	0.1282	0.1288	0.1175
Matern52	0.1404	0.1066	0.0869	0.1152	0.1290	0.1310	0.1184
MLP	0.1392	0.1027	0.0858	0.1145	0.1271	0.1283	0.1163
OU	0.1386	0.1038	0.0867	0.1183	0.1262	0.1249	0.1166
RatQuad	0.1398	0.1133	0.0863	0.1146	0.1292	0.1347	0.1199
StdPeriodic	0.1706	0.1807	0.1417	0.1547	0.1724	0.1996	0.1703
White	0.6938	0.8575	0.7049	0.6113	0.7696	0.7569	0.7324
Hybrid(MLP+Exponential)	0.1359	0.1037	0.858	0.1140	0.1263	0.1251	0.1153
Adaptive	0.169	0.1227	0.088	0.1188	0.1433	0.1351	0.1295
Proposed	0.1309	0.1066	0.0858	0.1040	0.1163	0.1251	0.1115

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 4 March 2025 doi:10.20944/preprints202503.0260.v1

Table 3. Accuracy of different kernels for FPGA data.

FPGA	FPGA-1	FPGA-2	FPGA-3	FPGA-4	FPGA-5	FPGA-6	FPGA-7	FPGA-8	FPGA-9	FPGA-10	AVG.
Bais	10.570	9.993	9.882	9.711	9.790	9.912	9.886	9.875	9.916	9.809	9.934
ExpQuad	9.050	8.590	8.584	8.515	8.508	8.479	8.412	8.326	8.293	8.229	8.499
Linear	216.582	205.011	206.836	204.727	204.727	205.799	205.085	203.605	203.242	201.628	205.724
vRBF	9.050	8.590	8.584	8.515	8.508	8.479	8.412	8.326	8.293	8.229	8.499
Exponential	8.863	8.437	8.420	8.355	8.385	8.350	8.309	8.232	8.186	8.113	8.365
GridRBF	9.050	8.590	8.584	8.515	8.508	8.479	8.412	8.326	8.293	8.229	8.499
Matern32	8.641	8.240	8.292	8.264	8.274	8.247	8.207	8.121	8.083	8.020	8.239
Matern52	8.646	8.235	8.224	8.175	8.178	8.145	8.104	8.027	7.983	7.913	8.163
MLP	11.714	16.503	15.706	17.086	16.748	15.688	14.898	14.539	14.043	14.666	15.159
OU	8.863	8.437	8.420	8.355	8.385	8.350	8.309	8.232	8.186	8.113	8.365
RatQuad	8.564	8.189	8.206	8.162	8.182	8.160	8.111	8.031	7.996	7.937	8.154
StdPeriodic	9.445	8.859	8.859	8.740	8.684	8.811	8.771	8.847	8.894	8.797	8.871
White	736.712	701.328	708.574	701.542	705.214	705.443	700.765	695.283	693.278	688.258	703.640
Hybrid	8.58	8.14	8.45	8.95	8.14	8.38	8.71	8.98	8.57	8.221	8.522
Adaptive	9.015	8.802	8.800	8.710	8.134	8.006	8.001	8.042	8.030	8.099	8.363
Proposed	8.815	8.402	8.110	8.102	8.104	8.101	7.992	8.032	7.920	7.946	8.152

The Exponential kernel is consistently strong across tasks, with vRBF, GridRBF, and ExpQuad performing well on wafer data, and RatQuad and Matern52 on FPGA data. White and Linear kernels under-perform on both, suggesting non-linearity is essential. The wafer task is generally easier, while FPGA delay prediction shows higher variability, indicating greater challenge.

For wafer data, Exponential and MLP kernels are suitable, with vRBF and ExpQuad as alternatives. For FPGA delays, the Exponential kernel is best, with RatQuad and Matern52 also worth considering. Avoid White and Linear kernels for both tasks.

The Hybrid kernel and the proposed adaptive kernels also perform well while in the proposed outperforms from all the case in terms of average and most of the case individual and lot wise average for both the wafer and FPGAs.

Figures 10 and 11 show the heat maps comparison for the wafer and FPGA, respectively. In the case of the wafer in Figure 10, the first wafer of the third lot is explored, while in the case of FPGA in Figure 11, the figure depicts the first path of the first FPGA with three good-performing and three bad-performing kernels.

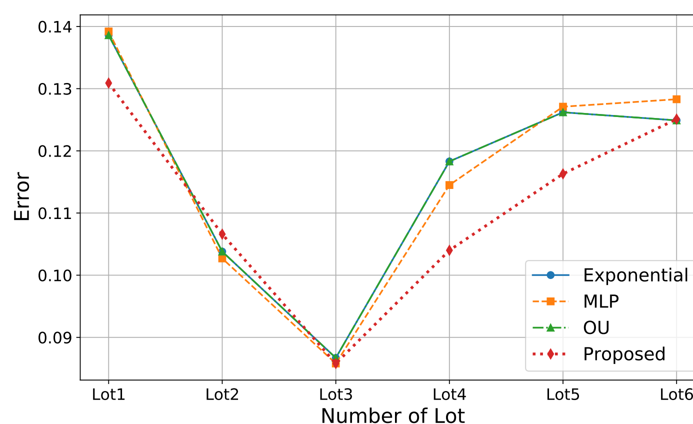
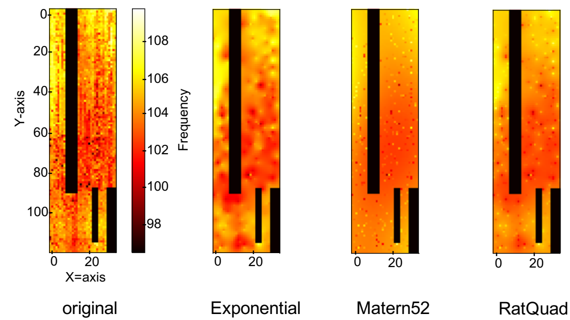
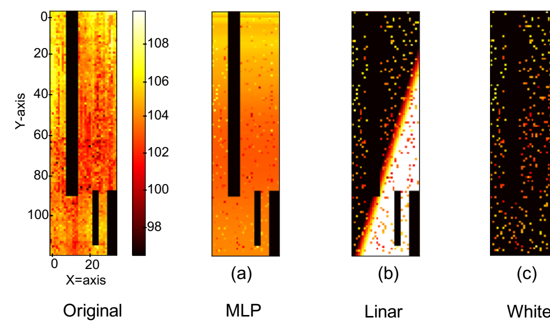


Figure 10. Average estimation error of the top four performing kernels.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 4 March 2025
doi:10.20944/preprints202503.0260.v1



(a) Path-1 of FPGA-1 high performing kernel.



(b) Path-1 of FPGA-1 low performing kernel.

Figure 11. Comparison of heat maps for the measured frequency between the original and other kernels.

In the wafer data, the exponential RBF and Matérn52 kernels very closely reproduce the original die, while in the case of bad performance, the linear, bias, and white kernels are also depicted in the corresponding heat maps. On the other hand, in Figure 11, the heat map of FPGA data shows that the exponential, Matérn, and TatQuad kernels are performing well, while MLP, linear, and white kernels are not performing well. From the figure, it is also clearly shown that in both cases of FPGAs and wafer data, the good-performing kernels are reflected in their corresponding heat maps as well.

The comparison of the top four performing kernels in both cases of wafer and FPGAs is depicted with a line graph in Figures 10 and 12, respectively. To find a common performer in both cases, only Proposed is outperforming in the two different environments.

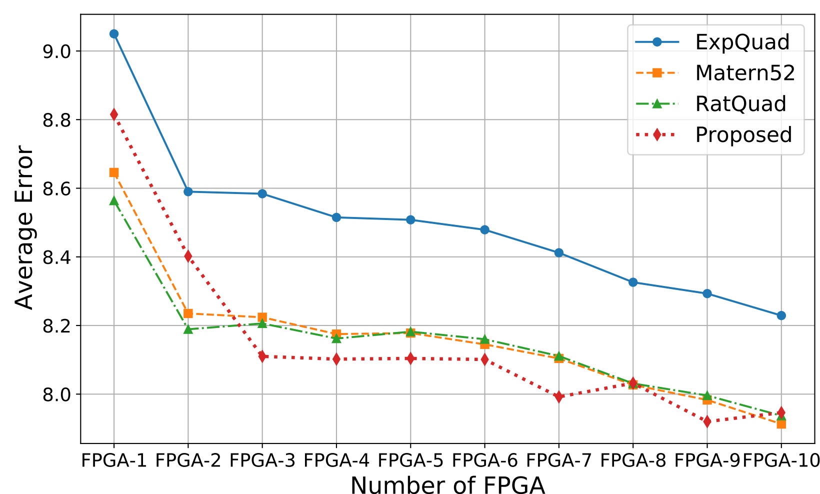


Figure 12. Average estimation error of the top four performing kernels.

Finally, Figure 13 illustrates the comparison of Wafer and FPGA accuracy across different kernel configurations to explore significant insights. For wafer accuracy measurements, the hybrid kernel

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 4 March 2025 doi:10.20944/preprints202503.0260.v1

achieves a low error rate of 0.115, which is comparable to the best kernel's error rate of 0.116. However, the *proposed* custom adaptive kernel demonstrates superior performance with the lowest Wafer error rate of 0.111.

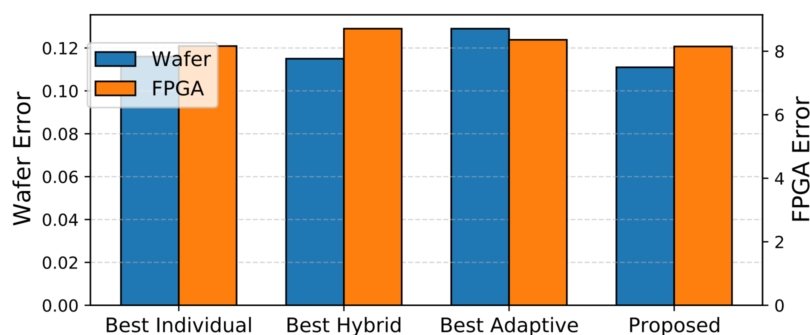


Figure 13. Overall estimation error comparison with the best Kernel with Hybrid, Adaptive, and proposed for FPGA and Wafer data.

In the FPGA dataset analysis also shown in Figure 13, the single best kernel slightly outperforms both the adaptive and hybrid kernels in baseline comparisons. Nevertheless, the *Proposed* custom adaptive kernel again demonstrates superior performance with the lowest FPGA error rate of 8.15, followed closely by the best kernel at 8.163 and the *adaptive* kernel with an error rate of 8.36. The hybrid kernel, however, exhibits notably higher FPGA error rates, indicating suboptimal performance for FPGA-based applications.

In the comprehensive analysis across both datasets, the *Proposed* kernel emerges as the most effective solution, consistently achieving superior performance. While the hybrid kernel demonstrates competitive accuracy in wafer-based applications, its significantly elevated FPGA error rate renders it unsuitable for universal deployment across both platforms.

5. Conclusions

This study investigated the application of Gaussian Process Regression (GPR), a state-of-the-art technique, for wafer-level variation modeling and FPGA delay prediction, with an emphasis on kernel function selection. Through exhaustive analysis, various kernel functions, including hybrid and adaptive kernels, were evaluated using wafer-level industry production data and silicon measurements from FPGAs. While no single kernel function consistently excelled across all datasets, our proposed custom adaptive kernel with a weighted kernel approach demonstrated superior versatility and reliability. This novel methodology enhances the predictive performance and adaptability of GPR, addressing limitations in current state-of-the-art applications. By offering robust accuracy across diverse scenarios, this approach presents a practical and innovative solution for optimizing semiconductor testing and production processes, with the potential to significantly reduce costs and improve efficiency. These contributions represent a meaningful advancement in leveraging GPR for semiconductor manufacturing and related domains.

Author Contributions: Riaz-ul-Haque Mian: Conceptualization, Methodology, Software, Visualization, Writing – Original Draft Preparation, Project Administration Foisal Ahmed: Data Curation, Validation, Writing – Review & Editing. Yoshito Hagihara: Experiment, Writing Yamane Souma: Validation, Formal Analysis. All authors have read and approved the final manuscript.

Funding: This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Data Availability Statement: This study utilized two datasets: 1. The FPGA dataset used in this study can be made available upon reasonable request from the corresponding author. Researchers requesting this dataset must provide a clear research purpose and will need to agree to terms of use. 2. The second dataset, which consists of

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 4 March 2025 doi:10.20944/preprints202503.0260.v1

industry production data, is subject to a non-disclosure agreement (NDA). Access to this dataset can be facilitated through direct communication with the industry contact, as provided by the corresponding author.

Conflicts of Interest: The authors declare that they have no competing interests.

References

1. Wan, Z.; Yu, B.; Li, T.Y.; Tang, J.; Zhu, Y.; Wang, Y.; Raychowdhury, A.; Liu, S. A survey of fpga-based robotic computing. *IEEE Circuits and Systems Magazine* **2021**, *21*, 48–74.
2. Riaz-ul-haque, M.; Michihiro, S.; Inoue, M. Hardware–Software Co-Design for Decimal Multiplication. *Computers* **2021**, *10*.
3. Nery, A.S.; Sena, A.C.; Guedes, L.S. Efficient Pathfinding Co-Processors for FPGAs. In Proceedings of the 2017 International Symposium on Computer Architecture and High Performance Computing Workshops (SBAC-PADW), 2017, pp. 97–102.
4. Shintani, M.; Mian, R.U.H.; Inoue, M.; Nakamura, T.; Kajiyama, M.; Eiki, M. Wafer-level Variation Modeling for Multi-site RF IC Testing via Hierarchical Gaussian Process. In Proceedings of the 2021 IEEE International Test Conference (ITC), 2021, pp. 103–112.
5. Violante, M.; Sterpone, L.; Manuzzato, A.; Gerardin, S.; Rech, P.; Bagatin, M.; Paccagnella, A.; Andreani, C.; Gorini, G.; Pietropaolo, A.; et al. A new hardware/software platform and a new 1/E neutron source for soft error studies: Testing FPGAs at the ISIS facility. *IEEE Transactions on Nuclear Science* **2007**, *54*, 1184–1189.
6. Bahukudumbi, S.; Chakrabarty, K. *Wafer-level testing and test during burn-in for integrated circuits*; Artech House, 2010.
7. Garrou, P. Wafer level chip scale packaging (WL-CSP): An overview. *IEEE Transactions on Advanced Packaging* **2000**, *23*, 198–205.
8. Ahmed, F.; Shintani, M.; Inoue, M. Accurate Recycled FPGA Detection Using an Exhaustive-Fingerprinting Technique Assisted by WID Process Variation Modeling. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* **2021**, *40*, 1626–1639.
9. Ahmed, F.; Shintani, M.; Inoue, M. Low Cost Recycled FPGA Detection Using Virtual Probe Technique. In Proceedings of the 2019 IEEE International Test Conference in Asia (ITC-Asia), 2019, pp. 103–108.
10. Ahmed, F.; Shintani, M.; Inoue, M. Feature engineering for recycled FPGA detection based on WID variation modeling. In Proceedings of the 2019 IEEE European Test Symposium (ETS). IEEE, 2019, pp. 1–2.
11. Wang, L.C. Experience of Data Analytics in EDA and Test—Principles, Promises, and Challenges. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* **2017**, *36*, 885–898.
12. Stratigopoulos, H.G. Machine learning applications in IC testing. In Proceedings of the Proceedings of IEEE European Test Symposium, 2018.
13. Shintani, M.; Inoue, M.; Nakamura, Y. Artificial Neural Network Based Test Escape Screening Using Generative Model. In Proceedings of the Proceedings of IEEE International Test Conference, 2018, p. 9.2.
14. Huang, K.; Kupp, N.; John M., Jr., C.; Makris, Y. Handling Discontinuous Effects in Modeling Spatial Correlation of Wafer-level Analog/RF Tests. In Proceedings of the Proceedings of IEEE Design Automation and Test in Europe, 2013, pp. 553–558.
15. Reda, S.; Nassif, S.R. Accurate Spatial Estimation and Decomposition Techniques for Variability Characterization. *IEEE Transactions on Semiconductor Manufacturing* **2010**, *23*, 345–357.
16. Li, X.; Rutenbar, R.R.; Blanton, R.D. Virtual probe: A statistically optimal framework for minimum-cost silicon characterization of nanoscale integrated circuits. In Proceedings of the Proceedings of IEEE/ACM International Conference on Computer-Aided Design, 2009, pp. 433–440.
17. Zhang, W.; Li, X.; Rutenbar, R.A. Bayesian Virtual Probe: Minimizing Variation Characterization Cost for Nanoscale IC Technologies via Bayesian Inference. In Proceedings of the Proceedings of ACM/EDAC/IEEE Design Automation Conference, 2010, pp. 262–267.
18. Zhang, W.; Li, X.; Liu, F.; Acar, E.; Rutenbar, R.A.; Blanton, R.D. Virtual Probe: A Statistical Framework for Low-Cost Silicon Characterization of Nanoscale Integrated Circuits. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* **2011**, *30*, 1814–1827.
19. Zhang, S.; Lin, F.; Hsu, C.K.; Cheng, K.T.; Wang, H. Joint Virtual Probe: Joint Exploration of Multiple Test Items' Spatial Patterns for Efficient Silicon Characterization and Test Prediction. In Proceedings of the Proceedings of IEEE Design Automation and Test in Europe, 2014.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 4 March 2025 doi:10.20944/preprints202503.0260.v1

20. Kupp, N.; Huang, K.; Carulli, Jr., J.M.; Makris, Y. Spatial correlation modeling for probe test cost reduction in RF devices. In Proceedings of the Proceedings of IEEE/ACM International Conference on Computer-Aided Design, 2012, pp. 23–29.
21. Ahmadi, A.; Huang, K.; Natarajan, S.; John M., Jr., C.; Makris, Y. Spatio-Temporal Wafer-Level Correlation Modeling with Progressive Sampling: A Pathway to HVM Yield Estimation. In Proceedings of the Proceedings of IEEE International Test Conference, 2014, p. 18.1.
22. Zhang, W.; Li, X.; Liu, F.; Acar, E.; Rutenbar, R.A.; Blanton, R.D. Virtual probe: A statistical framework for low-cost silicon characterization of nanoscale integrated circuits. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* **2011**, *30*, 1814–1827.
23. Donoho, D.L. Compressed sensing. *IEEE Transactions on Information Theory* **2006**, *52*, 1289–1306.
24. Candes, E.J.; Wakin, M.B. An Introduction To Compressive Sampling. *IEEE Signal Processing Magazine* **2008**, *25*, 21–30.
25. Shintani, M.; Mian, R.U.H.; Inoue, M.; Nakamura, T.; Kajiyama, M.; Eiki, M. Wafer-level Variation Modeling for Multi-site RF IC Testing via Hierarchical Gaussian Process. In Proceedings of the 2021 IEEE International Test Conference (ITC), 2021, pp. 103–112.
26. ul-haque MIAN, R.; NAKAMURA, T.; KAJIYAMA, M.; EIKI, M.; SHINTANI, M. Efficient Wafer-Level Spatial Variation Modeling for Multi-Site RF IC Testing. *IEICE TRANSACTIONS on Fundamentals of Electronics, Communications and Computer Sciences* **2023**, pp. 1139–1150.
27. Marinissen, E.J.; Singh, A.; Glotter, D.; Esposito, M.; Jr., J.M.C.; Nahar, A.; Butler, K.M.; Appello, D.; Portelli, C. Adapting to Adaptive Testing. In Proceedings of the Proceedings of IEEE Design Automation and Test in Europe, 2010, pp. 556–561.
28. Gotkhindikar, K.R.; Daasch, W.R.; Butler, K.M.; Carulli, Jr., J.M.; Nahar, A. Die-level Adaptive Test: Real-time Test Reordering and Elimination. In Proceedings of the Proceedings of IEEE International Test Conference, 2011, p. 15.1.
29. Yilmaz, E.; Ozev, S.; Sinanoglu, O.; Maxwell, P. Adaptive Testing: Conquering Process Variations. In Proceedings of the Proceedings of IEEE European Test Symposium, 2012.
30. Shintani, M.; Uezono, T.; Takahashi, T.; Hatayama, K.; Aikyo, T.; Masu, K.; Sato, T. A Variability-Aware Adaptive Test Flow for Test Quality Improvement. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* **2014**, *33*, 1056–1066.
31. Dempster, A.P.; Laird, N.M.; Rubin, D.B. Maximum Likelihood from Incomplete Data Via the EM Algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)* **1977**, *39*, 1–38.
32. GPy. GPy: A Gaussian process framework in python. http://github.com/SheffieldML/GPy, since 2012.
33. Xanthopoulos, C.; Huang, K.; Ahmadi, A.; Kupp, N.; Carulli, J.; Nahar, A.; Orr, B.; Pass, M.; Makris, Y., Gaussian Process-Based Wafer-Level Correlation Modeling and Its Applications. In *Machine Learning in VLSI Computer-Aided Design*; Elfadel, I.A.M.; Boning, D.S.; Li, X., Eds.; Springer International Publishing: Cham, 2019; pp. 119–173.
34. Ahmadi, A.; Huang, K.; Natarajan, S.; Carulli, J.M.; Makris, Y. Spatio-temporal wafer-level correlation modeling with progressive sampling: A pathway to HVM yield estimation. In Proceedings of the 2014 International Test Conference. IEEE, 2014, pp. 1–10.
35. Chang, C.; Chang, H.M.; Chiang, K. Study on Gaussian Process Regression to Predict Reliability Life of Wafer Level Packaging with cluster analysis. In Proceedings of the 2022 17th International Microsystems, Packaging, Assembly and Circuits Technology Conference (IMPACT), 2022, pp. 1–4.
36. Chang, C.; Zeng, T. A hybrid data-driven-physics-constrained Gaussian process regression framework with deep kernel for uncertainty quantification. *Journal of Computational Physics* **2023**, *486*, 112129.
37. Suwandi, R.C.; Lin, Z.; Sun, Y.; Wang, Z.; Cheng, L.; Yin, F. Gaussian Process Regression with Grid Spectral Mixture Kernel: Distributed Learning for Multidimensional Data. In Proceedings of the 2022 25th International Conference on Information Fusion (FUSION), 2022, pp. 1–8.
38. Bu, A.; Wang, R.; Jia, S.; Li, J. GPR-Based Framework for Statistical Analysis of Gate Delay under NBTI and Process Variation Effects. *Electronics* **2022**, *11*, 1336.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.