DATA-DRIVEN VARIATION MODELING AND MANAGEMENT WITH APPLICATION OF ADVANCED MANUFACTURING PROCESSES AND SYSTEMS

by

Jaesung Lee

A dissertation submitted in partial fulfillment of

the requirements for the degree of

Doctor of Philosophy

(Industrial and Systems Engineering)

at the

UNIVERSITY OF WISCONSIN–MADISON

2022

Date of final oral examination: 05/23/22

The dissertation is approved by the following members of the Final Oral Committee:

Shiyu Zhou (Chair), Professor, Industrial and Systems Engineering

Justin Boutilier, Assistant Professor, Industrial and Systems Engineering

Kaibo Liu, Associate Professor, Industrial and Systems Engineering

Garvesh Raskutti, Associate Professor, Statistics

Raj Veeramani, Professor, Industrial and Systems Engineering

ACKNOWLEDGMENTS

First and foremost, I want to express my sincere thanks to my advisor, Prof. Shiyu Zhou. I deeply appreciate his support and guidance throughout my Ph.D. study. I was very lucky to be supervised by Prof. Zhou. Without him, I would not have achieved any of these research works.

I would also like to thank my four other committee members, Prof. Justin Boutilier, Prof. Kaibo Liu, Prof. Garvesh Raskutti, and Prof. Raj Veeramani for their time and support.

I want to give my special thanks to my wife, Yoori Kim for going through a long and tough journey with me and for her endless support. I want to express my deep thanks to my parents, Heungseop Lee and Youngboon Shin, and my sister, Hyunjeong Lee for always being there and sending constant love and support. Because of them, I am here as myself. Last but not least, I would like to thank my little son, Liam Suho Lee, for being with me at this happiest moment. You came to me with the best of my luck. Without my family, I could not have gone through all these journeys.

I also want to say thanks to my dearest friends, Abhijeet Bhardwaj, Akash Deep, Congfang Huang, Salman Jahani, Minhee Kim, Raed Kontar, Chul Lee, Yunjia Lo, Junbo Son, Jinwen Sun, Chao Wang, Yiming Weng, and Honghan Ye for their support and going through this long journey of Ph.D. study with me. I am very pleased to have them all as my friends.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

Manufacturing variations refer to the uncertainties in the processes and inconsistency in the products produced. There have been increasing efforts to minimize the manufacturing variations, and reducing manufacturing variations in advanced manufacturing processes and systems is becoming more important. Advanced manufacturing processes and systems integrate manufacturing with innovative science and technologies and boost manufacturing efficiency and productivity. The integration with sensor technology now provides massive data, creating unprecedented research opportunities to model and analyze through data-driven models and methods. However, at the same time, advanced manufacturing processes and systems involve new critical challenges in modeling and managing the manufacturing variations. Many advanced manufacturing processes and systems have complex dynamics and transformation and multiple components involved, which create significant variations and uncertainties. However, physics-based models are often unavailable and often fail to address the uncertainties. This dissertation addresses multiple challenges listed below in modeling and managing the manufacturing variations:

1. **Variation source identification in multistage manufacturing systems:** In multistage manufacturing systems, where multiple operations are performed in a series of stages (e.g., workstations), the variations produced from operations propagate to downstream measurements. In such systems, it is crucial to identify faulty operations with excessive variations among a large number of

operations based on the quality measurements. We consider a common case where the measurements are not directly taken from the operations but from products in the downstream stage and the number of operations is much larger than the number of measurements. However, inferring underlying variations of numerous operations by limited measurements cause technical challenges in statistical inference. Therefore, we want to establish a statistical model that can identify faulty operations by leveraging the Engineering domain knowledge. Three types of domain knowledge are considered: 1) The fact that faults occur sporadically; 2) Practitioners' empirical knowledge of the faults occurrence frequency; 3) Various tolerance levels on variations across operations.

2. **Modeling inkjet printing manufacturing process:** The inkjet printing manufacturing process involves significant random variations due to the complex physical and chemical dynamics of the nanomaterial pieces in the printed ink. Process variations create significant uncertainties in the manufactured product quality, but such uncertainties have not been studied. Therefore, it is crucial to model the randomness in the manufacturing outcome in terms of process parameters. Building upon the statistical model, this work further aims to establish a statistic that evaluates the manufacturing outcome quality, and ultimately identifies abnormal manufacturing outcomes.

3. **Statistical calibration of underlying physical variable:** In designing manufacturing processes and products, inferring the underlying physical input variable, called statistical calibration, is often needed. For example, by using

the GFET nanosensor outputs, inferring the amount of the target substance in the environment is important. Furthermore, the uncertainty of the inferred variable needs to be quantified. However, due to significant process variations in manufacturing, the GFET nanosensor outputs involve significant random variations, and thus precise inferring is challenging. Specifically, random shapes and random locations of functional data need to be modeled for precise calibration.

4. **Optimal parameter design through Bayesian optimization:** It is very crucial to design manufacturing processes or products so that they have small quality variations while satisfying the overall quality (i.e., robust design). However, data are often costly to acquire especially in the designing stage. Furthermore, the underlying exact relationships between the design variables and the mean and variance of the outputs are not known and are in complex forms. Therefore, a sample efficient data-driven method to find the robust design needs to be established.

To address these challenges listed above, four problems are investigated in this dissertation. (*i*) *To build a special sparsity-enhanced Bayesian linear random-effects model to reflect Engineering domain knowledge.* With the proposed model, Engineering domain knowledge on sparse faults with excessive variations is incorporated into the model, and the variation sources are successfully identified. (*ii*) *To model the uncertainties in the inkjet printing manufacturing process in terms of physical process parameters.* Building upon the proposed model, abnormal manufacturing outcomes are successfully identified. (*iii*) *To establish a non-parametric model to characterize*

functional data with significant variations. The issue with random shapes and random shifting of functional data is addressed. (*iv*) *To establish sample-efficient stochastic constrained optimization method for constrained robust parameter design.* The proposed technique minimizes the variations while satisfying a constraint on the mean of the quality measurements by conducting a small number of experiments.

Because the proposed methods are driven by data, these models and methods are very flexible and can be used to address many general problems in other manufacturing processes.

1 INTRODUCTION

1.1 Motivation

Manufacturing variations refer to the uncertainties in the processes and inconsistency in the products produced. The production quality of a process is often defined as inversely proportional to the variations in the processes and products (Camelio et al., 2003). Therefore, understanding and managing the manufacturing variations are crucial. To maintain the high quality of manufacturing processes, there have been increasing efforts to minimize the variations from manufacturing processes and systems (Montgomery, 2012). Recent development in science and technologies have introduced new advanced manufacturing processes and systems, in which, manufacturing processes and products are integrated with innovative science and technologies and new materials; thereby, we can achieve low costs and large productivity and respond rapidly to customer demands (Science and Technology Policy Institute, 2010). Understanding and managing the variations of advanced manufacturing are thus becoming more important to fully realize their promises. Besides, modern data acquisition technologies embedded into the manufacturing processes and systems have provided unprecedented data, coming with research opportunities for data analytics. For example, many automotive industries have been transformed into automated smart manufacturing systems combined with robotics and sensor technologies. The data collected from the sensors can be exploited for effectively control the manufacturing variations. In nanomanufacturing, the inkjet printing technique is one of the additive nanosensor

manufacturing processes. By the the inkjet printing, components of nanosensors are printed on-demand very quickly and precisely. This printing technique is a key to the mass production of nanosensors; it scales up the fabrication process by removing the bottle-neck parts in conventional manufacturing processes. The nanomanufacturing data can be easily obtained from microscopic devices for the analysis.

Advanced manufacturing processes and systems provide significant opportunities. However, at the same time, there are critical challenges we are facing: (i) advanced manufacturing processes and systems are very complex and have not been studied deeply. Advanced manufacturing processes often involve complex physical and/or chemical dynamics and transformation. The complexity of advanced manufacturing systems comprising numerous operations often hinders investigating the variations and precise diagnosis of the excessive variations. Statistical modeling to understand the variations and methods to manage and control the process quality are crucial. (ii) Physics-based models are often not available to account for the processes and the uncertainties involved. Therefore, data-driven models and methods for controlling the process and quality and designing the processes and systems are urgently needed. (iii) However, the types of data obtained from advanced manufacturing processes and systems are very diverse; the examples of data types include image data and functional data. (iv) At the same time, engineering knowledge is often available, and such knowledge needs to be integrated into the data-driven models and methods to be established.

Because advanced manufacturing processes and systems have emerged recently,

only limited works are available on statistical modeling and methods that address the aforementioned challenges.

1.2 Research Objectives

This dissertation aims to analyze and manage manufacturing variations in advanced manufacturing processes and systems using statistical models and methods. Four research tasks are summarized as follows.

1. **Variation source identification in multistage manufacturing systems:** In multistage manufacturing systems, where multiple operations are performed in a series of stages (e.g., workstations), the variations produced from operations propagate to downstream measurements. In such a system, it is crucial to identify faulty operations with excessive variations among a large number of operations based on the quality measurements. We consider a common case where the measurements are not directly taken from the operations but from products in the downstream stage and the number of operations is much larger than the number of measurements. However, inferring underlying variations of numerous operations by limited measurements cause technical challenges in statistical inference. Therefore, we want to establish a statistical model that can identify the faulty operations by leveraging the Engineering domain knowledge. Three types of domain knowledge are considered: 1) The fact that faults occur sporadically; 2) Practitioners' empirical knowledge on the faults occurrence frequency; 3) Various tolerance levels on variations

across operations.

2. **Modeling inkjet printing manufacturing process:** The inkjet printing manufacturing process involves significant random variations due to the complex physical and chemical dynamics of the nanomaterial pieces in the printed ink. Process variations create significant uncertainties in the manufactured product quality, but such uncertainties have not been studied. Therefore, it is crucial to model the randomness in the manufacturing outcome in terms of process parameters. Building upon the statistical model, this work further aims to establish a statistic that evaluates the manufacturing outcome quality, and ultimately identifies abnormal manufacturing outcomes.

3. **Statistical calibration of underlying physical variable:** In designing manufacturing processes and products, inferring the underlying physical input variable, called statistical calibration, is often needed. For example, by using the GFET nanosensor outputs, inferring the amount of the target substance in the environment is important. Furthermore, the uncertainty of the inferred variable needs to be quantified. However, due to significant process variations in manufacturing, the GFET nanosensor outputs involve significant random variations, and thus precise inferring is challenging. Specifically, random shapes and random locations of functional data need to be modeled for precise calibration.

4. **Optimal parameter design through Bayesian optimization:** It is very crucial to design manufacturing processes or products so that they have small quality

variations while satisfying the overall quality (i.e., robust design). However, data is often costly to acquire especially in the designing stage. Furthermore, the underlying exact relationships between the design variables and the mean and variance of the outputs are not known and are in complex forms. Therefore, a sample efficient data-driven method to find the robust design needs to be established.

1.3 Outline of the Dissertation

The remainder of this dissertation focuses on the aforementioned problems in the application of advanced manufacturing processes and systems.

Chapter 2: *Variation Source Identification in Multistage Manufacturing Systems Using Bayesian Approach with Sparse Variance Components Prior*

Multistage manufacturing systems are modeled by a Bayesian linear random-effects model. Specifically, the product dimensions are modeled as observations with respect to the dimensional deviations of operations as random coefficients. The variances of the random coefficients are modeled by a special sparsity-enhanced Bayesian hierarchical prior distribution. The proposed prior distribution reflects the Engineering domain knowledge that the faults occur sporadically in a system and successfully addresses the challenge that the number of operations is much larger than the number of measurements. Furthermore, the proposed prior can incorporate the practitioners' empirical knowledge of the faults occurrence frequencies and effectively reflect the different tolerance levels for different operations. The

performance of the proposed model is demonstrated in extreme conditions in the numerical study and autobody assembly data in the case study.

Chapter 3: *Statistical Modeling and Analysis of k-Layer Coverage of Two-Dimensional Materials in Inkjet Printing Processes*

The inkjet-printed patterns, where random nanomaterial (specifically, 2d material) flakes are randomly dispersed, are modeled by a stochastic geometry model. In nanosensor manufacturing, both the random coverage and random thickness of the flakes determine the sensor performance; the random coverage and random thickness are represented by "k-layer coverage", defined by the proportion of the area covered exactly by the k layers of overlapping flakes. Then, based on the stochastic model, the mean and variance of the k-layer coverage are derived, and then the relationship between the 'mean and variance' of the k-layer coverage and the physical process parameters is investigated. Building upon the aforementioned results, a statistical framework that identifies abnormal printed inkjet-printed patterns based on image data is proposed. The performance of the proposed framework is validated with the real inkjet printed pattern image data.

Chapter 4: *Landmark-Embedded Gaussian Process Model With Applications for Functional Data Calibration*

A special Bayesian hierarchical Gaussian process model is proposed to model the functional data (i.e., the sensor outputs) whose shapes and locations are randomly determined. The proposed model successfully characterizes the random shapes and random locations by embedding the location features in a unified model. Then, a statistical calibration framework that infers both the point and interval estimates

of the unobserved input variable is proposed. The numerical and case studies demonstrate the beneficial aspects of our proposed model.

Chapter 5: *Robust Parameter Design on Dual Stochastic Response Models with Constrained Bayesian Optimization*

A novel stochastic constrained Bayesian optimization method is proposed for a robust parameter design problem, which is to find the optimal control variable values that minimize the quality variations of advanced Engineering systems (e.g., nanosensors) while satisfying their overall quality. In particular, the data at hand are very expensive to obtain and are generated under stochastic environments; under the same control variable values, the observed quality values are not the same. Special surrogate models and an acquisition function are established for stochastic constrained Bayesian optimization. The mean and variance of the quality values are represented by carefully designed Gaussian process models. The proposed acquisition function leverages both the prediction and quantified uncertainties of the surrogate models and explicitly encourages exploration over design space to find the global solution, not being satisfied by local optimums. Both the numerical and case studies confirm the performance of the proposed method.

Finally, Chapter 6 summarizes the contributions of this research work and discusses future work.

2 VARIATION SOURCE IDENTIFICATION IN MANUFACTURING PROCESSES USING BAYESIAN APPROACH WITH SPARSE VARIANCE COMPONENTS PRIOR *

We present a Bayesian linear random effects model for variation source identification in multistage manufacturing processes with a prior for sparse variance components. A modified horseshoe+ prior is used to tackle high dimensional problems with low sample size and sparse variation sources. Furthermore, we introduce the informed-horseshoe+ prior that incorporates the likelihood information of possible variation sources. To estimate the variations from the informed horseshoe+ prior, a specially-designed Gibbs sampler is established. Through a series of numerical experiments and case study, we showed that the proposed informed horseshoe+ outperforms the existing prior distributions when variation sources are sparse.

2.1 Introduction

Manufacturing variation refers to the uncertainties in the process and inconsistency in the products produced. The production quality of a process is often defined as inversely proportional to the process variation (Camelio et al., 2003). Thus, variation reduction in manufacturing processes is an essential objective of process quality control. On one hand, under normal working conditions, the process variation is

*This chapter is based on the paper: **Lee, J.**, J. Son, S. Zhou, and Y. Chen (2020). Variation source identification in manufacturing processes using bayesian approach with sparse variance components prior. *IEEE Transactions on Automation Science and Engineering*, 17(3), 1469–1485. https://doi.org/10.1109/tase.2019.2959605.

caused by the inevitable natural process variations, also called “common" causes in statistical process control literature. On the other hand, under abnormal conditions, the excessive process variation is caused by process faults or “variation sources". Take an example of the panel assembly process in Zhou et al. (2003). Fig. 2.1 shows the two panel assembly process via three stations, which is simplified from an autobody assembly process. At the first station, the panel 1 and 2 are assembled. At the station II, the panels 3 and 4 are attached to the previous subassembly that are combined at the station I. The whole assembled panel is inspected at the final station III. In assembly lines, one of the most important factors that plays a critical role in determining the dimensional integrity of the final assembly is the positioning accuracy of fixture locators (Zhou et al., 2003; Bastani et al., 2017, 2016; Huang et al., 2007). Deviation of the fixture locators leads to the dimensional deterioration of the final products. Under normal working conditions, the fixture locations are tight and accurate leading to very small dimensional variation in the final assembly. However, suppose a fixture is loosened and no longer able to fasten workpieces tightly at the station I. Then, the relative position of the panel 1 and 2, which is an important measure of dimensional quality, would have large variations and inconsistency among different products. In such case, the loosened fixture locator is considered as a variation source. Clearly, for effective variation reduction, we need to quickly identify the variation source and then make adjustments to rectify the issue. Generally, the target qualities (e.g., the relative positions of two panels in the above example) are called the Key Product Characteristics (KPCs). The underlying factors that impact on KPCs (e.g., the fixture locators) are called Key Control

Characteristics (KCCs). In most manufacturing processes, the KPCs are directly measured while many KCCs are not monitored closely due to the measurement cost and/or physical constraints. Thus, it is often necessary to identify the variation sources among KCCs based on the readily-available measurements of the KPCs.

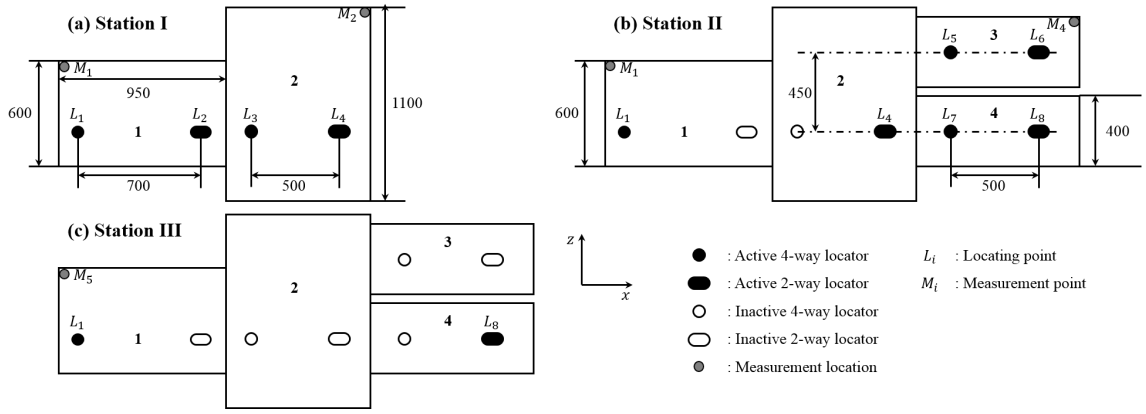


Figure 2.1: Illustration of a panel assembly process

Considering the importance of the problem, a large body of literature have dealt with variation modeling and variation source identification in manufacturing, particularly multistage processes. Although there are many methods available in the literature, the existing variation source identification approaches follow a similar three-step framework. First, the mathematical relationship between KPCs and KCCs are obtained through either the physical principles or data-driven approaches. In many cases, the exact relationship between KPCs and KCCs is nonlinear. However, most existing methods adopt a linear model because such nonlinear relationships can be approximated by a linear relationship using the Taylor series expansion. Second, based on the linear KPC-KCC relationship, the variance of the KCCs are estimated. The estimation can be done by various techniques, such as variation

pattern matching (e.g., Rong et al. (1999); Ding et al. (2002)) or direct statistical variation components estimation (e.g., Zhou et al. (2004); Apley and Shi (1998); Chang and Gossard (1998); Ding et al. (2005)). Lastly, the KCCs with excessive large variance are identified as variation sources. A comprehensive survey of the existing variation source identification approaches is provided in Shi and Zhou (2009).

Most of the existing works on variation source identification assume we have measurements on a large number of KPCs and the system is well-conditioned so there is no identifiability issues. In other words, it is assumed we can always uniquely estimate the variance of KCCs using the available rich KPC measurements. However, in practice, due to physical constraints or measurement cost, we may have measurements only on a very limited number of KPCs; hence, we may not be able to uniquely identify all the variation sources in the processes. Indeed, Zhou et al. Zhou et al. (2003) studied the diagnosability issue and established conditions under which the variation source is fully identifiable. The deficiency in measurement and the resulting lack of identifiability make most of available variation source identification methods vulnerable.

To resolve the intractability issue caused by measurement deficiency, we can consider the sparsity of the variation sources. The basic premise is that the existence of variation sources during a given period of time is sparse. This is a very reasonable assumption because the simultaneous occurrence of process errors are often low in practice. Despite the great potential, so far, the literature considering the sparsity in variation source identification has been scarce. The identifiability of variance

components under sparse condition is studied based on the concept of compressed sensing in Bastani et al. (2016). They utilized a measure for the sparsity to study sensor placements. The identification of the mean deviation fault (instead of the fault of excessive variation) under sparse condition is studied in Bastani et al. (2017). The sparse variation source identification method presented in Bastani et al. (2013) is the most relevant work to our study. The method can be characterized as a moment based method following several steps: first, from the relationship between KPCs and KCCs, the relationship among the variances of KPCs, KCCs, and measurement noises are established. Then, the authors treat the variance equations as a linear regression model. To find a sparse solution of the regression problem, the relevant vector machine (RVM) approach is used in the last step. One potential limitation of this approach is that the variance of the KPCs and KCCs are inherently non-negative. However, in their RVM approach, the normal distribution is adopted for the distribution of these variances, which may produce misleading negative values. Further, by converting the raw data into the second order moments and then conducting variation source identification, there is potential information loss. Our method proposed in this chapter can mitigate those issues by specifying proper distributions. Furthermore, our approach provides an opportunity to include additional information (domain knowledge) rather than incurring potential loss of information. Another relevant body of literature is pattern matching method based on principal component analysis (Ding et al., 2002; Li and Zhou, 2005; Li et al., 2007). Although the scope of the engineering problem they are trying to solve is similar to ours, it is easy to differentiate the rigorous Bayesian variation estimation approach

proposed in this chapter and the existing pattern matching methods. In pattern matching method, it is required to assume that there is only a single variation source in the process in order to employ the principal component analysis results. Comparing with that methods, our Bayesian approach can handle the situations of more than one variation sources and can incorporate prior knowledge in the identification. In addition, the Bayesian approach can provide the distribution information of the identification result, which is more statistically rigorous. Those advantageous features of the proposed method in contrast to the existing methods in the literature will be discussed later in detail.

In this chapter, we propose a new variation source identification approach considering sparse conditions. The goal is to identify the KCCs that have excessive variations based on the measurements on KPCs. We adopt the linear random effects model to describe the relationship among KPCs and KCCs and then establish a Bayesian approach to fully utilize *a priori* knowledge. A sparsity-enhanced prior distribution for the variance component that is well-suited for our model is developed. Unlike the typical choices of the prior distribution in Bayesian variance estimation, the prior distribution used in our proposed approach has a high density at zero. Consequently, during the inference, the algorithm tends to push estimates to be zero; hence, the sparsity in the estimation is naturally encouraged. Furthermore, noting that practitioners often have domain-specific empirical knowledge on how likely different variational faults may occur, we design the prior distribution in a way that such empirical knowledge can be taken into consideration. This is one of the main strengths of the Bayesian approach. To facilitate efficient inference, we

established a specially-designed Gibbs sampler for our model. Comprehensive numerical study and a real case study are also performed to illustrate the effectiveness of the proposed approach.

The rest of the chapter is organized as follows. In Section 2.2, we present the mathematical framework of Bayesian linear random effects model with sparse variance components. The Gibbs sampler is then proposed in Section 2.3 to illustrate how to estimate the parameters in the model. In Section 2.4, we evaluate the performance of the proposed method through a series of numerical studies, then the case study results based on a panel assembly multistage process are presented to validate our method in Section 2.5. Lastly, Section 2.6 concludes the chapter.

2.2 Bayesian Linear Random Effects Model for Variation Source Identification

2.2.1 Bayesian Linear Random Effects Model

Linear random effects model has been widely used for variation source identification (Apley and Shi, 1998; Chang and Gossard, 1998; Zhou et al., 2004). In the linear random effects model, process variations in KCCs are represented in terms of the variances of the random effects coefficients. A general linear random effects model is given as follows:

$$\mathbf{y}_i = \mathbf{A}\mathbf{x}_i + \boldsymbol{\epsilon}_i, \tag{2.1}$$

$$\mathbf{x}_i \sim \mathcal{N}\left(\mathbf{0}, \mathbf{\Sigma}_{\mathbf{x}}\right), \tag{2.2}$$

$$\mathbf{\Sigma}_{\mathbf{x}} = \text{diag}\left(\sigma_1^2, \ldots, \sigma_P^2\right), \tag{2.3}$$

$$\boldsymbol{\epsilon}_i \sim \mathcal{N}\left(\mathbf{0}, \sigma_\varepsilon^2 \, \mathbf{I}_{[N\times N]}\right) \tag{2.4}$$

where $i = 1, \ldots, M$ and $\mathbf{y}_i = (y_{1i}, \ldots, y_{Ni})^T \in \mathbb{R}^N$ is the KPC measurement vector for the i-th product. For each product i, N KPCs are observed. $\mathbf{A} = (a_{jk})_{j=1,\ldots,N,k=1,\ldots,P} \in \mathbb{R}^{N\times P}$ is the design matrix for the linear model, which is derived from the physical structure of the process. Because the products are produced by the same process, we assume that every product shares the same design matrix $\mathbf{A}$. $\mathbf{x}_i = (x_{1i}, \ldots, x_{Pi})^T$ is a random errors vector for the i-th product, which is a coefficient vector of random effects that follow the normal distribution with mean $\mathbf{0}$ and the variance-covariance matrix $\mathbf{\Sigma}_{\mathbf{x}}$. Since the variations are of our main concern, the fixed effects coefficients are set as $\mathbf{0}$. The process errors (e.g., wears of different locators in a fixture system) can often be assumed as independent (Zhou et al., 2003); hence, $\mathbf{\Sigma}_{\mathbf{x}}$ is a diagonal matrix where k-th diagonal element is σ_k^2 and non-diagonal elements are zeros. The diagonal elements of $\mathbf{\Sigma}_{\mathbf{x}}$ is also called variance components. In fact, variation sources in the process are represented by these variance components. If they are excessively large, we say we have significant variation sources in the process. $\boldsymbol{\epsilon}_i = (\varepsilon_{1i}, \varepsilon_{2i}, \ldots, \varepsilon_{Ni})^T$ is the measurement error vector where ε_{ji} is independent and identically distributed normal random variable with mean 0 and variance σ_ε^2. This model has been widely used in variation source identification and has been proven to be quite effective (Zhou et al., 2003).

We also want to point out that in practice, different tolerance requirements may

exist for different KCCs. In such case, variance of a faulty variation source may be smaller than those of the normal ones if its tolerance allowance is much tighter than others. In these scenarios, we need to transform the model in (2.1) to make the scales of the variation sources identical to each other. Such model standardization can be easily achieved as follows. Denote t_k as the tolerance level of the variation of k-th KCC. When $\sigma_k > t_k$, the k-th source is regarded as faulty variation source. The model of (2.1) can be transformed as below.

$$\mathbf{y}_i = \left(\mathbf{A}\mathbf{T}^{-1}\right)(\mathbf{T}\mathbf{x}_i) + \boldsymbol{\epsilon}_i \tag{2.5}$$

$$\mathbf{T} = \text{diag}\left(\frac{1}{t_1}, \ldots, \frac{1}{t_P}\right) \tag{2.6}$$

where $\mathbf{T} \in \mathbb{R}^{P\times P}$ is a diagonal matrix with its k-th diagonal element being $1/t_k$. It is easy to see that the transformed model still can be written in the same form of (2.1) as

$$\mathbf{y}_i = \mathbf{A}'\mathbf{x}'_i + \boldsymbol{\epsilon}_i \tag{2.7}$$

$$\mathbf{x}'_i \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}'_{\mathbf{x}}) \tag{2.8}$$

$$\boldsymbol{\Sigma}'_{\mathbf{x}} = \text{diag}\left({\sigma'_1}^2, \ldots, {\sigma'_P}^2\right) \tag{2.9}$$

where $\mathbf{A}' = \mathbf{A}\mathbf{T}^{-1}$, $\mathbf{x}'_i = \mathbf{T}\mathbf{x}_i$, and $\sigma'_k = \sigma_k/t_k$.

In above model, σ'_k represents the relative standard deviation against the tolerance level of k-th KCC. Under this model, the k-th variation source should be identified as a faulty source when the estimate of σ'_k is larger than 1 because $\hat{\sigma}'_k > 1$ is equivalent to $\hat{\sigma}_k > t_k$. In this way, variations from all the KCCs can remain on the

same scale. The assumption that any non-faulty KCCs will have smaller variances than that of the faulty KCCs will be satisfied. Without loss of generality, we will simply use model in (2.1) as the underlying model in the rest sections.

In most existing variation source identification methods, researchers have been using conventional approaches such as maximum likelihood estimation or moment based approach to estimate the variance components $(\sigma_1^2, \ldots, \sigma_P^2)$ based on the observation $\mathbf{y} = (\mathbf{y}_1^T, \ldots, \mathbf{y}_M^T)^T$. A good review and comparison of those approaches can be found in Ding et al. (2005).

In this work, we propose to use a Bayesian approach for variation source identification instead. In a Bayesian model, we treat the parameters to be estimated as random variables following certain prior distributions. In our case, we have

$$(\sigma_\varepsilon^2,\ \sigma_1^2, ..., \sigma_P^2) \sim \pi\,(.) \tag{2.10}$$

where $\pi\,(.)$ is the prior distribution for σ_ε^2 and $(\sigma_1^2, ..., \sigma_P^2)$. Given the prior, we can find the posterior distribution $P\,(\sigma_\varepsilon^2, \sigma_1^2, \ldots, \sigma_P^2|\mathbf{y})$ through the Bayes theorem as

$$P\left(\sigma_\varepsilon^2, \sigma_1^2, \ldots, \sigma_P^2 \middle| \mathbf{y}\right) \tag{2.11}$$

$$= \frac{P\,(\mathbf{y}, \sigma_\varepsilon^2, \sigma_1^2, \ldots, \sigma_P^2)}{\int\cdots\int\limits_{\sigma_\varepsilon^2,\sigma_1^2,...,\sigma_P^2} P\,(\mathbf{y}, \sigma_\varepsilon^2, \sigma_1^2, \ldots, \sigma_P^2) d\sigma_\varepsilon^2\, d\sigma_1^2 \ldots d\sigma_P^2} \tag{2.12}$$

$$= \frac{P\,(\mathbf{y}|\sigma_\varepsilon^2, \sigma_1^2, ..., \sigma_P^2)\,\pi\,(\sigma_\varepsilon^2,\ \sigma_1^2, ..., \sigma_P^2)}{\int\cdots\int\limits_{\sigma_\varepsilon^2,\sigma_1^2,...,\sigma_P^2} P\,(\mathbf{y}|\sigma_\varepsilon^2, \sigma_1^2, ..., \sigma_P^2)\,\pi\,(\sigma_\varepsilon^2,\ \sigma_1^2, ..., \sigma_P^2) d\sigma_\varepsilon^2 ... d\sigma_P^2} \tag{2.13}$$

We estimate the parameters by taking the mean of the posterior distribution

(e.g., mean values of $\sigma_1^2, ..., \sigma_P^2$ under the distribution $P\left(\sigma_\varepsilon^2, \sigma_1^2, \ldots, \sigma_P^2 | \mathbf{y}\right)$) through sampling or analytical evaluation. A typical challenge in Bayesian approaches for a complex problem is how to sample from the posterior distribution because, in most cases, an explicit expression of the posterior mean is not available. Therefore, we established an efficient algorithm for sampling from (2.11) which is illustrated later in the section 2.3.

One of the prominent advantages of the Bayesian model is the capability of specifying the prior distribution freely to enhance the sparsity of the solution. Also, it allows us to utilize *a priori* knowledge into the model. In Section 2.2.2, we propose a prior distribution, namely the informed horseshoe+ distribution, for σ_ε^2 and $(\sigma_1^2, ..., \sigma_P^2)$. This prior can take some empirical domain knowledge of variation source occurrences and enhance the solution sparsity.

2.2.2 Prior Distributions Considering Sparsity: Informed Horseshoe+ Prior

In the literature, the inverse gamma prior is one of the widely-used prior distributions for variances in Bayesian estimation scheme. For example, the inverse gamma prior has been used for the variance components in a non-sparse Bayesian linear random effects model in Zeng and Chen (2015) to monitor the means and variance components of the optical profiles in low-E glass manufacturing processes. The inverse gamma is a conjugate prior for the variances of normally-distributed random variables, which means the posterior distribution is in the same distribution family. The inverse gamma distribution has two parameters: shape and scale parameters.

The left panel of Fig. 2.2 illustrates the inverse gamma distributions with different parameter specifications. One obvious feature of the inverse gamma distribution is that the density of the inverse gamma at zero is always zero. In other words, the samples from the distribution will never get zero values. Using the inverse gamma prior forces the estimated variance components to be away from zero. This characteristics directly conflict to our rationale of sparse variance estimation, where we want to assign zero to most of the variance components and only keep a very small number of nonzero components.

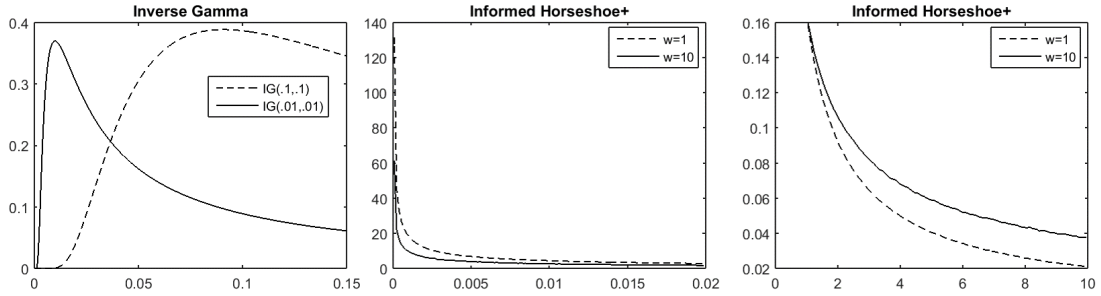


Figure 2.2: Probability density functions of variance component σ_k^2 from the inverse gamma and informed horseshoe+ priors

Keeping a small number of nonzero variance component (ensuring sparsity) is crucial. To illustrate its importance, let $\mathfrak{F}$ denotes the index set of the faulty variation sources and $\mathfrak{N}$ is that of the non-faulty sources. If we note that the probability of multiple process faults occurring simultaneously is very small, we have $\sigma_k^2 \gg 0$ for $k \in \mathfrak{F}$ and $\sigma_k^2 \gg \sigma_h^2 > 0$ for $k \in \mathfrak{F}$, $h \in \mathfrak{N}$ with $|\mathfrak{F}| \ll |\mathfrak{N}|$. The faulty variation sources (i.e. the number of faulty sources with large variance) are sparse and the non-faulty sources still have their own non-zero normal process noises. However, the variability of the non-faulty sources is considerably smaller than the variability

of the faulty sources. The sparsity in the solution allows us to identify the variation sources under some extreme conditions. Specifically, if the number of potential variation sources is much larger than the number of KPCs to be measured, i.e., $P \gg N$, the variation sources are generally not identifiable. However, under sparsity condition (e.g., only a few variation sources are present), we can uniquely identify them.

Although it may have not been popular in variance source identification literature, the Bayesian sparse parameter estimation has been used in various fields. Most of the existing Bayesian sparse estimation works focus on the variable selection and mean estimation problem (Park and Casella, 2008; Carvalho et al., 2010; Bhadra et al., 2017; Li and Chen, 2016). To take the sparsity into consideration, it is important to construct appropriate prior distributions. There are two main types of priors considering sparsity: discrete mixtures and shrinkage priors (Carvalho et al., 2009). Both priors try to create nonzero probability mass or density at zero. The first method achieves the goal through a mixture distribution with point mass at zero and the second method uses a continuous density function with positive density at zero. The discrete mixture prior is conceptually simple but computational challenging. The shrinkage prior is relatively complex to construct but more attractive computationally. In this work, we propose a shrinkage type prior distribution for the linear random effects models. The proposed prior distribution is inspired by the horseshoe+ distribution that has been used in Bayesian variable selection in regression (Bhadra et al., 2017). We significantly extend the horseshoe+ distribution by adding more flexibility and apply it to variance component estimation with sparsity.

We called this prior distribution as the “informed horseshoe+" (IHS+) distribution. The Bayesian linear random effects model with the informed horseshoe+ prior is presented below.

$$
\begin{aligned}
y_{ji}|\mathbf{x}_{ki}, \sigma_\varepsilon^2 &\sim \mathcal{N}\left(\sum_{k=1}^{P} a_{jk}x_{ki},\ \sigma_\varepsilon^2\right) \\
&\quad , \ j = 1, \ldots, N,\ i = 1, \ldots, M \\
x_{ki}|\sigma_k^2 &\sim \mathcal{N}\left(0, \sigma_k^2\right),\ k = 1, \ldots, P, \quad i = 1, .., M \\
\sigma_k|\tau, \eta_k &\sim \mathcal{C}^+\left(\tau\eta_k\right),\ k = 1, \ldots, P \\
\tau|\sigma_\varepsilon &\sim \mathcal{C}^+\left(\sigma_\varepsilon\right) \\
\eta_k|w &\sim \mathcal{C}^+\left(w\right),\ k \in \Omega \\
\eta_k &\sim \mathcal{C}^+\left(1\right),\ k \in \{1, \ldots, P\} \setminus \Omega \\
\sigma_\varepsilon &\sim \mathcal{C}^+\left(1\right)
\end{aligned}
\tag{2.14}
$$

where $\mathcal{C}^+$ is the half Cauchy distribution and Ω is a prioritized index set that contain indexes of fault-likely variation sources based on domain knowledge. The first two rows in (2.14) specify the linear random effects model that is consistent with that in (2.1). y_{ji} is a measurement for the product i from the sensor j. a_{jk} is the $(j,\ k)$ th element of the design matrix $\mathbf{A}$. x_{ki} is a random effect coefficient for the KCC k of the product i, which is a normal random variable with mean 0 and variance component σ_k^2. σ_k^2 and σ_ε^2 are the variance components of the random effects coefficients and the measurements, respectively.

Added to the linear random effects model, the priors for σ_k^2 and σ_ε^2 are specified in

the rest 5 rows in (2.14). For each KCC k, the standard deviation σ_k follows the half Cauchy distribution with a scale parameter of the multiplication of τ and η_k, where τ and η_k are called the global shrinkage and local shrinkage parameters, respectively. The parameter τ follows the half Cauchy distribution with a scale parameter σ_ε. The parameter η_k follows the half Cauchy distribution with a scale parameter w when $k \in \Omega$. Otherwise, η_k follows the standard half Cauchy distribution. The σ_ε follows the standard half Cauchy distribution. The hierarchical structure of the specified prior distribution looks complicated. A graphical representation of the structure is shown in Fig. 2.3, where a square represents a constant, a circle represents a random variable, and a directed line indicates the dependence of the variables.

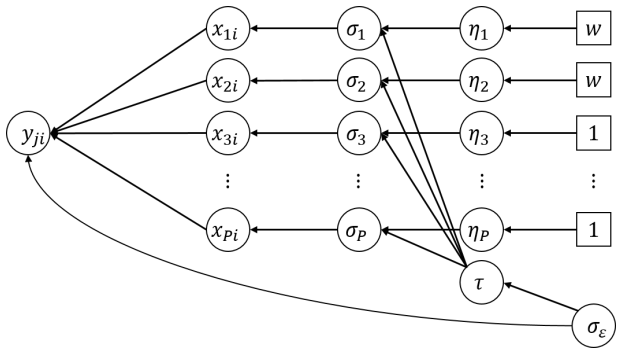


Figure 2.3: Graphical representation of our model

A good prior distribution for our problem should possess the following features: 1) It should have a large density at zero so the shrinkage, as well as the sparsity, is enforced; 2) It should have flexibility to adjust the sparsity so we may take some prior knowledge into consideration; 3) There should be an efficient way to compute or sample from the posterior distribution. The proposed informed horseshoe+ distribution indeed has those features. First, as shown in the middle and right

panels of Fig. 2.2, with the proposed informed horseshoe+ prior distribution, the marginal distribution of the variance components has a positive density at zero and the magnitude of the density can be adjusted. In other words, the variance component estimation can be sparse. Second, the informed horseshoe+ distribution is a "shrinkage" type prior distribution and is quite flexible to adjust the level of sparsity or the strength of the shrinkage. As shown in Fig. 2.3, τ is a global parameter that influences the distribution of all the variance components $\sigma_1^2, ..., \sigma_P^2$, while $\eta_1, ..., \eta_P$ are local parameters that influence the distribution of individual variance components. As shown in (2.14), the standard deviations $\sigma_1, ..., \sigma_P$ follow the half Cauchy distributions with distribution parameters determined by τ and η_k. The probability density function of the half Cauchy distribution is shown in Fig. 2.4. When the distribution parameter decreases, the density around zero becomes thicker. Therefore, if τ is small enough (i.e., close to zero), all the variable components $\sigma_1^2, \ldots, \sigma_P^2$ tend to have almost zero values except for the ones with local shrinkage parameter η_k large enough to offset the shrinkage of τ. Intuitively, τ can be viewed as the penalty parameter in the lasso and ridge regression in terms of its role in overall shrinkage enforcement. A smaller τ leads to stronger shrinkage and more sparsity in the solution. On the other hand, $\eta_1, ..., \eta_P$ can be viewed as local shrinkage parameters that allow the individual adjustment of the shrinkage for each variance component. A larger η_k value indicates the corresponding standard deviation σ_k is likely to be nonzero while a smaller η_k value indicates σ_k is likely to be zero. In the informed horseshoe+ prior distribution, instead of assigning some fixed values to τ and $\eta_1, ..., \eta_P$, we assume that τ and $\eta_1, ..., \eta_P$ also follow the half Cauchy

distribution, which further improves the model flexibility. Specifically, τ follows the half Cauchy distribution with the parameter determined by σ_ε. This model structure can avoid the distribution of the variance components being ill-posed with a shape of bimodal density (Polson and Scott, 2011). Regarding the distribution of the local shrinkage parameters $\eta_1, ..., \eta_P$, we allow to assign a fixed distribution parameter either 1 or w. If, based on domain knowledge, we believe the occurrence of variations at certain sources is more likely than others, then we can assign a relatively large parameter w to the corresponding η_k. With a large w, we will get a larger η_k; hence, leading to a nonzero variance component estimation. Through this arrangement, the prior knowledge can be incorporated in the informed horseshoe+ distribution. Third, we can implement the efficient Gibbs sampling scheme to sample from the posterior distribution for the variance component estimation with the informed horseshoe+ prior distribution. The technical details of the specially-designed Gibbs sampler can be found in section 2.3.

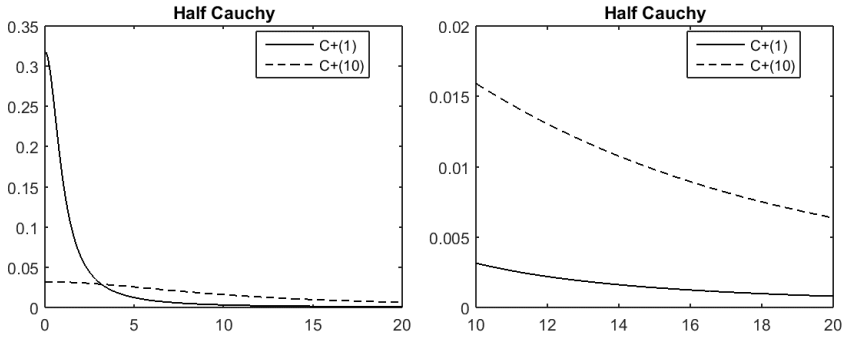


Figure 2.4: Probability density functions of half Cauchy distribution

From above discussion, it can be seen that the proposed informed horseshoe+ distribution possesses several desirable features for Bayesian sparse estimation. Be-

cause the proposed prior distribution is an extension of the conventional horseshoe+ distribution (Bhadra et al., 2017), we point out some key differences between these two. In the conventional horseshoe+ distribution, all the shrinkage parameters including τ and $\eta_1, ..., \eta_P$ are assigned to follow the standard half Cauchy distribution with parameter 1. The simple structure of the existing horseshoe+ prior distribution has been appreciated especially for variable selection in regression models. The proposed informed horseshoe+ prior distribution has different parametrization to ensure more flexibility for sparse variance component estimation in random effects models. Due to the structural differences and distinctive objectives, the resulting posterior distributions and the corresponding sampling strategy of the proposed prior distribution in this chapter are noticeably different.

2.3 Inference Based on Gibbs Sampler for the Informed Horseshoe+

In this section, we present the efficient Gibbs sampling strategy for the posterior distribution derived from the informed horseshoe+ prior distribution. Our parameters of interest are the variance components $\{\sigma_k\}$. The joint distribution of our

model is

$$
\begin{aligned}
&P\left(\mathbf{y}, \sigma_{\varepsilon}, x_{11}, \ldots, x_{ki}, \ldots, x_{PM}, \sigma_1, \ldots, \sigma_P, \eta_1, \ldots, \eta_P, \tau\right) \\
&= \prod_{i=1}^{M} P\left(\mathbf{y}_i | \sigma_{\varepsilon}, \{\mathbf{x}_i\}_{i=1}^{M}\right) \\
&\quad \prod_{k=1}^{P} \left\{ \prod_{i=1}^{M} \left\{ P\left(x_{ki} | \sigma_k\right) \right\} P\left(\sigma_k | \eta_k, \tau\right) P\left(\eta_k\right) \right\} P\left(\tau | \sigma_{\varepsilon}\right) P\left(\sigma_{\varepsilon}\right)
\end{aligned}
\tag{2.15}
$$

where $\mathbf{y}$ represents the aggregated KPC measurements from M products. A closed form expression of the posterior distribution of the variance components is not tractable due to the complex joint distribution. Therefore, sampling method such as Markov chain Monte Carlo (MCMC) is considered.

Gibbs sampler is a MCMC method that is to sample from the posterior distribution (Hobert and Casella, 1996). In order to do the Gibbs sampling, we need to derive the full conditional probability distribution, which is a conditional probability distribution of the variable given the rest of all the variables, for every variables included in the distribution function. Then, for each variable, we draw a sample from its full conditional probability distribution. After the dispersion of the samples converge to a distribution, the converged distribution of the sample is the posterior distribution of the parameter. Gibbs sampler is more efficient than other sampling methods such as rejection sampling or Metropolis algorithm, because of its 100% acceptance rate of samples (Gelman et al., 2013b).

Gibbs sampler requires two conditions. 1) The posterior distribution has to be proper, that is, the integration of the posterior over its support should be finite. The proper prior ensures the proper posterior distribution. 2) The full conditional

probability distributions should be available. Our prior is proper because it consists of probability distributions whose integrals over their supports are one. The main challenge for using Gibbs sampler is that full conditional distributions are not always available. To avoid such issue, conjugate priors are widely used so that the full conditional distributions can be analytically derived. Then, sampling from the posterior by the Gibbs sampler becomes feasible.

Despite the attractive properties of the horseshoe+ prior distribution, it is difficult to implement Gibbs sampler because the half Cauchy distribution is not a conjugate prior. However, a half Cauchy random variable can be sampled with two hierarchical inverse gamma distributions by introducing an auxiliary variable (Wand et al., 2011). Makalic and Schmidt (Makalic and Schmidt, 2016) derived Gibbs sampler for the conventional horseshoe+ prior that deals with mean parameters in a simple regression with hierarchical inverse gamma priors. However, they omitted the global shrinkage parameter, which is a very important variable in our setting with sparsity issue. Based on the model in Section 2.2.2, we propose the complete Gibbs sampler for the informed horseshoe+ prior distribution.

The half Cauchy random variable with the scale parameter ς , $\sigma \sim \mathcal{C}^{+}(\varsigma)$, is equivalent to the hierarchical representation of two inverse gamma distributions with an auxiliary variable (Wand et al., 2011). Since they did not present the proof, we present the proof in the appendix 2.A.

$\sigma \sim \mathcal{C}^{+}(\varsigma)$ is equivalent to

$$\begin{aligned} \sigma^2|\nu &\sim \mathcal{IG}(0.5, 1/\nu) \\ \nu|\varsigma^2 &\sim \mathcal{IG}\left(0.5, 1/\varsigma^2\right) \end{aligned} \tag{2.16}$$

Then, (2.14) can be represented just with multiple inverse gamma distributions. The random variables that follow the half Cauchy distributions are decomposed into the two hierarchical inverse gamma distributions. Similarly, $\sigma_k \sim \mathcal{C}^{+}(\tau\,\eta_k)$ is equivalent to

$$\begin{aligned} \sigma_k^2|\nu_k &\sim \mathcal{IG}(0.5, 1/\nu_k) \\ \nu_k|\tau^2, \eta_k^2 &\sim \mathcal{IG}\left(0.5, 1/\left(\tau^2\,\eta_k^2\right)\right) \end{aligned} \tag{2.17}$$

where ν_k is the auxiliary variable.

Similarly, $\eta_k \sim \mathcal{C}^{+}(w)$ is equivalent to

$$\begin{aligned} \eta_k^2|\zeta_k &\sim \mathcal{IG}(0.5, 1/\zeta_k) \\ \zeta_k|w &\sim \mathcal{IG}\left(0.5, 1/w^2\right) \end{aligned} \tag{2.18}$$

In other words, η_k can be sampled with two inverse gamma distributions by introducing an auxiliary variable ζ_k. For the conventional horseshoe+, $w = 1$ is used for all the local shrinkage parameters η_k. For the proposed informed-horseshoe+, $w = 1$ is used for $k \notin \Omega$ and $w > 1$ is used for $k \in \Omega$ where Ω is a set of index for the prioritized sources.

$\tau \sim \mathcal{C}^{+}(\sigma_{\varepsilon})$ is equivalent to

$$\begin{aligned} \tau^2|\xi &\sim \mathcal{IG}(0.5, 1/\xi) \\ \xi|\sigma_{\varepsilon}^2 &\sim \mathcal{IG}\left(0.5, 1/\sigma_{\varepsilon}^2\right) \end{aligned} \tag{2.19}$$

where ξ is the auxiliary variable and

$\sigma_{\varepsilon} \sim \mathcal{C}^{+}(1)$ is equivalent to

$$\begin{aligned} \sigma_{\varepsilon}^2 &\sim \mathcal{IG}(0.5, 1/\phi) \\ \phi &\sim \mathcal{IG}(0.5, 1) \end{aligned} \tag{2.20}$$

with an auxiliary variable ϕ.

Then, the posterior distribution of each variance component variable is sampled via Gibbs sampling algorithm. The Gibbs sampler is presented in Algorithm 1. The details about how probability distributions in Algorithm 1 were specified are shown in Appendix 2.B. From the Gibbs sampler provided above, we can estimate variance components of the random effects coefficients.

Algorithm 1 Gibbs sampling algorithm for the informed horseshoe+ prior model.

Input: The number of iterations $nIter$, the design matrix $\mathbf{A}$, the measurements data $\mathbf{y}$, the number of products M, the number of parameters P, the number of measurements N

Initialize $iter = 1$, $x_{ki} = 0, \sigma_k^2, v_k, \eta_k^2, \tau^2, \zeta_k, \xi, \phi, \sigma_\varepsilon^2 = 1$ $(k = 1, \ldots, P,\ i = 1, \ldots, M)$
while $iter \leq nIter$ **do**
  Initialize $k = 1$
  while $k \leq P$ **do**
    Initialize $i = 1$
    while $i \leq M$ **do**
      $x_{ki}|\mathbf{y}_{ji}, \sigma_\varepsilon^2, \left\{x_{1i}, \ldots, x_{(k-1)i}, x_{(k+1)i}, \ldots, x_{Pi}\right\}, \sigma_k^2 \sim \mathcal{N}\left(\frac{\sigma_k^2 \sum_{j=1}^{N} \left\{a_{jk}\left(y_{ji} - \sum_{s \neq k} a_{js}\, x_{si}\right)\right\}}{\sigma_k^2 \sum_{j=1}^{N} {a_{jk}}^2 + \sigma_\varepsilon^2}, \frac{\sigma_k^2 \sigma_\varepsilon^2}{\sigma_k^2 \sum_{j=1}^{N} {a_{jk}}^2 + \sigma_\varepsilon^2}\right)$
      $i = i + 1$
    end while
    $\sigma_k^2 | \left\{x_{ki}\right\}_{i=1}^{M}, \nu_k \sim \mathcal{IG}\left(\frac{M+1}{2}, \frac{\sum_{i=1}^{M} (x_{ki})^2}{2} + \frac{1}{\nu_k}\right)$
    $\nu_k | \sigma_k^2, \tau^2, \eta_k^2 \sim \mathcal{IG}\left(1, \frac{1}{\sigma_k^2} + \frac{1}{\tau^2 \eta_k^2}\right)$
    $\eta_k^2 | \tau^2, \nu_k, \zeta_k \sim \mathcal{IG}\left(1, \frac{1}{\tau^2 \nu_k} + \frac{1}{\zeta_k}\right)$
    $\zeta_k | \eta_k^2, w^2 \sim \mathcal{IG}\left(1, \frac{1}{\eta_k^2} + \frac{1}{w^2}\right)$
    $k = k + 1$
  end while
  $\tau^2 | \left\{\nu_k\right\}_{k=1}^{P}, \left\{\eta_k\right\}_{k=1}^{P}, \xi \sim \mathcal{IG}\left(\frac{P+1}{2}, \sum_{k=1}^{P}\left(\frac{1}{\eta_k^2 \nu_k}\right) + \frac{1}{\xi}\right)$
  $\xi | \tau^2, \sigma_\varepsilon^2 \sim \mathcal{IG}\left(1, \frac{1}{\tau^2} + \frac{1}{\sigma_\varepsilon^2}\right)$
  $\sigma_\varepsilon^2 | \mathbf{y}, \left\{\mathbf{x}_i\right\}_{i=1}^{M}, \xi, \phi \sim \mathcal{IG}\left(\frac{MN}{2} + 1, \sum_{j=1}^{N} \sum_{i=1}^{M} \frac{\left(y_{ji} - \sum_{k=1}^{P} a_{jk}\, x_{ki}\right)^2}{2} + \frac{1}{\xi} + \frac{1}{\phi}\right)$
  $\phi | \sigma_\varepsilon \sim \mathcal{IG}\left(1, \frac{1}{\sigma_\varepsilon^2} + 1\right)$
  $iter = iter + 1$
end while

The overall implementation procedure is summarized in Fig. 2.5. The implementation steps can be classified into three categories: process-defined steps,

user-defined steps, and variation source identification steps. The process-defined steps, represented as white boxes with solid line border, require the design information from the physical process. In those steps, the user needs to identify the KPCs and KCCs. Based on the relationship between the KPCs and KCCs in the physical process, the design matrix $\mathbf{A}$ can be constructed as shown in Jin and Shi (1999b); Ding et al. (2002); Camelio et al. (2003). The tolerance requirements on the KCCs, if any, should be collected in this step.

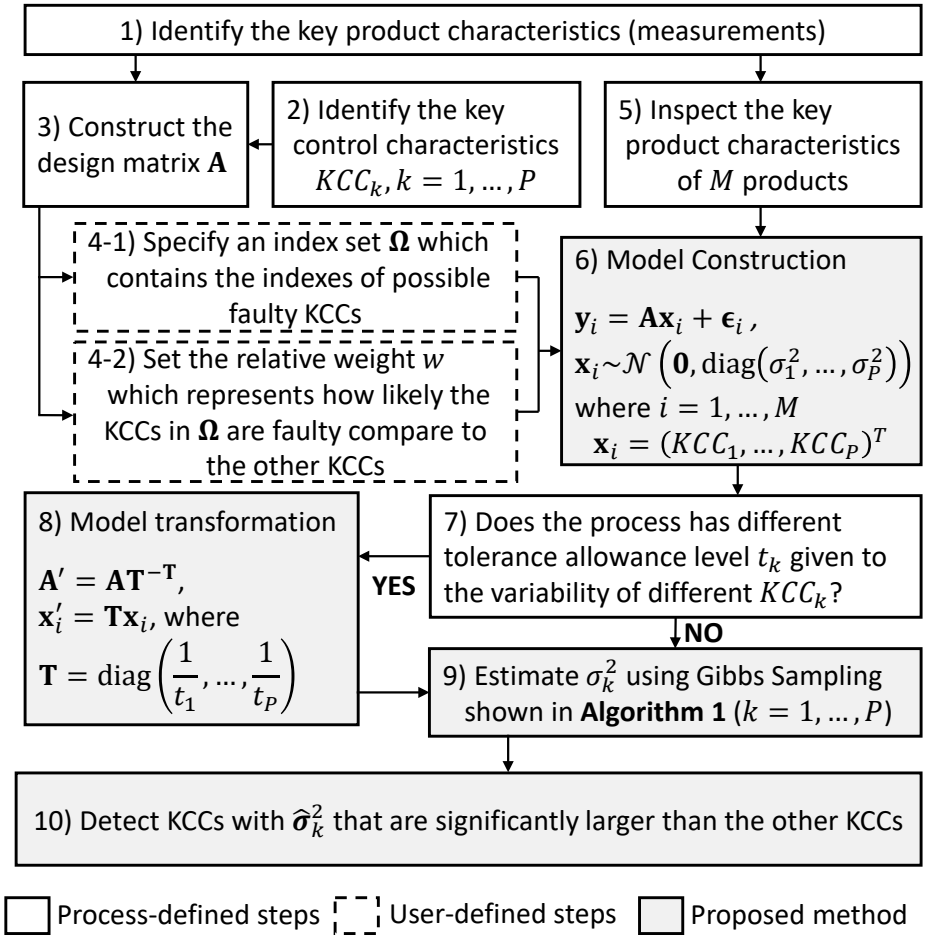


Figure 2.5: Summary of implementation procedure

The user-defined steps, represented as white boxes with dashed line border, provide an opportunity for practitioners to provide additional prior knowledge. Here, the user identifies the likely faulty variance sources that we should pay extra attention in the later variation source identification step. If known, the weight on those variation sources, w can be provided as well. If unknown, simply a uniform weight can be used.

The proposed variation source identification steps are shown in gray boxes with solid line border. In these steps, we first construct the model based on the input information from previous steps. If different tolerance levels are specified for the KCCs, we need to standardize the model to make the scales of the variation sources identical. After the model construction step, the variability of each KCC is estimated by the proposed Gibbs sampling in Algorithm 1. Based on the estimated variability, the faulty KCCs can be identified.

In the following section, we illustrate the numerical study to validate our proposed method.

2.4 Numerical Study

For performance evaluation and comparison, we conducted a series of numerical simulations. The performance evaluation focuses on two aspects. First, the model proposed in this chapter should be able to identify the variation sources correctly. Therefore, the detection power for identifying the variation sources under sparse condition is measured by the area under the curve (AUC). Second, once the varia-

tion sources have been identified, our model should provide accurate estimates of those identified variance components. This can be done by comparing the estimated values with the underlying true values that have been used for simulating data. In the numerical study, we compare the horseshoe+ (HS+), informed horseshoe+ (IHS+), and inverse gamma (IG) priors where the IG serves as a baseline.

2.4.1 Simulation Design

The simulation was conducted in three steps: data generation, parameter estimation, and performance evaluation. The data generation begins with randomly sampling an index set of the true faulty variation sources $\mathfrak{F}$ with a pre-specified size of $|\mathfrak{F}|$. The rest (non-faulty process natural variation sources) is set into $\mathfrak{N}$. Once the two groups of sources ($\mathfrak{F}$ and $\mathfrak{N}$) have been determined, we generate M random effects coefficients as $x_{ki} \sim \mathcal{N}(0, \sigma_k^2)$, $i = 1, \ldots, M$ for each location of the KCC $k = 1, \ldots, P$. In the numerical study, we assume that $\sigma_k = 1$ for $k \in \mathfrak{F}$ and $\sigma_k = 0.02$ for $k \in \mathfrak{N}$. It should be noted that the non-faulty sources in $\mathfrak{N}$ have non-zero σ_k in order to incorporate natural process variations under normal working conditions which is common in manufacturing practice. To generate the observations $\mathbf{y} = (\mathbf{y}_1^T, \ldots, \mathbf{y}_M^T)^T$, we need to specify a design matrix $\mathbf{A}$ and measurement noise $\boldsymbol{\epsilon}$. The (j, k) th element a_{jk} of the design matrix $\mathbf{A}$ is generated from a normal distribution $\mathcal{N}(0, 100)$ and $\varepsilon_{ji} \sim \mathcal{N}(0, 0.2)$ for $j = 1, \ldots, N, i = 1, \ldots M$. The standard deviation of measurement noise σ_ε is significantly greater than σ_k for $k \in \mathfrak{N}$. Also, σ_ε is 44.22% of the standard deviation of the faulty variation sources σ_k for $k \in \mathfrak{F}$ ($\sigma_k^2/\sigma_\varepsilon^2 = 0.2 \; \forall k \in \mathfrak{F}$). In other words, measurement noise is significantly large;

hence, detecting the faulty sources should not be trivial for the readily-available methods such as a IG-based model. The last step of the data generation process is to obtain a set of simulated observations $\mathbf{y} = (\mathbf{y}_1^T, \ldots, \mathbf{y}_M^T)^T$ based on $\mathbf{y}_i = A\mathbf{x}_i + \boldsymbol{\epsilon}_i$ for every product i. The data generation process is summarized in Fig. 2.6.

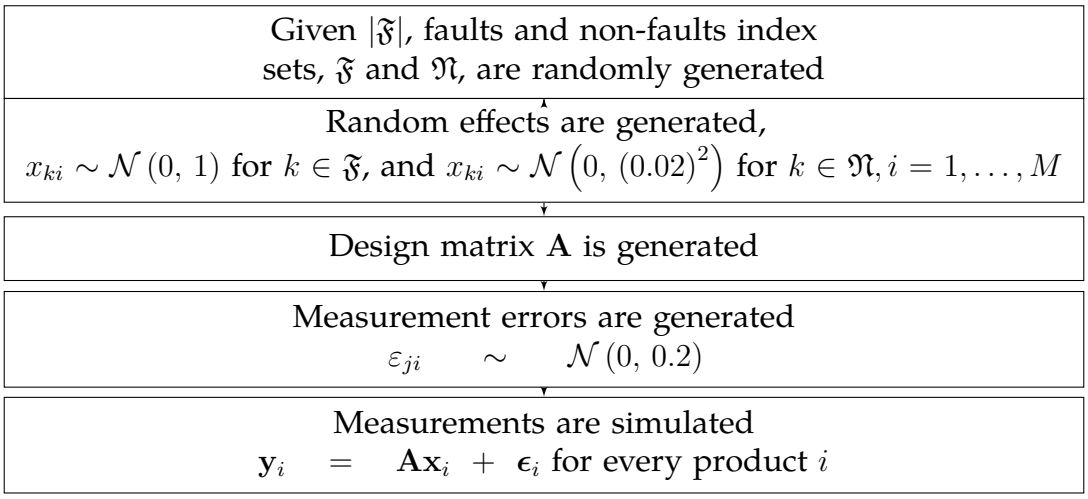


Figure 2.6: Flow chart of the data generation process

To include various scenarios in the performance evaluation, we considered the number of potential fault locations to be P = 10, 30, or 50. Also, different number of faulty sources $|\mathfrak{F}|$ were considered, $|\mathfrak{F}| = 1$ and $|\mathfrak{F}| = 2$. For instance, if P = 10 and $|\mathfrak{F}| = 1$, then there should be only one faulty variation source with $\sigma_k = 1$ and nine non-faulty sources with $\sigma_k = 0.02$. The complexity of the identification problem increases as both P and $|\mathfrak{F}|$ increase. For each scenario, we conducted 100 numerical simulations. We assume that we have three products (M = 3) and the number of observations for each product is set as 3 (N = 3). From the diagnosability theory (Zhou et al., 2003), we can show that the sample size specified in this simulation makes our problem undiagnosable. Notice that these scenarios are

designed challenging. For example, the scenario with 50 variance components of the process errors has 50+1=51 variance components to be estimated, including the measurement noise. On the other hand, only 9 measurements ($N \times M = 9$) are available. Estimating 51 parameters based on 9 data points is not a trivial task. In other words, conventional Bayesian estimation methods such as the one based on the IG should not perform well unless a sophisticated sparsity-enhancing technique has been applied.

In this comparative performance evaluation, we considered three prior distributions: the IG, HS+, and IHS+ priors. For the IG, the hyper-parameter is specified as 0.001 and a non-informative prior $\log \sigma_\varepsilon \propto 1$ is used for measurement noise. The IG specifications are adopted from the literature (Zeng and Chen, 2015; Gelman, 2006). Unlike the HS+ which does not take any prior input, the proposed IHS+ is capable of using domain knowledge of fault-likely KCCs of the manufacturing process. In other words, we may select a few indexes of variance components in advance and provide the index set (denoted by Ω) to the IHS+. The variance components corresponding to the indexes in Ω should empirically have shown high probability of being variation sources. In this case, it is easy to foresee that the choice of Ω would have a significant impact on the variation source identification performance. To investigate this issue, we considered two aspects regarding the "quality" of the knowledge set Ω. First is the size of fault-likely set $|\Omega|$. Suppose we have 50 variation sources and, among them, there are 5 prioritized variance components ($|\Omega|/P = 0.1$) that have frequently been the major variation sources. Intuitively, this domain knowledge is better than a knowledge set with 15 prioritized sources

($|\Omega|/P = 0.3$). In addition to size, another important aspect is the accuracy of domain knowledge defined as

$$acc\left(\Omega, \mathfrak{F}\right) = \frac{|\Omega \cap \mathfrak{F}|}{\min\left(|\Omega|, |\mathfrak{F}|\right)} \tag{2.21}$$

The accuracy is defined as the number of correctly identified variations among the pre-specified set ($|\Omega \cap \mathfrak{F}|$) over the maximum number of identifiable variations with the prioritized set ($\min\left(|\Omega|, |\mathfrak{F}|\right)$). In practice, given 5 specified locations ($|\Omega| = 5$), it is possible that among 2 variational faults ($|\mathfrak{F}| = 2$), only one of them is correctly specified ($|\Omega \cap \mathfrak{F}| = 1$). In that case, the proportion of the correctly specified prioritized sources is 50%. Intuitively, the higher $acc\left(\Omega, \mathfrak{F}\right)$ is the better; hence, $acc\left(\Omega, \mathfrak{F}\right)$ can be viewed as a measure of the accuracy of Ω. Both the size and accuracy of the domain knowledge have impact on the performance of our proposed model. Therefore, in this simulation, we used two different sizes of Ω, $|\Omega|/P = 0.1$ and $|\Omega|/P = 0.3$. Also, we considered five different cases where the average of $acc\left(\Omega, \mathfrak{F}\right)$ over experiments are 0%, 30%, 50%, 70%, and 100%. When it is 0%, no variation source in Ω is correctly specified whereas 100% means that $|\Omega \cap \mathfrak{F}| = \min\left(|\Omega|, |\mathfrak{F}|\right)$.

It should be noted that, for the IHS+ prior, there is a hyper-parameter w that needs to be specified. The hyper-parameter w determines the scale of local shrinkage parameter and there is no golden standard for computing the best w for the half Cauchy distribution. However, it is known that the half Cauchy distribution with scale parameter w is w times widely distributed than the one with scale parameter 1, i.e., $P\left(a < 1\right) = P\left(b < w\right)$ when $a \sim \mathcal{C}\left(1\right)^{+}$ and $b \sim \mathcal{C}\left(w\right)^{+}$. Therefore, w can be

selected based on empirical knowledge of the relative likelihood of faults on the variation sources. For instance, suppose an engineer knows that a specific KCC is likely to be a variation source with 50% chance while other KCCs are likely to be variation sources with only 10% chance based on his/her experience. Then, $w = 5$ would be a reasonable choice. Throughout the numerical study, we specified $w = 10$ based on a series of trial-and-error using the case study data obtained from a multistage panel assembly process. We would like to emphasize that w needs to be tailored for a different application because inappropriately tuned w may have negative impact on the performance of the IHS+ prior.

Based on the simulated data, we estimate variance components using Gibbs sampler for each scenario. The estimator is set as the mean of the samples from 10,000 iterations in Gibbs sampler algorithm after burning in 20,000 iterations.

2.4.2 Performance Evaluation

2.4.2.1 Fault Detection Capability

The proposed model classifies all variation sources into two categories $\hat{\mathfrak{F}}$ and $\hat{\mathfrak{N}}$, where $\hat{\mathfrak{F}}$ and $\hat{\mathfrak{N}}$ are the sets of indexes classified to variational faults and non-faults, respectively. Once we obtain the estimates of all parameters in the model, we identify the variance components with larger estimates. Then, we can compare the model-identified variation sources with the true variation sources that have been used for generating the data to assess the detection power of the model. By repeating this procedure, we can compute the rates of true positive (TP), true negative (TN), false positive (FP), and false negative (FN). TP means that the variation source

detected by the algorithm was indeed one of the true variation sources. Similarly, TN indicates that the non-fault source determined by the algorithm was truly natural process variation. FP and FN respectively refer to false alarm and misdetection. The variation source identification (fault detection) process is illustrated in Fig. 2.7.

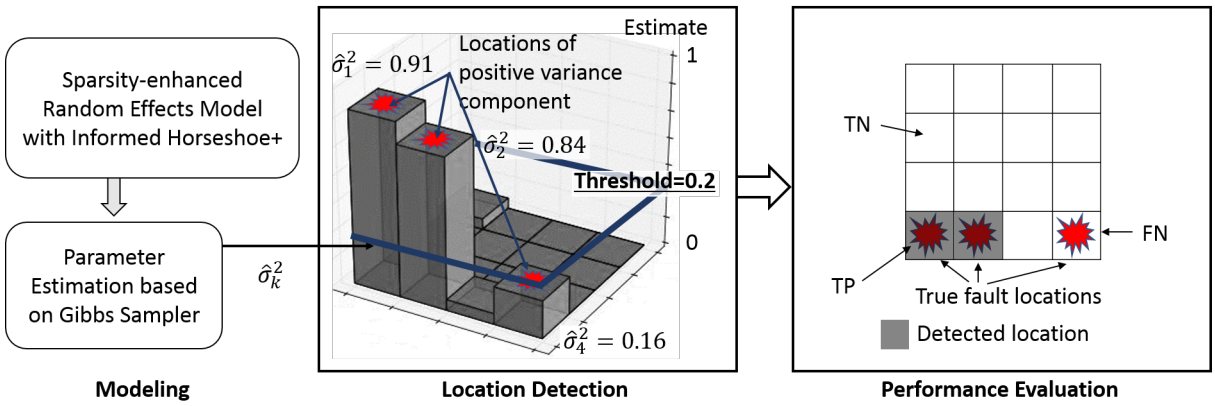


Figure 2.7: Illustration of the variation source identification process

The power of detecting variation sources can be conveniently summarized by the AUC which is the area under the Receiver Operating Characteristics (ROC) curve (Hanley and McNeil, 1982). The AUC is widely used for assessing classification accuracy because it considers the trade-off between type I and type II errors. If the algorithm provides a perfect classification result, the AUC will be 1. The AUC with a value close to 0.5 indicates that the classification algorithm did not perform significantly better than a random guess. The AUCs computed by our numerical simulation are shown in Table 2.1.

Table 2.1: Average AUCs from numerical study

				$\|\Omega\|/P = 10\%$					$\|\Omega\|/P = 30\%$				
Pr{Correct}				**0%**	**30%**	**50%**	**70%**	**100%**	**0%**	**30%**	**50%**	**70%**	**100%**
$\|\mathfrak{F}\|$	**P**	**IG**	**HS+**	**IHS+**					**IHS+**				
1	**10**	0.9354	0.9885	0.9941	0.9931	0.9974	0.9930	0.9996	0.9789	0.9777	0.9943	0.9944	0.9989
	30	0.5278	0.9337	0.8910	0.9542	0.9459	0.9814	0.9964	0.7704	0.8587	0.8721	0.9246	0.9859
	50	0.4986	0.8665	0.8049	0.8450	0.9104	0.9133	0.9930	0.6306	0.7682	0.8707	0.8663	0.9779
2	**10**	0.7797	0.8843	0.8671	0.8969	0.9193	0.9209	0.9454	0.7773	0.8745	0.9099	0.9573	0.9837
	30	0.5185	0.7448	0.6881	0.8100	0.8061	0.9070	0.9613	0.5634	0.6508	0.7425	0.8253	0.9285
	50	0.5012	0.6884	0.6352	0.7201	0.8301	0.8527	0.9424	0.5349	0.6153	0.6953	0.7578	0.8700

From Table 2.1, we can see that the IG prior always performs worse than both the HS+ and IHS+ for all cases. The IG shows acceptable performance for less-sparse cases when P = 10. However, the performance degrades quickly as the size of the problem increases. Interestingly, even the IHS+ with 0% accurate Ω outperforms the IG. This observation highlights the effectiveness of using sparsity-enhanced priors when the variation source identification problem is undiagnosable. As expected, for all three prior distributions, the AUC decreases as P and $|\mathfrak{F}|$ increases. However, unlike the IG prior, the sparsity-enhanced priors (HS+ and IHS+) show reasonable performance even with complex cases. Especially, the proposed method (IHS+) was able to identify the variation sources satisfactorily under the most challenging case considered in the simulation (P = 50 and $|\mathfrak{F}| = 2$). Comparing the HS+ and IHS+, in general, we can see that the IHS+ outperforms the HS+ if the quality of domain knowledge embedded in Ω was not too bad. In other words, unlike the conventional prior distributions, the fault detection performance of the proposed IHS+ prior should always have a room for improvement as long as we could obtain a better knowledge set Ω. The sparsity prior enables ones to

identify a larger number of faulty variation sources than observed data points by forcing the variance estimates of the non-faulty sources to be close to zero so that only a few parameters that correspond to the significant faults are estimated as positive. Unlike our methods taking advantage of the sparsity condition, the inverse gamma prior distribution assumes positive values for all parameters (Polson and Scott, 2011). Therefore, the inverse gamma prior model should estimate almost 51 positive variances with only 9 measures. Although some variance components might be estimated close to zero, the tendency of pushing the estimates away from zero makes the model fail in estimation. As a results, the sparsity prior of the Bayesian model becomes more crucial for the estimation based on data with higher-dimension and lower-sample size.

2.4.2.2 Estimation Accuracy

Correctly identifying the variation sources is undoubtedly the most important goal of the proposed sparsity-enhanced Bayesian method. However, it would be meaningful to investigate not only the variation source identification performance but also the estimation accuracy because, once the variation sources were identified, one may want to take a closer look at how large the variances are. To obtain useful insights along this line, we report the average of estimated variance components that were correctly classified into $\mathfrak{F}$ and $\mathfrak{N}$.

The classification depends on the decision threshold; hence, as the first step, an optimal threshold is defined based on the ROC curve. The optimal threshold is the one that corresponds to a point on a ROC curve which is closest to (FPR,

TPR)$= (0, 1)$ in terms of the Euclidean distance where FPR and TPR respectively denote a false positive rate and a true positive rate (Perkins and Schisterman, 2006). Fig. 2.8 shows the (FPR, TPR) pair computed by the optimal threshold for the IG, HS+, and IHS+ priors. In Fig. 2.8, the IHS+(100%) represents the case of $|\Omega \cap \mathfrak{F}| = \min(|\Omega|, |\mathfrak{F}|)$, i.e., perfectly accurate Ω whereas the IHS+(0%) indicates the opposite worst case.

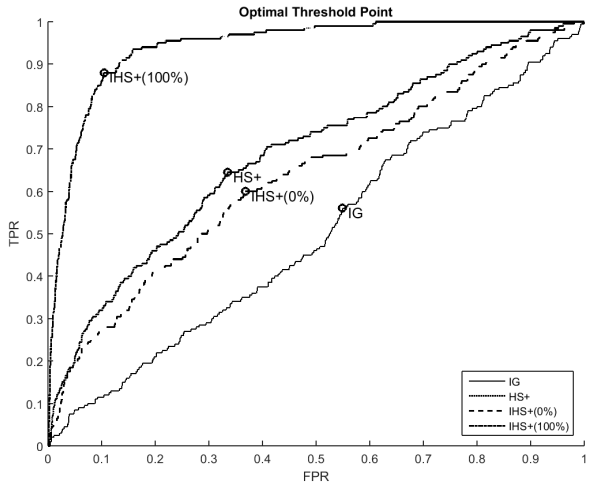


Figure 2.8: An optimal threshold point for the IG, HS+, 0% IHS+, and 100% IHS+ in a setting with $P = 50$, $|\mathfrak{F}| = 2$, and $|\Omega| = 5$.

Based on the optimal threshold, we classify the indexes of the variance components into $\hat{\mathfrak{F}}$ and $\hat{\mathfrak{N}}$. Then, we present the estimates of the correctly classified variance components in Table 2.2 (i.e. $\hat{\sigma}_k^2$, where $k \in (\mathfrak{F} \cap \hat{\mathfrak{F}}) \cup (\mathfrak{N} \cap \hat{\mathfrak{N}})$) to evaluate the estimation accuracy. In Table 2.2, the column of $\mathfrak{F}$ presents the faulty variation sources while the column of $\mathfrak{N}$ contains the non-faulty sources. In other words, the true value for $\mathfrak{F}$ column should be 1 whereas the true value for $\mathfrak{N}$ is $(0.02)^2$.

Table 2.2: Average estimated variance components that are correctly classified in numerical study

Pr{Correct}							**0%**		**30%**		**50%**		**70%**		**100%**	
Priors			**IG**		**HS+**		**IHS+**									
$\lvert\mathfrak{F}\rvert$	**P**	$\lvert\Omega\rvert/P$	$\mathfrak{F}$	$\mathfrak{N}$	$\mathfrak{F}$	$\mathfrak{N}$	$\mathfrak{F}$	$\mathfrak{N}$	$\mathfrak{F}$	$\mathfrak{N}$	$\mathfrak{F}$	$\mathfrak{N}$	$\mathfrak{F}$	$\mathfrak{N}$	$\mathfrak{F}$	$\mathfrak{N}$
1	**10**	**10%**	2.841	0.045	1.792	0.007	1.744	0.007	1.735	0.005	1.819	0.005	1.864	0.005	1.919	0.004
		30%					1.650	0.005	1.754	0.005	1.756	0.005	1.795	0.005	1.896	0.005
	30	**10%**	15.8	0.242	1.008	0.003	1.007	0.003	1.178	0.003	1.226	0.003	1.329	0.003	1.399	0.003
		30%					1.006	0.002	1.419	0.004	1.314	0.002	1.362	0.002	1.445	0.004
	50	**10%**	1198.0	4.395	1.035	0.002	0.893	0.001	1.010	0.002	1.091	0.001	1.257	0.002	1.292	0.002
		30%					0.621	0.001	1.055	0.001	0.948	0.002	1.047	0.001	1.103	0.002
2	**10**	**10%**	2.704	0.078	1.488	0.039	1.409	0.034	1.593	0.036	1.672	0.030	1.760	0.035	1.750	0.023
		30%					1.421	0.038	1.514	0.029	1.806	0.021	1.784	0.020	1.884	0.019
	30	**10%**	15.1	0.223	1.304	0.010	0.970	0.010	1.049	0.007	1.090	0.008	1.204	0.010	1.353	0.009
		30%					0.577	0.004	1.023	0.009	1.129	0.007	1.117	0.009	1.189	0.008
	50	**10%**	311.7	2.364	0.545	0.005	0.399	0.004	0.646	0.004	0.773	0.006	0.788	0.005	0.870	0.007
		30%					0.510	0.003	0.594	0.003	0.506	0.002	0.740	0.005	0.764	0.006

From Table 2.2, we can see that the HS+ and IHS+ priors significantly outperform the IG prior in all cases. More importantly, the IG prior tends to overestimate the variance components and the accuracy of the IG-based estimation degrades quickly as P increases. For $P = 50$, the average estimate based on the IG prior becomes extremely large. This result, in addition to what we observed in AUC comparison, suggests that the IG prior is not suitable for solving a sparse variance identification problem. Unlike the IG prior, the estimates from the sparsity-enhanced priors (HS+ and IHS+) show acceptable results. An interesting observation is that the average estimates of σ_k^2 tend to decrease as P becomes larger for both the HS+ and IHS+. This decreasing trend is the opposite of what we observe for the IG prior. The phenomenon is mainly due to the global shrinkage parameter. As P increases, data become more sparse and sparse data inevitably makes the

global shrinkage parameters quickly shrink down by a large amount. The powerful shrinkage effect naturally pushes the estimates towards 0. As mentioned earlier, the quality of prior knowledge incorporated into the IHS+ affects the estimation accuracy. The estimation accuracy of the IHS+ increases as Ω with better quality becomes available. Especially, as $|\Omega|$ decreases, the estimation accuracy of the IHS+ generally increases.

2.5 Case Study

In this section, we demonstrate the variation source identification method based on the proposed model with application in multistage panel assembly process. The assembly process is adopted from Zhou et al. (2003) and the process has been already illustrated in Fig. 2.1 in the Introduction of this chapter. As described earlier, this multistage process consists of three stations: two assembly stations and an inspection station. At each assembly station, the panels are secured by the activated fixture locators denoted by $L_i,\ i = 1, \ldots, 8$. In the case study, we have two types of locators: a 4-way locator which controls the motion of a panel in both x- and z-coordinates and a 2-way locator which controls the movement only in z-coordinate. The positional dimension of each assembly is observed in x- and z-dimensions by the sensors installed at the corner of each station. Each sensor, $M_i,\ i = 1, \ldots, 5$, measures two-dimensional deviations of panel. Throughout the case study, the assembly process described above is referred as Process I. There exists another type of assembly process as discussed in Zhou et al. (2003). This

process, namely Process II, has different sensor location in the third station and the sensor placement makes differences in terms of diagnosability (Zhou et al., 2003). Therefore, we shall consider both processes (Process I and II) in the case study.

The original assembly process described in Zhou et al. (2003) is not a complex process. There are only 18 possible directions for assembly deviation ($P = 18$). Therefore, the two-dimensional deviation data (measurements of x- and z-coordinates for each product) collected by a few sensors were sufficient for providing enough measurements to make the variation source identification problem diagnosable. The case study needs to be based on a more challenging dataset, i.e., sparse and high-dimensional problem with limited samples, to emphasize the advantageous features of the sparsity-enhanced priors. Therefore, in the design matrix, we removed some rows and columns that are associated with the z-coordinate sensor. In other words, we assumed that the sensors only measure x-coordinate. In this way, the design matrices for Process I and II have dimensions of (5×11) and (5×12), respectively. From the revised assembly process, we have $P = 11$ for Process I and $P = 12$ for Process II. The details of the design matrices are presented in Appendix 2.C.

For the assembly process, we consider two cases: $|\mathfrak{F}| = 1$ and $|\mathfrak{F}| = 2$. For each case, we conducted a series of multiple variation sources identification to include all variation sources. For instance, among 11 variance components for Process I, the actual fault can occur anywhere. Thus, we first use data when the true variational fault has occurred at location 1 and see if the proposed method can correctly detect the true location of the variation source. After that, we use different

data with the true variation source location at 2 and repeat the same variation source identification procedure. Basically, we perform the variation source identification for all 11 locations sequentially. In other words, the variation source identification has been conducted P times for $|\mathfrak{F}| = 1$ and $\binom{P}{2}$ times for $|\mathfrak{F}| = 2$. Furthermore, each detection for a given variation source has been repeated 10 times. This case study procedure makes the total number of variation source identification trials $10 \times P$ for $|\mathfrak{F}| = 1$ and $10\times \binom{P}{2}$ times for $|\mathfrak{F}| = 2$.

Another key player in the case study is the quality of Ω provided to the IHS+. We considered $|\Omega| = 1$ and $|\Omega| = 3$. Also, we examined performance of the IHS+ prior with probability of correctness of 0%, 30%, 50%, 70%, and 100% for Ω as similar to the numerical simulation. For the sake of performance evaluation, we wanted to consider various Ω. Therefore, the prior information Ω has been randomly generated according to the size and the accuracy specified above. Based on the case study data, all parameters of the IG, HS+, and IHS+ were estimated by the Gibbs sampler. The estimator is defined as the mean of the samples from 10,000 iterations after burning in the first 20,000 iterations. Unlike the numerical study, estimation accuracy may not be easy to evaluate because the true values of σ_k^2 and σ_ε^2 are typically not available in practice. Therefore, we mainly use AUC as our performance metric for evaluating the detection power of variation source identification.

There are two key factors that may have impact on the performance of our proposed method. The difference magnitude between the process noises of non-faulty and faulty variation sources is one of them. Another challenge for the fault

identification under a sparsity condition would be the measurement noise. To make the performance evaluation section more complete, three scenarios are considered: (i) small process noise with large measurement noise, (ii) large process noise with small measurement noise, and (iii) large process noise with large measurement noise. For the analysis (i), we follow the same standard deviations of faulty, non-faulty, measurement noise to the numerical study, i.e. $\sigma_{\mathfrak{N}}/\sigma_{\mathfrak{F}} = 0.02$ and $\sigma_{\varepsilon}/\sigma_{\mathfrak{F}} = 0.447$ where $\sigma_{\mathfrak{F}}$, $\sigma_{\mathfrak{N}}$, σ_{ε} are the standard deviations of faulty, non-faulty, and measurement noise, respectively. In this setting, significantly large measurement noise makes the variation source identification challenging. In (ii) we adopted standard deviation ratios that already have been used and validated in the variation source identification literature (Ding et al., 2002), that is, $\sigma_{\mathfrak{N}}/\sigma_{\mathfrak{F}} = 1/3$ and $\sigma_{\varepsilon}/\sigma_{\mathfrak{F}} = 0.01$. Compared to (i), it is easy to see that the difference between $\sigma_{\mathfrak{N}}$ and $\sigma_{\mathfrak{F}}$ is substantially reduced (i.e., natural process noise for non-faulty sources is increased) whereas the measurement noise has become less significant. In (iii), both the large standard deviation of measurement noise from (i) and the large process noise from (ii) are adopted (i.e.$\sigma_{\mathfrak{N}}/\sigma_{\mathfrak{F}} = 1/3$ and $\sigma_{\varepsilon}/\sigma_{\mathfrak{F}} = 0.447$). The average AUCs from the case study (i), (ii), and (iii) are summarized in Table 2.3, 2.4, and 2.5.

Table 2.3: Average AUCs from case study under small process noises and large measurement noises ($\sigma_{\mathfrak{N}}/\sigma_{\mathfrak{F}} = 0.02$ and $\sigma_{\varepsilon}/\sigma_{\mathfrak{F}} = 0.4472$)

				$\|\Omega\| = 1$					$\|\Omega\| = 3$				
Pr{Correct}				**0%**	**30%**	**50%**	**70%**	**100%**	**0%**	**30%**	**50%**	**70%**	**100%**
$\|\mathfrak{F}\|$	**Process**	**IG**	**HS+**	**IHS+**					**IHS+**				
1	**I**	0.8023	0.8768	0.8516	0.8964	0.9034	0.9409	0.9679	0.7839	0.8209	0.8668	0.8903	0.9382
	II	0.7646	0.8433	0.8253	0.8510	0.8910	0.8955	0.9349	0.7406	0.8028	0.8317	0.8559	0.9008
2	**I**	0.7262	0.8209	0.7906	0.8183	0.8427	0.8540	0.8785	0.7331	0.7916	0.8296	0.8777	0.9332
	II	0.7015	0.7808	0.7584	0.7836	0.7990	0.8186	0.8462	0.7138	0.7767	0.8195	0.8436	0.9070

Table 2.4: Average AUCs from case study under large process noises and small measurement noises ($\sigma_{\mathfrak{N}}/\sigma_{\mathfrak{F}} = 1/3$ and $\sigma_{\varepsilon}/\sigma_{\mathfrak{F}} = 0.01$)

				$\|\Omega\| = 1$					$\|\Omega\| = 3$				
Pr{Correct}				**0%**	**30%**	**50%**	**70%**	**100%**	**0%**	**30%**	**50%**	**70%**	**100%**
$\|\mathfrak{F}\|$	**Process**	**IG**	**HS+**	**IHS+**					**IHS+**				
1	**I**	0.6815	0.8692	0.8433	0.8737	0.9016	0.9316	0.9570	0.7898	0.8233	0.8549	0.8841	0.9364
	II	0.6679	0.8228	0.7920	0.8335	0.8627	0.8798	0.9256	0.7848	0.8323	0.8680	0.8833	0.9268
2	**I**	0.6748	0.8170	0.7882	0.8146	0.8395	0.8498	0.8808	0.7402	0.8059	0.8379	0.8684	0.9333
	II	0.6349	0.7942	0.7711	0.7956	0.8111	0.8316	0.8574	0.7162	0.7758	0.8255	0.8543	0.9073

Table 2.5: Average AUCs from case study under large process and measurement noises ($\sigma_{\mathfrak{N}}/\sigma_{\mathfrak{F}} = 1/3$ and $\sigma_{\varepsilon}/\sigma_{\mathfrak{F}} = 0.4472$)

				$\|\Omega\| = 1$					$\|\Omega\| = 3$				
Pr{Correct}				**0%**	**30%**	**50%**	**70%**	**100%**	**0%**	**30%**	**50%**	**70%**	**100%**
$\|\mathfrak{F}\|$	**Process**	**IG**	**HS+**	**IHS+**					**IHS+**				
1	**I**	0.6265	0.7436	0.7135	0.7670	0.8134	0.8571	0.9138	0.6983	0.7667	0.8018	0.8464	0.9157
	II	0.6561	0.7374	0.7176	0.7723	0.8001	0.8442	0.8947	0.6524	0.7136	0.7517	0.8126	0.8509
2	**I**	0.6382	0.7353	0.7037	0.7449	0.7656	0.7909	0.8230	0.6644	0.7446	0.7888	0.8391	0.9073
	II	0.6225	0.7016	0.6756	0.7076	0.7340	0.7542	0.7869	0.6355	0.7043	0.7491	0.8038	0.8674

Based on Table 2.3, 2.4, and 2.5, we can see that, unlike the numerical simulation, the performance of the IG in the case study is comparable to the other two sparsity-

enhanced priors. This result is due to the relatively low dimension of the multistage assembly process used in the case study. However, we still see that the HS+ and IHS+ outperform the IG in most cases. When $|\mathfrak{F}|$ increases from 1 to 2, the AUC of the HS+ reduces by approximately 5% in both Process I and II. Similar performance degradation can be observed for the IHS+ as well. The performance of the IHS+ prior depends on the quality of prior domain knowledge Ω. Better fault detection can be achieved using the IHS+ with $|\Omega| = 1$ than $|\Omega| = 3$. Also, the AUC of the IHS+ increases as proportion of the correctly identified variation sources in Ω increases. Comparing the HS+ and IHS+, the IHS+ outperforms the HS+ if the accuracy of Ω was greater than 30% when $|\Omega| = 1$. For $|\Omega| = 3$, the IHS+ demands higher quality Ω to outperform the HS+, e.g., accuracy of Ω may need to be greater than 70%.

Table 2.4 and Table 2.5 show that the proposed sparsity enhanced prior models are robust to the natural process noises than IG. In Table 2.4, the AUCs of IG are significantly smaller than those in Table 2.3. On the other hand, the AUCs of the HS+ and the IHS+ are almost similar to those in Table 2.3. For the most challenging scenario where both measurements noises and natural process noises are large (Table 2.5), the AUCs of all the models are considerably smaller than the other cases where either measurement noises or process noises are small. However, even in this challenging case, we observe that the proposed sparsity enhanced model outperforms IG consistently.

The results shown in the case study along with the numerical simulation results, we can clearly see the importance of considering a sparsity-enhanced prior for

detecting faulty variance components. In addition, the proposed IHS+ prior can provide significantly better fault detection performance as long as we had reasonably good domain knowledge of the multistage process. In practice, such empirical knowledge is not difficult to obtain. Therefore, the proposed IHS+ prior model has significant potentials for improving the variation source detection power especially for solving undiagnosable variance identification problems.

2.6 Conclusion

Modern multistage manufacturing processes typically involve a lot of different types of operations and a critical fault may occur anywhere in the complex multistage process. Because the impact of a variation easily propagates to downstream operations, it is crucial (at the same time, challenging) to efficiently identify the source of variation. In this chapter, we proposed a variation source identification method based on Bayesian random effects model. The proposed model introduces a unique sparsity-enhanced prior called the informed horseshoe+ prior and it can address the sparse nature of the variation source identification problem effectively. In addition, the proposed method can incorporate the prior domain knowledge into the model so that variation source identification can be done even for a complex and undiagnosable system. We showed the promising performance of our model through a series of numerical simulations and case study. We believe the proposed approach can be used for a general variation component estimation based on the linear mixed effects models. The proposed model can solve variation source

detection problems in diverse manufacturing processes such as machining process.

The specific contributions of the proposed variation source identification method are as follows: first, an innovative sparsity-enhanced prior distribution, the informed horsesoe+, is established and integrated into the Bayesian random effects model. In the literature, a few prior distributions for addressing the sparsity issue already have been investigated. However, those existing priors are not directly applicable to the variance component identification problem. Therefore, we proposed the informed horseshoe+ prior distribution which significantly extends the conventional horseshoe+ prior. Second, a Gibbs sampler specifically designed for the proposed model is developed. Bayesian inference often suffers from the heavy computational burden. Due to the multiple hierarchical structure of our Bayesian random effects model, an efficient inference method was crucial to make the variation source identification method feasible in practice. The proposed Gibbs sampler decomposes the half Cauchy distributions into two hierarchical inverse gamma distributions so that full conditional probability distributions can be easily computed. Third, we demonstrated the importance of utilizing domain knowledge about the specific process of our interest for solving a variation source identification problem. Unlike the existing Bayesian inference methods that are available in relevant literature, our approach provides a straightforward way of embedding empirical domain knowledge into Bayesian inference procedure. Furthermore, we investigated how the quality of empirical domain knowledge affect the performance of variation source identification. To the best of our knowledge, there is no similar study in current literature.

In this chapter, one of the assumptions is that all process errors are independent. However, in some manufacturing processes, the process errors might be correlated to each other. If the correlation is strong, it may affect the fault detection results. Estimating the true correlation structure is a very challenging task especially when the manufacturing system has a complex multistage processes and it may demand a new algorithm to achieve the goal. However, expanding our current method so that it can incorporate correlated process errors might be an interesting research topic. It should be also noted that the proposed model has been tested and validated mainly in the application of multistage panel assembly process. Although the extensive numerical simulations have been conducted for a comprehensive performance evaluation, a different application may require different parameter specifications. Therefore, we plan to perform fault detection based on more types of manufacturing system to highlight the generalizability of the proposed method. We shall investigate along this line and report the results in the near future.

APPENDIX

2.A Proof of the Half Cauchy Distribution Equivalent to two Inverse Gamma Distributions

As Wand et al. Wand et al. (2011) omitted the proof, we present the proof

$$\sigma \sim \mathcal{C}^{+}\left(\varsigma\right)$$

is equivalent to

$$\sigma^2|\nu \sim \mathcal{IG}\left(0.5, 1/\nu\right)$$
$$\nu|\varsigma^2 \sim \mathcal{IG}\left(0.5, 1/\varsigma^2\right)$$

Proof. The marginal probability distribution of $\sigma^2|\varsigma^2$ is

$$\begin{aligned} f\left(\sigma\right) &= \int P\left(\sigma^2|\nu\right) P\left(\nu|\varsigma^2\right) d\nu \, |\mathbf{J}| \\ &= \int_0^{\infty} \frac{\left(\frac{1}{\nu}\right)^{\frac{1}{2}}}{\Gamma\left(\frac{1}{2}\right)} \left(\sigma^2\right)^{-\frac{3}{2}} \exp\left(-\frac{1}{\nu\sigma^2}\right) \frac{\left(\frac{1}{\varsigma^2}\right)^{\frac{1}{2}}}{\Gamma\left(\frac{1}{2}\right)} \left(\nu\right)^{-\frac{3}{2}} \\ &\qquad\qquad \exp\left(-\frac{1}{\nu\varsigma^2}\right) d\nu \, |\mathbf{J}| \\ &= \frac{2}{\sigma^2\Gamma(\frac{1}{2})} \frac{\left(\frac{1}{\varsigma^2}\right)^{\frac{1}{2}}}{\Gamma(\frac{1}{2})} \int_0^{\infty} \left(\nu\right)^{-2} \exp\left\{-\frac{1}{\nu}\left(\frac{1}{\sigma^2} + \frac{1}{\varsigma^2}\right)\right\} d\nu \\ &= \frac{1}{\pi\varsigma\sigma^2} \frac{\Gamma\left(1\right)}{\left(\frac{1}{\sigma^2} + \frac{1}{\varsigma^2}\right)} \end{aligned}$$

$$= \frac{1}{\pi\varsigma\left(1+\left(\frac{\sigma}{\varsigma}\right)^2\right)}$$

where, $P\left(\sigma^2|\nu\right)$ and $P\left(\nu|\varsigma^2\right)$ are inverse gamma probability distribution of σ^2 and ν, respectively, and $\mathbf{J}$ is a Jacobian of the transformation from the variable σ^2 to σ. ■

2.B Gibbs Sampling

The joint distribution

$$P\left(\mathbf{y}\right) = \prod_{i=1}^{M} P\left(\mathbf{y}_i|\sigma_\varepsilon^2, \mathbf{x}_i\right) \prod_{k=1}^{P}\left\{\prod_{i=1}^{M}\left\{P\left(x_{ki}|\sigma_k\right)\right\}P\left(\sigma_k|\eta_k,\tau\right)P\left(\eta_k\right)\right\}P\left(\tau|\sigma_\varepsilon\right)P\left(\sigma_\varepsilon\right)$$

is transformed into the one with the inverse gamma distributions.

$$P\left(\mathbf{y}\right) = \prod_{i=1}^{M} P\left(\mathbf{y}_i|\sigma_\varepsilon^2, \{x_{ki}\}\right) \prod_{k=1}^{P}\left\{\prod_{i=1}^{M}\left\{P\left(x_{ki}|\sigma_k^2\right)\right\}P\left(\sigma_k^2|\nu_k\right)P\left(\nu_k|\eta_k^2,\tau^2\right)P\left(\eta_k^2|\zeta_k\right)P\left(\zeta_k\right)\right\} P\left(\tau^2|\xi\right)P\left(\xi|\sigma_\varepsilon^2\right)P\left(\sigma_\varepsilon^2|\phi\right)P\left(\phi\right)$$

Based on the full joint distribution, the full conditional distributions for all the variables are presented.

1.

$$x_{ki}|\mathbf{y}_i, \left\{x_{1i}, \ldots, x_{(k-1)i}, x_{(k+1)i}, \ldots, x_{Pi}\right\}, \sigma_\varepsilon^2, \sigma^2 \sim \mathcal{N}\left(\bar{\mu}_{ki},\, \bar{\sigma}_k^2\right)$$
$$k = 1, \ldots, P, \quad i = 1, \ldots, M$$

where $\bar{\mu}_{ki} = \frac{\sigma_k^2 \sum_{j=1}^{N} \left\{a_{jk}\left(y_{ji} - \sum_{s\neq k} a_{js}\, x_{si}\right)\right\}}{\sigma_k^2 \sum_{j=1}^{N} {a_{jk}}^2 + \sigma_\varepsilon^2}$ and $\bar{\sigma}_k^2 = \frac{\sigma_k^2 \sigma_\varepsilon^2}{\sigma_k^2 \sum_{j=1}^{N} {a_{jk}}^2 + \sigma_\varepsilon^2}$

Proof.

$$y_{ji} - \sum_{s\neq k} a_{js}\, x_{si} = a_{jk} x_{ki} + \varepsilon$$

Let $z_j = y_{ji} - \sum_{s\neq k} a_{js}\, x_{si}$, then,

$$z_j|x_{ki}, \sigma_\varepsilon^2 \sim \mathcal{N}\left(a_{jk}\, x_{ki},\, \sigma_\varepsilon^2\right),\ j = 1, \ldots, N$$
$$x_{ki}|\sigma_k^2 \sim \mathcal{N}\left(0, \sigma_k^2\right)$$

The full conditional distribution for x_{ki} is

$$P\left(x_{ki}\middle|\left\{x_{1i}, \ldots, x_{(k-1)i}, x_{(k+1)i}, \ldots, x_{Pi}\right\}, \sigma_\varepsilon^2, \sigma_k^2\right)$$
$$\propto \prod_{j=1}^{N} \left\{\mathcal{N}\left(z_j|a_{jk}x_{ki},\, \sigma_\varepsilon^2\right)\right\} \mathcal{N}\left(x_{ki}|0,\, \sigma_k^2\right)$$
$$\propto \exp\left(-\frac{\sum_{j=1}^{N}\left(z_j - a_{jk}x_{ki}\right)^2}{2\sigma_\varepsilon^2} - \frac{x_{ki}^2}{2\sigma_k^2}\right)$$
$$\propto \exp\left(-\frac{\left(x_{ki} - \frac{\sigma_k^2 \sum_{j=1}^{N} z_j a_{jk}}{\sigma_k^2 \sum_{j=1}^{N} a_{jk}^2 + \sigma_\varepsilon^2}\right)^2}{2\frac{\sigma_k^2 \sigma_\varepsilon^2}{\sigma_k^2 \sum_{j=1}^{N} a_{jk}^2 + \sigma_\varepsilon^2}}\right)$$

$$\propto \mathcal{N}\left(\frac{\sigma_k^2 \sum_{j=1}^N z_j a_{jk}}{\sigma_k^2 \sum_{j=1}^N a_{jk}^2 + \sigma_\varepsilon^2},\ \frac{\sigma_k^2 \sigma_\varepsilon^2}{\sigma_k^2 \sum_{j=1}^N a_{jk}^2 + \sigma_\varepsilon^2}\right)$$

■

2.

$$\sigma_k^2 | x_{k1}, \ldots, x_{kM}, \nu_k \sim \mathcal{IG}\left(\frac{M+1}{2},\ \frac{\sum_{i=1}^M (x_{ki})^2}{2} + \frac{1}{\nu_k}\right),\ k = 1, \ldots, P$$

Proof.

$$\begin{aligned}
&P\left(\sigma_k^2 \middle| \{x_{ki}\}_{i=1}^M,\ \nu_k\right) \\
&\propto \prod_{i=1}^M \left\{\mathcal{N}\left(x_{ki}|0,\ \sigma_k^2\right)\right\} \mathcal{IG}\left(\sigma_k^2 \middle| \frac{1}{2},\ \frac{1}{\nu_k}\right) \\
&\propto \left(\frac{1}{\sigma_k}\right)^M \exp\left(-\frac{\sum_{i=1}^M (x_{ki})^2}{2\sigma_k^2}\right) \left(\sigma_k^2\right)^{-\frac{3}{2}} \exp\left(-\frac{1}{\nu_k \sigma_k^2}\right) \\
&\propto \left(\sigma_k^2\right)^{-\left(\frac{M}{2}+\frac{1}{2}\right)-1} \exp\left(-\left\{\frac{\sum_{i=1}^M (x_{ki})^2}{2} + \frac{1}{\nu_k}\right\} \frac{1}{\sigma_k^2}\right) \\
&\propto \mathcal{IG}\left(\frac{M+1}{2},\ \frac{\sum_{i=1}^M (x_{ki})^2}{2} + \frac{1}{\nu_k}\right)
\end{aligned}$$

■

3.

$$\nu_k | \sigma_k^2, \tau^2, \eta_k^2 \sim \mathcal{IG}\left(1,\ \frac{1}{\sigma_k^2} + \frac{1}{\tau^2\, \eta_k^2}\right),\ k = 1, \ldots, P$$

Proof.

$$
\begin{aligned}
& P\left(\nu_k \middle| \sigma_k^2,\, \tau^2,\, \eta_k^2\right) \\
& \propto \mathcal{IG}\left(\sigma_k^2 \middle| \frac{1}{2},\, \frac{1}{\nu_k}\right) \mathcal{IG}\left(\nu_k \middle| \frac{1}{2},\, \frac{1}{\tau^2\,\eta_k^2}\right) \\
& \propto \frac{\left(\frac{1}{\nu_k}\right)^{\frac{1}{2}}}{\Gamma\left(\frac{1}{2}\right)} \exp\left(-\frac{1}{\nu_k \sigma_k^2}\right) (\nu_k)^{-\frac{3}{2}} \exp\left(-\frac{1}{\tau^2\,\eta_k^2 \nu_k}\right) \\
& \propto (\nu_k)^{-1-1} \exp\left(-\left\{\frac{1}{\sigma_k^2} + \frac{1}{\tau^2\,\eta_k^2}\right\} \frac{1}{\nu_k}\right) \\
& \propto \mathcal{IG}\left(1,\, \frac{1}{\sigma_k^2} + \frac{1}{\tau^2\,\eta_k^2}\right)
\end{aligned}
$$

■

4.

$$
\eta_k^2 | \tau^2, \nu_k, \zeta_k \sim \mathcal{IG}\left(1, \frac{1}{\tau^2\,\nu_k} + \frac{1}{\zeta_k}\right),\; k = 1, \ldots, P
$$

Proof.

$$
\begin{aligned}
& P\left(\eta_k^2 \middle| \tau^2, \nu_k, \zeta_k\right) \\
& \propto \mathcal{IG}\left(\nu_k \middle| \frac{1}{2},\, \frac{1}{\tau^2\,\eta_k^2}\right) \mathcal{IG}\left(\eta_k^2 \middle| \frac{1}{2},\, \frac{1}{\zeta_k}\right) \\
& \propto \frac{\left(\frac{1}{\tau^2\,\eta_k^2}\right)^{\frac{1}{2}}}{\Gamma\left(\frac{1}{2}\right)} \exp\left(-\frac{1}{\tau^2\,\eta_k^2 \nu_k}\right) \left(\eta_k^2\right)^{-\frac{3}{2}} \exp\left(-\frac{1}{\zeta_k \eta_k^2}\right) \\
& \propto \left(\eta_k^2\right)^{-1-1} \exp\left(-\left\{\frac{1}{\tau^2\,\nu_k} + \frac{1}{\zeta_k}\right\} \frac{1}{\eta_k^2}\right) \\
& \propto \mathcal{IG}\left(1,\, \frac{1}{\tau^2\,\nu_k} + \frac{1}{\zeta_k}\right)
\end{aligned}
$$

■

5.

$$\tau^2|\eta_k^2, \nu_k, \xi \sim \mathcal{IG}\left(\frac{P+1}{2}, \sum_{k=1}^{P}\left(\frac{1}{\eta_k^2\,\nu_k}\right) + \frac{1}{\xi}\right)$$

Proof.

$$\begin{aligned}
&P\left(\tau^2\middle|\{\nu_k\}_{k=1}^{P}, \{\eta_k\}_{k=1}^{P}, \xi\right) \\
&\propto \prod_{k=1}^{P}\left\{\mathcal{IG}\left(\nu_k\middle|\frac{1}{2}, \frac{1}{\tau^2\,\eta_k^2}\right)\right\}\mathcal{IG}\left(\tau^2\middle|\frac{1}{2}, \frac{1}{\xi}\right) \\
&\propto \frac{\left(\frac{1}{\tau^2\,\eta_k^2}\right)^{\frac{P}{2}}}{\Gamma\left(\frac{1}{2}\right)}\exp\left(-\sum_{k=1}^{P}\left\{\frac{1}{\tau^2\,\eta_k^2\nu_k}\right\} - \frac{1}{\xi\,\tau^2}\right)\left(\tau^2\right)^{-\frac{3}{2}} \\
&\propto \left(\tau^2\right)^{-\frac{P+1}{2}-1}\exp\left(-\left\{\sum_{k=1}^{P}\left(\frac{1}{\eta_k^2\,\nu_k}\right) + \frac{1}{\xi}\right\}\frac{1}{\tau^2}\right) \\
&\propto \mathcal{IG}\left(\frac{P+1}{2}, \sum_{k=1}^{P}\left(\frac{1}{\eta_k^2\,\nu_k}\right) + \frac{1}{\xi}\right)
\end{aligned}$$

■

6.

$$\zeta_k|\eta_k^2, w^2 \sim \mathcal{IG}\left(1, \frac{1}{\eta_k^2} + \frac{1}{w^2}\right),\ k = 1, \ldots, P$$

Proof.

$$\begin{aligned}
&P\left(\zeta_k\middle|\eta_k^2, w^2\right) \\
&\propto \mathcal{IG}\left(\eta_k^2\middle|\frac{1}{2}, \frac{1}{\zeta_k}\right)\mathcal{IG}\left(\zeta_k\middle|\frac{1}{2}, \frac{1}{w^2}\right)
\end{aligned}$$

$$
\begin{aligned}
&\propto \frac{\left(\frac{1}{\zeta_k}\right)^{\frac{1}{2}}}{\Gamma\left(\frac{1}{2}\right)} \exp\left(-\frac{1}{\zeta_k \eta_k^2}\right) (\zeta_k)^{-\frac{3}{2}} \exp\left(-\frac{1}{w^2\,\zeta_k}\right) \\
&\propto (\zeta_k)^{-1-1} \exp\left(-\left\{\frac{1}{\eta_k^2}+\frac{1}{w^2}\right\}\frac{1}{\zeta_k}\right) \\
&\propto \mathcal{IG}\left(1,\ \frac{1}{\eta_k^2}+\frac{1}{w^2}\right)
\end{aligned}
$$

■

7.

$$
\xi|\tau^2, \sigma_\varepsilon^2 \sim \mathcal{IG}\left(1, \frac{1}{\tau^2}+\frac{1}{\sigma_\varepsilon^2}\right) \tag{2.22}
$$

Proof.

$$
\begin{aligned}
&P\left(\xi\middle|\tau^2,\ \sigma_\varepsilon^2\right) \\
&\propto \mathcal{IG}\left(\tau^2\middle|\frac{1}{2},\ \frac{1}{\xi}\right)\mathcal{IG}\left(\xi\middle|\frac{1}{2},\ \frac{1}{\sigma_\varepsilon^2}\right) \\
&\propto \frac{\left(\frac{1}{\xi}\right)^{\frac{1}{2}}}{\Gamma\left(\frac{1}{2}\right)} \exp\left(-\frac{1}{\xi\,\tau^2}\right)(\xi)^{-\frac{3}{2}} \exp\left(-\frac{1}{\sigma_\varepsilon^2\,\xi}\right) \\
&\propto (\xi)^{-1-1} \exp\left(-\left\{\frac{1}{\tau^2}+\frac{1}{\sigma_\varepsilon^2}\right\}\frac{1}{\xi}\right) \\
&\propto \mathcal{IG}\left(1,\ \frac{1}{\tau^2}+\frac{1}{\sigma_\varepsilon^2}\right)
\end{aligned}
$$

■

8.

$$
\sigma_\varepsilon^2|\mathbf{y}, \{\mathbf{x}_i\}_{i=1}^M, \xi, \phi \sim \mathcal{IG}\left(\alpha_0,\ \beta_0\right)
$$

where $\alpha_0 = \frac{MN}{2} + 1$ and
$\beta_0 = \frac{\sum_{i=1}^{M}\sum_{j=1}^{N}\left(y_{ji} - \sum_{k=1}^{P} a_{jk}x_{ki}\right)^2}{2} + \frac{1}{\xi} + \frac{1}{\phi}$

Proof.

$$
\begin{aligned}
& P\left(\sigma_\varepsilon^2 \middle| \mathbf{y}, \{\mathbf{x}_i\}_{i=1}^{M}, \xi, \phi\right) \\
& \quad \propto \prod_{i=1}^{M}\prod_{j=1}^{N} \left\{P\left(y_{ji} \middle| \{\mathbf{x}_i\}_{i=1}^{M}, \sigma_\varepsilon^2\right)\right\} P\left(\xi | \sigma_\varepsilon^2\right) P\left(\sigma_\varepsilon^2 | \phi\right) \\
& \propto \prod_{i=1}^{M}\prod_{j=1}^{N} \left\{\mathcal{N}\left(y_{ji} \middle| \sum_{k=1}^{P} a_{jk}x_{ki},\, \sigma_\varepsilon^2\right)\right\} \mathcal{IG}\left(\xi \middle| \frac{1}{2},\, \frac{1}{\sigma_\varepsilon^2}\right) \\
& \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \mathcal{IG}\left(\sigma_\varepsilon^2 \middle| \frac{1}{2},\, \frac{1}{\phi}\right) \\
& \propto \prod_{i=1}^{M}\prod_{j=1}^{N} \left\{\exp\left(-\frac{\left(y_{ji} - \sum_{k=1}^{P} a_{jk}x_{ki}\right)^2}{2\sigma_\varepsilon^2}\right)\right\} \\
& \quad \left(\frac{1}{\sigma_\varepsilon^2}\right)^{\frac{MN}{2}+\frac{1}{2}} \exp\left(-\frac{1}{\sigma_\varepsilon^2 \xi}\right) \left(\sigma_\varepsilon^2\right)^{-\frac{3}{2}} \exp\left(-\frac{1}{\phi\, \sigma_\varepsilon^2}\right) \\
& \propto \left(\sigma_\varepsilon^2\right)^{-\left(\frac{MN}{2}+1\right)-1} \exp\left(-\frac{\beta_0}{\sigma_\varepsilon^2}\right) \\
& \propto \mathcal{IG}\left(\alpha_0,\, \beta_0\right)
\end{aligned}
$$

where $\alpha_0 = \frac{MN}{2} + 1$ and
$\beta_0 = \frac{\sum_{i=1}^{M}\sum_{j=1}^{N}\left(y_{ji} - \sum_{k=1}^{P} a_{jk}x_{ki}\right)^2}{2} + \frac{1}{\xi} + \frac{1}{\phi}$ ■

9.

$$\phi | \sigma_\varepsilon^2 \sim \mathcal{IG}\left(1, \frac{1}{\sigma_\varepsilon^2} + 1\right)$$

Proof.

$$
\begin{aligned}
& P\left(\phi|\nu_k,\, \eta_k,\, \xi\right) \\
& \propto \mathcal{IG}\left(\sigma_\varepsilon^2 \middle| \frac{1}{2},\, \frac{1}{\phi}\right) \mathcal{IG}\left(\phi \middle| \frac{1}{2},\, 1\right) \\
& \propto \frac{\left(\frac{1}{\phi}\right)^{\frac{1}{2}}}{\Gamma\left(\frac{1}{2}\right)} \exp\left(-\frac{1}{\phi\,\sigma_\varepsilon^2}\right) (\phi)^{-\frac{3}{2}} \exp\left(-\frac{1}{\phi}\right) \\
& \propto (\phi)^{-1-1} \exp\left(-\left\{\frac{1}{\sigma_\varepsilon^2}+1\right\}\frac{1}{\phi}\right) \\
& \propto \mathcal{IG}\left(1,\, \frac{1}{\sigma_\varepsilon^2}+1\right)
\end{aligned}
$$

■

2.C Design Matrices of the Case Study

Process I

$$
\mathbf{A} = \begin{bmatrix}
1 & 0.7857 & -0.7857 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 1.1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0.4011 & -0.7857 & 0 & 0.3846 & 1 & 0.3846 & -0.3846 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0.4011 & -0.7857 & 0 & 0.3846 & 0 & 0.1215 & -0.3846 & 0 & 0.2632 & 1
\end{bmatrix}
$$

Process II

$$\mathbf{A} = \begin{bmatrix} 1 & 0.7857 & -0.7857 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1.1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.4011 & -0.7857 & 0 & 0.3846 & 1 & 0.3846 & -0.3846 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & -0.0957 & 0 & 0 & 1 & 0.4 & -0.0957 \end{bmatrix}$$

3 STATISTICAL MODELING AND ANALYSIS OF K-LAYER COVERAGE OF TWO-DIMENSIONAL MATERIALS IN INKJET PRINTING PROCESSES *

Two-dimensional layered materials/flakes, also known as crystalline atom-thick layer nanosheets, have recently been receiving great attention in electronics fabrication due to their unique and intriguing properties. The k-layer coverage area (i.e., the area covered by k number of overlapping layers) of the printed flake pattern significantly impacts on the properties of the printed electronics. In this work, we constructed a statistical model to describe the k-layer coverage of randomly distributed two-dimensional materials. A series of results are obtained to provide not only the expectation but also the variance of the coverage area. The boundary effects on the random flakes coverage are also studied. In addition, an approximated statistical testing approach is also developed in this work to detect abnormal coverage patterns. The case studies based on simulated data and real flakes images obtained from the inkjet printing process demonstrate the accuracy and effectiveness of the proposed model and analysis methods.

*This chapter is based on the paper: **Lee, J.**, S. Zhou, and J. Chen (2021). Statistical modeling and analysis of *k*-layer coverage of two-dimensional materials in inkjet printing processes. *Technometrics*, 63(3), 410–420. https://doi.org/10.1080/00401706.2020.1805020.

3.1 Introduction

Two-dimensional layered materials, also known as crystalline atom-thick layer nanosheets, have recently been receiving great attention in electronics fabrication due to their unique and intriguing properties. Graphene, for example, has high electronic and thermal conductivity, optical transparency, and mechanical strength and flexibility (Li et al., 2014). After the discovery of an exfoliation method of graphene from graphite by Novoselov (2004), a large body of literature has been dealing with graphene and its variants, such as graphene oxide, as well as other two-dimensional materials, such as molybdenum disulfide (MoS_2). The inkjet printing technique has been gaining growing interests to fabricate electronics with these two-dimensional materials (Sowade et al., 2016). Inkjet printing is an additive patterning technique that deposits functional ink, which may contain two-dimensional materials as a solute, through nozzles onto the substrates. Inkjet-printed graphene and its variants have shown promising opportunities in a wide range of applications (Li et al., 2014), including sensors (Dua et al., 2010; Huang et al., 2011), wearable textiles (Li et al., 2012), antennas (Shin et al., 2011), and memory (Huber et al., 2017).

This work is motivated by the recently developed field-effect transistor(FET) sensors, which are used to detect the heavy metal ions in water (Chang et al., 2019). The sensor is illustrated in Figure 3.1.1. In such a sensor, the flakes of two-dimensional materials, namely rGO flakes, are inkjet-printed on the substrate. Two electrodes, named drain and source, are put on the printed pattern. As the gate voltage (denoted by V_g in the figure) is applied, the current between drain and source can be measured. When the sensor is exposed to the water contaminated by

heavy metal ions, the current between the drain and source will deviate from its normal value.

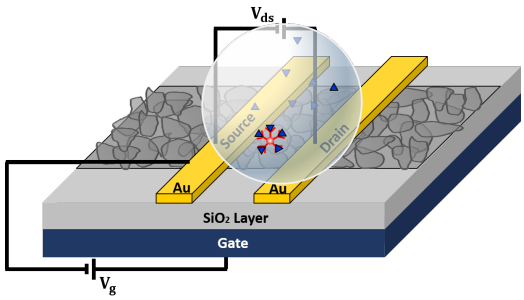


Figure 3.1.1: Field-effect transistor sensor for heavy metal detection in water

The performance of the FET sensor is most significantly influenced by the coverage and thickness (i.e., number of overlapping layers) of flakes on the area between two electrodes. The FET sensor needs a high gap in the currents between on and off states. This can be achieved when flakes cover more surface with less flake overlap (Sui and Appenzeller, 2009). In the inkjet printing process, flakes distribute randomly between the electrodes, and we cannot directly control them. It is highly desirable to have a model to describe and analyze the randomness in the coverage and the overlapping of randomly distributed two-dimensional materials. With such a model, we can link the process parameters to the flake distribution and predict the performance including both the sensitivity and repeatability of the fabricated sensors. In addition, we can use the model to identify if the flakes are uniformly distributed on the substrate, which is important for the process quality control purposes.

The existing literature on the random two-dimensional flakes coverage cannot address the practical needs in the inkjet-printing process. The study directly on the

inkjet-printed pattern is mostly done based on the first principles. Researchers have investigated the drying process based on the physical movements of particles (Deegan et al., 1997; Fischer, 2002; Hu and Larson, 2006). The physics-based research gives insights into flakes behaviors during the ink's drying. However, the works in this category do not deal with the statistical behavior of flake coverage. The most widely-known statistical model for the random two-dimensional coverage is the boolean model. The boolean model is based on the union set of every flake set whose location and shape is random (Chiu et al., 2013). The area covered by one or more layers is equally treated as *covered*, and the complement set as *uncovered*. The boolean model is popularly used in the literature on the coverage of random wireless sensor networks (Hsin and Liu, 2004; Liu et al., 2005; Liu and Towsley, 2003, 2004; Liu et al., 2005). There are two significant limitations of this model in pursuit of our objective. (1) It cannot explain multi-layer coverage. We need a more sophisticated model that can distinguish the thickness of the coverage because the performance of the electronics differs with respect to different thicknesses. (2) Its main focus is on the expectation of the coverage. To quantify the uncertainty in the manufacturing process, the variance of coverage should also be modeled and analyzed.

Some relevant literature can also be found in the application of randomly deployed wireless sensor networks. Their main interest is the detectability of special events such as wildfire by randomly deployed sensors. Many researchers have studied this field (Hsin and Liu, 2004; Liu et al., 2005; Liu and Towsley, 2003, 2004; Liu et al., 2005). However, it is hard to adopt these models because (1) fixed radius

circular ranges are assumed, (2) only expectation is studied, (3) many works used the boolean model; therefore, they cannot distinguish different thicknesses. In Wan and Chih-Wei (2006), the authors studied the problem of a point/region being covered by *at least* k sensors. However, the *exact* k-layer coverage problem is not investigated. Furthermore, in their study, the coverage area of a single sensor is assumed to be a circle with a fixed radius. However, in the problem we are facing, flakes are randomly created from the ink fabrication process with different sizes and shapes. As a result, their model cannot address our needs.

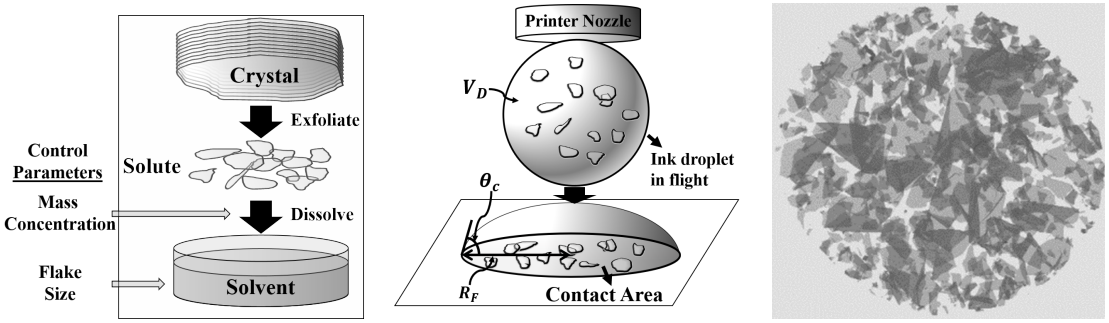
In this chapter, to fill this research gap, we establish a statistical model that describes the uncertainties in the flakes dispersion and coverage with respect to different levels of thicknesses, namely *k-layer coverage fraction*, of the printed pattern. The flakes are defined by combining the uniformly distributed random locations and random shapes. A series of analytic results are obtained providing the expectation and variance of the coverage fraction with different thicknesses. The boundary effects on the random flakes coverage are also studied. Based on this model, we further propose a statistical testing method that detects the nonuniform printed pattern. These proposed models and methods are tested and validated through extensive numerical study and real flakes distribution images obtained from an inkjet process.

The rest of the chapter is organized as follows. Section 3.2 introduces the inkjet printing process and the relevant process parameters. Some basic assumptions on the flake distribution based on the process physics are also introduced. Section 3.3 delineates the proposed random flakes model and the expectation and variance of

the coverage fraction with different thicknesses. The statistical testing approach to detect abnormal flake coverage is presented in Section 3.4. A numerical study that validates the proposed model is presented in Section 3.5. The validation based on the real flakes image is conducted in Section 3.6. Finally, we draw a conclusion and discussion in Section 3.7.

3.2 Inkjet Printing and Basic Assumptions

The inkjet printing process consists of two separate steps: ink preparation and inkjet printing. To provide the desired functionality, the ink is customized in the ink preparation step by controlling the mass concentration and size of the flakes in the ink (Figure 3.2.1(a)). First, two-dimensional material (e.g. graphene) flakes are exfoliated from crystal (e.g. graphite) and dissolved into the solvent up to the target mass concentration of the flakes in the ink. Then, the flakes sizes in the ink are reduced up to the target size by controlling the exposure time to the ultrasonic milling. The prepared ink is printed through the inkjet printer (Figure 3.2.1(b)). After the ink dries out, the dispersed flakes are left with a pattern, providing functionality. The real image of a flake pattern printed by an inkjet printer is presented in Figure 3.2.1(c) that is produced in He and Derby (2017).



(a) (b) (c)

Figure 3.2.1: Two stages of two-dimensional material inkjet printing process are described: (a) ink preparation process and (b) inkjet printing process. (c) An image of the real pattern printed by an inkjet printer produced.

Table 3.2.1: Physical parameters

Symbol	Definition
c_F	Mass concentration of the two-dimensional material flakes in the ink droplet
h_F	Height of the two-dimensional material flakes
ρ_F	Density of the two-dimensional material flakes
V_D	Volume of the ink droplet
θ_c	Contact angle of the ink droplet
R_F	Radius of the contact area of the ink droplet on the substrate

The critical parameters in the printing step are summarized in Table 3.2.1. Among these parameters, h_F and ρ_F are constant material properties. The volume of the droplet ejected from an inkjet printer, V_D, can also be viewed as constant because V_D can be precisely controlled in a modern inkjet printing process (Singh et al., 2010) at picoliter level. Once the droplet falls on the substrate, it forms a spherical cap as shown in Figure 3.2.1(b). The contact angle θ_c is determined by the combination of the ambient temperature and the material properties of the substrate and droplet. The relationship among the droplet volume V_D, contact

angle θ_c, and radius of the contact area R_F is given as (Picknett and Bexon, 1977)

$$V_D = \frac{\pi}{3}\left(1-\cos\theta_c\right)^2\left(2+\cos\theta_c\right)\left(\frac{R_F}{\sin\theta_c}\right)^3 \tag{3.1}$$

R_F can be obtained given V_D and θ_c from (3.1). The mass concentration of the flakes in a droplet, c_F, is defined by the mass of the flakes in the ink droplet divided by the volume of the droplet. Due to the random dispersion of the flakes in the ink, c_F of different droplets are different. Thus, c_F of a droplet should be viewed as a random variable.

With $\mathbb{E}\left[c_F\right]$, we can derive the expected number of flakes in the droplet. Specifically, the mass of the flakes in the droplet is equal to the total summed sizes of flakes $\sum_{i=1}^{N}|F_i|$ multiplied by the height and density of the flakes, where $|F_i|$ denotes the area of ith flake and N is the number of flakes in the droplet; therefore, $\sum_{i=1}^{N}|F_i| = \left(c_F V_D\right)/\left(h_F\rho_F\right)$. Because the flake size $|F_i|$ is an independent random variable from each other and N, $\mathbb{E}_N\left[\mathbb{E}\left[\sum_{i=1}^{N}|F_i|\middle|N\right]\right] = \mathbb{E}_N\left[N\,\mathbb{E}\left[|F_i||N\right]\right] = \mathbb{E}\left[N\right]\mathbb{E}\left[|F_i|\right]$. Then, the expectation of the number of flakes, $\mathbb{E}\left[N\right]$, is

$$\mathbb{E}\left[N\right] = \frac{\mathbb{E}\left[c_F\right]V_D}{\mathbb{E}\left[|F_i|\right]h_F\rho_F} \tag{3.2}$$

According to the physical principles of the inkjet printing process, we can have the following three non-restrictive assumptions:

- Flakes are uniformly distributed within the printed pattern: from a large body of literature, it is widely known that flakes are uniformly distributed when a coffee-ring does not form (Fischer, 2002; He, 2016).

- The number of flakes in a printed pattern is an independent Poisson random variable: the number of points drawn from an area where the point event occurrence follows the continuous uniform distribution is known to follow the Poisson distribution. This assumption has been widely used in the literature. For example, the number of printed cells in a printing process is known to follow the Poisson distribution (Kim et al., 2016; Merrin et al., 2007).

- Flake shape and size are independent random variables from the other process parameters: because the flake shape and size are determined before the printing process, the flake shapes and sizes are independent random variables from the other process parameters in the inkjet printing step.

The relationship in (3.2) and the above assumptions will be used in the following k-coverage model.

3.3 *k*-Layer Random Coverage Fraction

3.3.1 Statistical Model of Flake Coverage

In this chapter, we propose a statistical flakes model that accounts for the random coverage and thickness of flakes in terms of the expectation and variance of the random coverage fraction. We consider random flakes. A flake F_i $(i = 1, \ldots, N)$ has its center location ϕ_i uniformly distributed over $\mathbb{S}_F \subset \mathbb{R}^2$, and $\mathbb{S}_F$ is the *flake space* or *printed space* within which flakes are deposited. The number of flakes N is a Poisson random variable with the parameter of mean $\mathbb{E}[N]$; thus, the flakes center

locations $(\phi_1, \phi_2, \ldots)$ follow the Poisson process. The shape of the flake F_i is defined by an independent and identically distributed (i.i.d.) random compact set, and the size of the flake F_i is denoted by $|F_i|$. A rigorous definition of the random flakes is included in Appendix 3.A.1. The definition of the random flakes is illustrated in Figure 3.3.1.

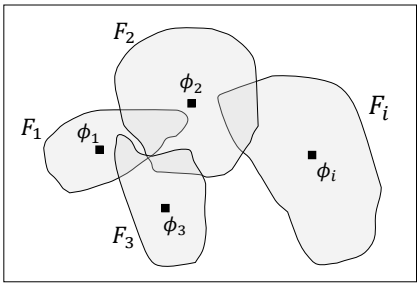


Figure 3.3.1: Random flakes are considered. The random flakes' center locations are uniformly distributed, and their shapes are defined by random compact sets.

Our objective is to study the expectation and variation of the coverage fraction that is covered by k layers of overlapping flakes, hereafter called k-layer coverage fraction, deposited in the space $\mathbb{S}_F$. The k-layer coverage fraction, denoted by $\mathcal{C}_k$, is evaluated by measuring the thickness at every point in $\mathbb{S}_F$ through the indicator $T_k(\mathbf{z})$ $(k = 0, 1, \ldots)$; $T_k(\mathbf{z})$ is a random variable that is 1 if the point $\mathbf{z} \in \mathbb{S}_F$ is covered by k layers of flakes or 0 otherwise. Conditioned on the point $\mathbf{z}$, $T_k(\mathbf{z})$ is a Bernoulli random variable. $T_k(\mathbf{z})$ is statistically dependent on $T_k(\mathbf{w})$ for $\mathbf{w} \in \mathbb{S}_F$. Two points with a closer distance have a higher dependency because a near point $\mathbf{w}$ is also likely to be covered by the same flake covering $\mathbf{z}$. The k-layer coverage

fraction, $\mathcal{C}_k$, can be calculated as

$$\mathcal{C}_k = \frac{\int_{\mathbf{z}\in\mathbb{S}_F} T_k(\mathbf{z})\, d\mathbf{z}}{\int_{\mathbf{z}\in\mathbb{S}_F} d\mathbf{z}} \tag{3.3}$$

We first introduce the expectation and variance of $\mathcal{C}_k$ under the absence of the boundary effects in Section 3.3.2.1 and 3.3.2.2. In general, the flake distribution close to the boundary of the printed area (i.e., the contact area of the droplet on the substrate) is different from the distribution around the center of the area. Such difference refers to the boundary effects. If the printed area is much larger than the size of flakes, the boundary effects can be ignored. Otherwise, the boundary effects may be significant. The boundary effects on the mean and variance of $\mathcal{C}_k$ are considered in Section 3.3.3. The method to calculate the exact expectation and variance with the random circular flakes is presented in Section 3.3.4. To make the flow smooth, the details of the mathematical derivations of these results are deferred to Appendix.

3.3.2 k-Layer Coverage Fraction Without Boundary Effects

3.3.2.1 Expectation of k-Layer Coverage Fraction

The expectation of k-layer random coverage fraction is obtained by evaluating the thickness at every point over the space $\mathbb{S}_F$. The probability of any point $\mathbf{z}$ being covered by any k flakes is the same over $\mathbb{S}_F$ because every flake is uniformly distributed; therefore, $\mathbb{E}[\mathcal{C}_k] = \mathbb{E}[T_k(\mathbf{z})]$. Because $T_k(\mathbf{z})$ is a binary random variable, $\mathbb{E}[T_k(\mathbf{z})] = \mathbb{P}[T_k(\mathbf{z})]$ where $\mathbb{P}[T_k(\mathbf{z})]$ is the probability that a point $\mathbf{z}$ is covered by

exact k flakes. Then, $\mathbb{E}[\mathcal{C}_k]$ is as follows.

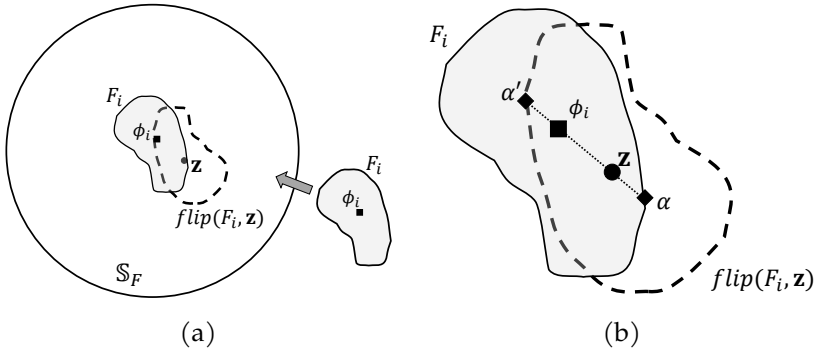
$$\mathbb{E}[\mathcal{C}_k] = \mathbb{E}[T_k(\mathbf{z})] = \mathbb{P}[T_k(\mathbf{z})] = \frac{\exp\{-\mathbb{E}[N]\,p\}\,(\mathbb{E}[N]\,p)^k}{k!} \tag{3.4}$$

where N is the number of flakes in $\mathbb{S}_F$ following the Poisson distribution, and p is the expected probability that a point $\mathbf{z}$ in $\mathbb{S}_F$ is covered by a single random flake:

$$p = \mathbb{E}\left[\frac{|F_i|}{|\mathbb{S}_F|}\right] \tag{3.5}$$

We denote $p_i = |F_i|/|\mathbb{S}_F|$ as the probability that the flake F_i covers a point $\mathbf{z}$ (i.e., $p = \mathbb{E}[p_i]$). The result in (3.4) is obtained through deriving the probability that the point $\mathbf{z}$ is covered by any k among N flakes. The detailed derivation of (3.4) can be found in Appendix 3.B.1.

Point $\mathbf{z}$ is covered by the flake F_i when the flake's center location ϕ_i is located within a specific region, $flip(F_i, \mathbf{z}) \subset \mathbb{R}^2$; $flip(F_i, \mathbf{z})$ is defined as the region where F_i is rotated by 180° and translated so that its center is on $\mathbf{z}$ (Figure 3.3.2(a)). Because ϕ_i is uniformly distributed, the probability that ϕ_i is located within $flip(F_i, \mathbf{z})$ is $|F_i| \,/\, |\mathbb{S}_F|$, which does not rely on the flake center location ϕ_i nor the point $\mathbf{z}$. The rationale for using $flip(F_i, \mathbf{z})$ is illustrated in Figure 3.3.2(b). For the flake F_i to cover $\mathbf{z}$, the distance between ϕ_i and $\mathbf{z}$ must be shorter than the distance between ϕ_i and α that is the point where a line from ϕ_i crossing $\mathbf{z}$ meets the boundary of the flake. Equivalently, if ϕ_i falls in between α' and $\mathbf{z}$, meaning ϕ_i is within $flip(F_i, \mathbf{z})$, $\mathbf{z}$ is covered by the flake. The mathematical definition of $flip(F_i, \mathbf{z})$ can be found in Appendix 3.A.2.



(a) (b)

Figure 3.3.2: (a) The event that a point $\mathbf{z}$ is covered by the flake F_i is equivalent to the event that the center location ϕ_i is located within the dashed area, denoted by $flip\,(F_i, \mathbf{z})$ (point reflection of F_i whose center is on $\mathbf{z}$). (b) Illustration of rationale for using $flip\,(F_i, \mathbf{z})$.

The expectation of $\mathcal{C}_k$ is in the form of the Poisson probability mass function with its parameter $\mathbb{E}[N]\,p$. It is notable that $\mathbb{E}[\mathcal{C}_k]$ does not depend on the individual flake size or shape, but it is determined by $\mathbb{E}[N]\,p = \mathbb{E}\left[\sum_i^N |F_i|\right] / |\mathbb{S}_F|$, which is the relative total size of the flakes to the size of the printed pattern. This relative size is proportional to the mass concentration of the flakes in the ink. In other words, $\mathbb{E}[\mathcal{C}_k]$ is determined by the mass concentration.

3.3.2.2 Variance of k-Layer Coverage Fraction

The variance of $\mathcal{C}_k$ is obtained through the spatial correlation, which is represented by the covariance or the correlation coefficient, of the thickness at every pair of two points in $\mathbb{S}_F$.

$$\mathbb{V}\mathrm{ar}\left[\mathcal{C}_k\right] = \frac{1}{\left|\mathbb{S}_F\right|^2}\left\{\int_{\mathbf{z}\in\mathbb{S}_F}\int_{\mathbf{w}\in\mathbb{S}_F}\mathbb{C}\mathrm{ov}\left[T_k\left(\mathbf{z}\right),\, T_k\left(\mathbf{w}\right)\right]\, d\mathbf{w}\, d\mathbf{z}\right\} \tag{3.6}$$

$$= \mathbb{V}\mathrm{ar}\left[T_k\left(\mathbf{z}\right)\right] \frac{\int_{\mathbf{z}\in\mathbb{S}_F}\int_{\mathbf{w}\in\mathbb{S}_F} \mathbb{C}\mathrm{orr}\left[T_k\left(\mathbf{z}\right), T_k\left(\mathbf{w}\right)\right] d\mathbf{w}\, d\mathbf{z}}{\left|\mathbb{S}_F\right|^2} \leq \mathbb{V}\mathrm{ar}\left[T_k\left(\mathbf{z}\right)\right] \quad (3.7)$$

where $\mathbb{C}\mathrm{ov}$ is covariance and $\mathbb{C}\mathrm{orr}$ is correlation coefficient. The detailed mathematical derivations of (3.6), (3.7), and the following results regarding the variance can be found in Appendix 3.B.2. $\mathbb{V}\mathrm{ar}\left[\mathcal{C}_k\right]$ can be decomposed into two parts, $\mathbb{V}\mathrm{ar}\left[T_k\left(\mathbf{z}\right)\right]$ and the normalized integration of $\mathbb{C}\mathrm{orr}\left[T_k\left(\mathbf{z}\right), T_k\left(\mathbf{w}\right)\right]$. $\mathbb{V}\mathrm{ar}\left[T_k\left(\mathbf{z}\right)\right]$ can be found in terms of $\mathbb{E}\left[T_k\left(\mathbf{z}\right)\right]$ (presented in (3.4)): $\mathbb{V}\mathrm{ar}\left[T_k\left(\mathbf{z}\right)\right] = \mathbb{E}\left[T_k\left(\mathbf{z}\right)\right] - \mathbb{E}\left[T_k\left(\mathbf{z}\right)\right]^2$, for any $\mathbf{z}$ in $\mathbb{S}_F$. Therefore, $\mathbb{V}\mathrm{ar}\left[T_k\left(\mathbf{z}\right)\right]$ does not rely on the individual flake shapes or sizes but it relies on the total size of the flakes. In contrast, smaller flakes will produce smaller correlations of the thickness at two different points ($\mathbb{C}\mathrm{orr}\left[T_k\left(\mathbf{z}\right), T_k\left(\mathbf{w}\right)\right]$), leading to a smaller variance of $\mathcal{C}_k$. The upper-bound of $\mathbb{V}\mathrm{ar}\left[\mathcal{C}_k\right]$ is $\mathbb{V}\mathrm{ar}\left[T_k\left(\mathbf{z}\right)\right]$ by setting $\mathbb{C}\mathrm{orr}\left[T_k\left(\mathbf{z}\right), T_k\left(\mathbf{w}\right)\right] = 1$ for all $\mathbf{z}$ and $\mathbf{w}$. However, this upper-bound is generally not tight.

We can also obtain a more detailed expression of $\mathbb{C}\mathrm{ov}\left[T_k\left(\mathbf{z}\right), T_k\left(\mathbf{w}\right)\right]$ as

$$\sum_{l=0}^{k} \frac{\mathbb{E}\left[N\right]^{2k-l} p_{I\!I}(\mathbf{z},\mathbf{w})^l \left\{p - p_{I\!I}(\mathbf{z},\mathbf{w})\right\}^{2(k-l)}}{l!\left\{(k-l)!\right\}^2} \exp\left\{-\mathbb{E}\left[N\right]\left(2p - p_{I\!I}(\mathbf{z},\mathbf{w})\right)\right\} - \mathbb{E}\left[\mathcal{C}_k\right]^2 \quad (3.8)$$

where $p_{I\!I}(\mathbf{z},\mathbf{w}) = \mathbb{E}\left[p_{I\!I\,i}(\mathbf{z},\mathbf{w})\right]$ is the expected probability that two points $\mathbf{z}$ and $\mathbf{w}$ are covered by a single random flake, and $p_{I\!I\,i}(\mathbf{z},\mathbf{w}) = \mathbb{P}\left[\mathbf{z}\in F_i \cap \mathbf{w}\in F_i\right]$ is the probability that the both points are covered by the same flake F_i, which can be

written as

$$p_{\mathbb{I}i}(\mathbf{z}, \mathbf{w}) = \frac{|(flip\,(F_i, \mathbf{z})) \cap (flip\,(F_i, \mathbf{w}))|}{|\mathbb{S}_F|} \tag{3.9}$$

The basic idea to obtain (3.8) is that $\mathbb{C}\mathrm{ov}\left[T_k\left(\mathbf{z}\right), T_k\left(\mathbf{w}\right)\right]$ can be calculated with the probability of an event that two points $\mathbf{z}$ and $\mathbf{w}$ are both covered by k flakes ($\mathbb{P}\left[T_k\left(\mathbf{z}\right) = 1, T_k\left(\mathbf{w}\right) = 1\right]$). This event can be divided into $k+1$ mutually exclusive collectively exhaustive subevents: both points $\mathbf{z}$ and $\mathbf{w}$ are covered by the l same flakes ($l = 0, 1, \ldots, k$) and by $(k - l)$ layers with different flakes (Figure 3.3.3).

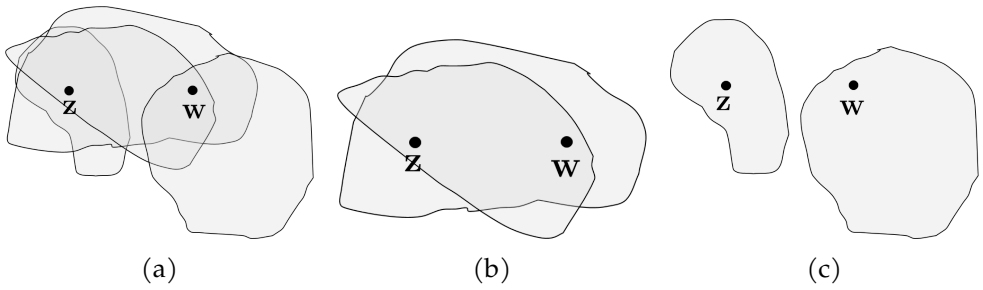


Figure 3.3.3: The event that two points are covered by k layers can be divided into $k+1$ mutually exclusive collectively exhaustive subevents: points $\mathbf{z}$ and $\mathbf{w}$ are covered by the same l flakes and by $k - l$ layers of different flakes. The subevent when $l = 2, k = 3$ is illustrated in (a). Two points are covered by l layers with the same flakes in (b) and by $k - l$ layers with different flakes in (c).

The covariance of two different levels of coverage fractions $\mathbb{C}\mathrm{ov}\left[\mathcal{C}_k, \mathcal{C}_h\right]$ can be

calculated in a similar manner.

$$\mathbb{C}\mathrm{ov}\left[\mathcal{C}_k, \mathcal{C}_h\right] = \frac{1}{\left|\mathbb{S}_F\right|^2} \int_{\mathbf{z}} \int_{\mathbf{w}} \sum_{l=0}^{\min(k,h)} \frac{\mathbb{E}\left[N\right]^{k+h-l} p_{I\!I}(\mathbf{z},\mathbf{w})^l \left\{p - p_{I\!I}(\mathbf{z},\mathbf{w})\right\}^{(k+h-2l)}}{l!\left\{(k-l)!\right\}^2} \times \exp\left[-\mathbb{E}\left[N\right]\left\{2p - p_{I\!I}(\mathbf{z},\mathbf{w})\right\}\right] - \mathbb{E}\left[\mathcal{C}_k\right]\mathbb{E}\left[\mathcal{C}_h\right] d\mathbf{w} d\mathbf{z} \tag{3.10}$$

where $\mathbb{V}\mathrm{ar}\left[\mathcal{C}_k\right] = \mathbb{C}\mathrm{ov}\left[\mathcal{C}_k, \mathcal{C}_k\right]$.

Notice that $p_{I\!I\,i}(\mathbf{z},\mathbf{w})$ in (3.9) depends on the shape of the flakes. Consider, for example, two different shapes of flakes, circle and ellipse, that have the same size $|F_i|$. Then, because the sizes of the flakes are given the same, the major axis of the elliptical flake is longer than the diameter of the circular flake. Therefore, there will exist some points $\mathbf{z}$ and $\mathbf{w}$ that the circular flake cannot cover both points ($p_{I\!I\,i}(\mathbf{z},\mathbf{w}) = 0$) whereas the wider elliptical flake can cover the both points ($p_{I\!I\,i}(\mathbf{z},\mathbf{w}) > 0$).

In the following section, the expectation and variance of $\mathcal{C}_k$ with boundary effects are presented.

3.3.3 k-Layer Coverage Fraction Considering Boundary Effects

The boundary effects can be ignored when the size of the printed region $\mathbb{S}_F$ is significantly larger than the flakes sizes. However, the boundary effect may not be ignored when $\mathbb{S}_F$ is not large compared to the sizes of the flakes.

The boundary of the ink droplet restricts whole flakes to be located within it, which forces the flake center location ϕ_i to be placed in a smaller space $\mathbb{S}_{\phi_i} \subset \mathbb{S}_F$.

The mathematical definition of $\mathbb{S}_{\phi_i}$ can be found in Appendix 3.A.3. This restriction causes complications in the calculation of the mean and variance of k-layer coverage fractions. This restricted flake center space is illustrated in Figure 3.3.4.

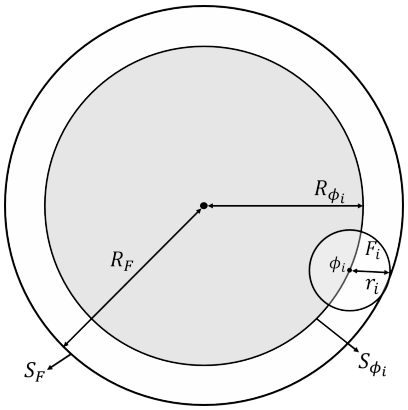


Figure 3.3.4: With the boundary condition, the center point space $\mathbb{S}_{\phi_i}$, where the center point of flake F_i can be located, depends on the size of the flake.

With the boundary effects, the mean and variance of k-layer coverage fraction presented in (3.4) and (3.10) still hold, but the parameter $p(\mathbf{z})$ and $p_{II}(\mathbf{z}, \mathbf{w})$ need to be adjusted accordingly. Although $p(\mathbf{z})$ is constant over the space $\mathbb{S}_F$ when the boundary effects are considered, $p(\mathbf{z})$ relies on point $\mathbf{z}$ with the consideration of boundary effects because of the restriction of $\mathbb{S}_{\phi_i}$. Now, flake F_i covers point $\mathbf{z}$ if the flake's center is located within $\mathbb{S}_{\phi_i} \cap (flip(F_i, \mathbf{z}))$ among its possible position $\mathbb{S}_{\phi_i}$. Thus, $p(\mathbf{z})$ is

$$p(\mathbf{z}) = \mathbb{E}\left[\frac{|\mathbb{S}_{\phi_i} \cap (flip(F_i, \mathbf{z}))|}{|\mathbb{S}_{\phi_i}|}\right] \tag{3.11}$$

Similarly, $p_{I\!I}\left(\mathbf{z},\mathbf{w}\right)$ is as follows.

$$p_{I\!I}\left(\mathbf{z},\mathbf{w}\right)=\mathbb{E}\left[\frac{\left|\mathbb{S}_{\phi_i}\cap\left(flip\left(F_i,\mathbf{z}\right)\right)\cap\left(flip\left(F_i,\mathbf{w}\right)\right)\right|}{\left|\mathbb{S}_{\phi_i}\right|}\right] \tag{3.12}$$

3.3.4 k-Layer Coverage Fraction for Circular Flakes

3.3.4.1 k-Layer Coverage Fraction Without Boundary Effects

Calculation of $\mathbb{V}\mathrm{ar}\left[\mathcal{C}_k\right]$ involves $p_{I\!I}(\mathbf{z},\mathbf{w})$, which relies on the shape of the flakes. In this section, we show how to calculate $p_{I\!I}(\mathbf{z},\mathbf{w})$ based on the random sized circular flakes; the random radius is denoted by r_i, and its probability density function is denoted by $f\left(r_i\right)$. The robustness of the outcome with respect to different flake shapes are presented in the numerical study.

With circular flakes, now, $\left|\left(flip\left(F_i,\mathbf{z}\right)\right)\cap\left(flip\left(F_i,\mathbf{w}\right)\right)\right|$ in (3.9) reduces to $\left|\mathcal{D}(\mathbf{z},r_i)\cap\mathcal{D}(\mathbf{w},r_i)\right|$ where $\mathcal{D}(\mathbf{z},r_i)=\left\{x\mid\left\|x-\mathbf{z}\right\|\leq r_i\right\}$ is a circular disc whose radius is r_i and center is on $\mathbf{z}$. Now, $\left|\mathcal{D}(\mathbf{z},r_i)\cap\mathcal{D}(\mathbf{w},r_i)\right|$ can be found in a closed form (shaded area in Figure 3.3.5). Therefore, $p=\mathbb{E}\left[p_i\right]$ and $p_{I\!I}(\mathbf{z},\mathbf{w})=\mathbb{E}\left[p_{I\!I\,i}(\mathbf{z},\mathbf{w})\right]$ can be obtained as follows.

$$p=\frac{r^2}{R_F^2} \tag{3.13}$$

$$p_{I\!I}(\mathbf{z},\mathbf{w})=\frac{1}{\left|\mathbb{S}_F\right|}\int_{r=\frac{d}{2}}^{\infty}f\left(r\right)\left\{2r^2\cos^{-1}\frac{d}{2r}-d\sqrt{r^2-\frac{d^2}{4}}\right\}dr \tag{3.14}$$

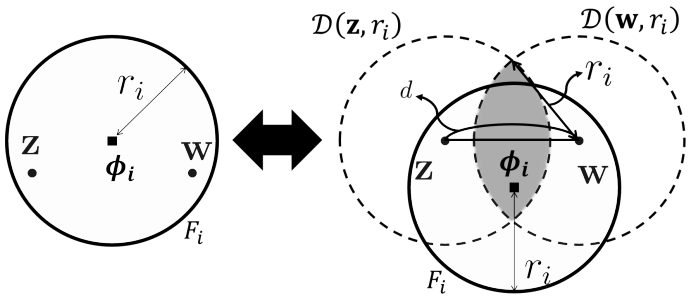


Figure 3.3.5: A circular flake covers both points $\mathbf{z}$ and $\mathbf{w}$ if and only if the center of the flake (square dot, ϕ_i) is located within the overlap of $\mathcal{D}(\mathbf{z}, r_i)$ and $\mathcal{D}(\mathbf{w}, r_i)$ (shaded area) where $\mathcal{D}(\mathbf{z}, r_i)$ is a circular disc whose center is $\mathbf{z}$ with radius r_i.

3.3.4.2 k-Layer Coverage Fraction With Boundary Effects

With the boundary effects, to calculate $p(\mathbf{z})$ and $p_{I\!I}(\mathbf{z}, \mathbf{w})$, a complex geometrical relationship needs to be considered: $\mathbb{S}_{\phi_i} \cap (flip(F_i, \mathbf{z}))$ and $\mathbb{S}_{\phi_i} \cap (flip(F_i, \mathbf{z})) \cap (flip(F_i, \mathbf{w}))$. To make the problem tractable, we propose an approximation method to estimate $p(\mathbf{z})$ and $p_{I\!I}(\mathbf{z}, \mathbf{w})$ by reducing the space to be evaluated.

We want to study $\mathbb{E}[\mathcal{C}_k]$ and $\mathbb{V}\mathrm{ar}[\mathcal{C}_k]$ over a reduced space $\mathbb{S}_e$, namely *evaluation space*. In particular, we want to precisely estimate $p(\mathbf{z})$ and $p_{I\!I}(\mathbf{z}, \mathbf{w})$ for $\mathbb{E}[\mathcal{C}_k]$ and $\mathbb{V}\mathrm{ar}[\mathcal{C}_k]$ while losing the least amount of information on the coverage. We specify $\mathbb{S}_e$ as follows.

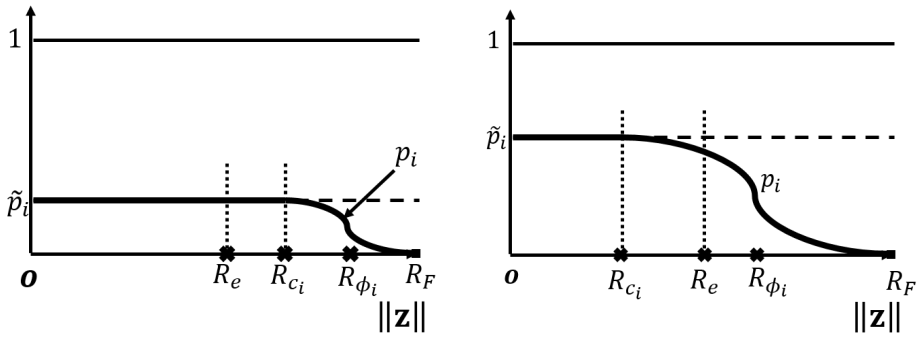
$$\mathbb{S}_e = \{\mathbf{z} | \mathbb{P}(\|\mathbf{z}\| > R_F - 2r_i) \leq \varepsilon\} \tag{3.15}$$

where ε is a very small number, and $\|\cdot\|$ is the Euclidean distance from the origin.

We used 0.001 for ε for this work. Under $\mathbb{S}_e$, $p_i(\mathbf{z})$ is approximated as follows.

$$p_i(\mathbf{z}) \approx \tilde{p}_i = \frac{|F_i|}{|\mathbb{S}_{\phi_i}|} \tag{3.16}$$

Figure 3.3.6 shows p_i values with respect to different $\|\mathbf{z}\|$. When $\mathbf{z}$ falls in $\|\mathbf{z}\| \leq R_{c_i} = R_F - 2r_i$, p_i is constant as $\tilde{p}_i$, and as $\|\mathbf{z}\|$ surpasses R_{c_i}, p_i diminishes to zero at R_F. Therefore, all the points $\mathbf{z} \in \mathbb{S}_e$, will be exactly evaluated by the true value $\tilde{p}_i$ with the probability of $1 - \varepsilon$ (Figure 3.3.6(a)). When the flake radius is very large, some points $\mathbf{z}$ where $R_{c_i} \leq \|\mathbf{z}\| \leq R_e$ have p_i smaller than $\tilde{p}_i$ with a small chance ε (Figure 3.3.6(b)). Therefore, $p(\mathbf{z})$ will be slightly overestimated. $p_{II}(\mathbf{z}, \mathbf{w})$ is also calculated similarly.



(a) p_i is constant within $\|\mathbf{z}\| < R_e$ when the flake radius is small and $r_i \leq 0.5(R_F - R_e)$

(b) p_i reduces within $\|\mathbf{z}\| < R_e$ as $\|\mathbf{z}\|$ increases when the flake radius is large and $r_i > 0.5(R_F - R_e)$

Figure 3.3.6: The probability that a point is covered by k layers of flakes is affected by the boundary effect.

Then, $p(\mathbf{z})$ and $p_{II}(\mathbf{z}, \mathbf{w})$ can be obtained as follows.

$$p(\mathbf{z}) = \int_{r=0}^{R_F/2} \left\{ \frac{r^2}{(R_F - r)^2} \right\} f(r)\, dr + \int_{R_F/2}^{\infty} f(r)\, dr \tag{3.17}$$

$$p_{II}(\mathbf{z},\mathbf{w}) = \int_{r=0}^{R_F/2} \left\{ \frac{|\mathcal{D}(\mathbf{z},r) \cap \mathcal{D}(\mathbf{w},r)|}{(R_F - r)^2} \right\} f(r)\, dr + \int_{R_F/2}^{\infty} f(r)\, dr \tag{3.18}$$

where $\mathcal{D}(\mathbf{z}, r)$ is a circular disc whose center is on $\mathbf{z}$ with radius r, and $|\mathcal{D}(\mathbf{z},r) \cap \mathcal{D}(\mathbf{w},r)| = 2r^2 \cos^{-1}(d/2r) - d\sqrt{r^2 - d^2/4}$.

This method provides a quite accurate approximation of the k-coverage fractions with the boundary effects. Its accuracy is validated in the numerical studies.

3.4 Statistical Testing for Nonuniform Coverage Patterns

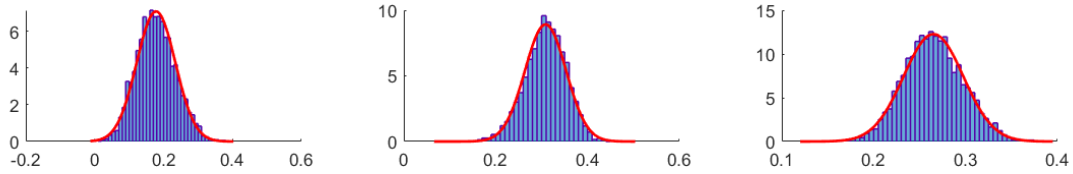
The expectation and variance of coverage fraction we have derived are based on the assumption that the flake center points $(\phi_1, \phi_2, \ldots)$ are uniformly distributed. Therefore, when the flakes are no longer distributed uniformly, the distribution of $\mathcal{C}_k$, along with $\mathbb{E}[\mathcal{C}_k]$ and $\mathbb{V}\mathrm{ar}[\mathcal{C}_k]$, changes. Based on this property, we provide a statistical testing method to detect the nonuniform patterns based on the k-layer coverage fraction $\mathcal{C}_k$ evaluated from the real image. In the inkjet printing process, some combination of unfavorable process conditions may lead to a “coffee-ring" effect: most of the flakes will be clustered around the boundary of the printed region. Such an effect is very detrimental to the sensor performance, and we want to detect such a condition when it happens. Furthermore, due to the overlapping of flakes, it is generally very difficult to measure the center locations of the flakes from images. Thus, the flake center locations are not available to assess the flake distribution. Instead, $\mathcal{C}_k$ is generally obtainable from images. Therefore, the proposed method

of testing the flake distribution based on $\mathcal{C}_k$ is very useful.

The hypothesis testing is based on the fact that $\mathcal{C}_k$ approximately follows the normal distribution. Billingsley (1995) showed that the summation of the associated Bernoulli random variables in a sequence follows the normal distribution asymptotically as the number of the random variables goes to infinity when the random variables far apart from each other in the sequence are nearly independent. This condition corresponds to the coverage fractions: $\mathcal{C}_k = \int_{\mathbf{z}\in\mathbb{S}_F} T_k(\mathbf{z}) \,/\, |\mathbb{S}_F|$ is the integration of the Bernoulli random variable $T_k(\mathbf{z})$ that has a nonzero correlation with $T_k(\mathbf{w})$ if $\mathbf{z}$ and $\mathbf{w}$ are close but has a nearly zero correlation when $\mathbf{z}$ and $\mathbf{w}$ are distant due to the limitations in the flake sizes. In particular, as the correlation between two points lessens, the variances of the coverage fractions diminish, and the distribution of the coverage fractions gets closer to the normal distribution.

When the variance of $\mathcal{C}_k$ is large, however, its distribution may be discrepant from the normal. According to the previous analysis, we know larger flakes induce a larger variance of coverage fractions. Because flakes must be deposited through the inkjet printer nozzle, there exists a limitation in the size of printable flakes, relative to the size of the droplet. As a result, we can examine the distribution of the $\mathcal{C}_k$ in the worst-case scenario, when $\mathcal{C}_k$ has the largest variance. The largest printed flakes given nozzle sizes are investigated in He and Derby (2017). According to their study, we simulated flakes with $\mathbb{E}[r] = 17.95$, $\mathrm{Sd}[r] = 11.6$, $\mathbb{E}[N] = 122.1$, and $R_F = 180$ for the worst-case scenario with 1,000 repetitions where Sd stands for the standard deviation. The histogram of the zero- and single-layer coverage fractions simulated based on the given parameters are Figure 3.4.1, and they are well fitted

to the normal distribution. As the variances of the coverage fractions reduce, the coverage fractions fit better to the normal distribution.



(a) Histogram of C_0 (b) Histogram of C_1 (c) Histogram of C_2

Figure 3.4.1: The histograms of $\mathcal{C}_0$ (left), $\mathcal{C}_1$ (center), and $\mathcal{C}_2$ (right) based on the simulated data suggest that they fit well to the normal distribution.

With the normal approximation of the distribution of $\mathcal{C}_k$, a χ^2 test can be established. A vector of the multiple layer coverage fractions from zero- to $(m$-1)-overlapping flakes is defined by $\boldsymbol{\mathcal{C}}_m = (\mathcal{C}_0, \mathcal{C}_1, \ldots, \mathcal{C}_{m-1})^T$ with its mean $\mathbb{E}[\boldsymbol{\mathcal{C}}_m] = (\mathbb{E}[\mathcal{C}_0], \mathbb{E}[\mathcal{C}_1], \ldots, \mathbb{E}[\mathcal{C}_{m-1}])^T$, which are calculated through (3.4). The covariance matrix, $\Sigma_{\boldsymbol{\mathcal{C}}_m}$ where its kth diagonal component is $\mathbb{V}\mathrm{ar}[\mathcal{C}_k]$ and its (k,h) element is $\mathbb{C}\mathrm{ov}[\mathcal{C}_k, \mathcal{C}_h]$, is calculated through (3.8) and (3.10). Then, the proposed statistic Q_m is as follows.

$$Q_m = (\boldsymbol{\mathcal{C}}_m - \mathbb{E}[\boldsymbol{\mathcal{C}}_m])^T \Sigma_{\boldsymbol{\mathcal{C}}_m}^{-1} (\boldsymbol{\mathcal{C}}_m - \mathbb{E}[\boldsymbol{\mathcal{C}}_m]) \sim \chi_m^2 \tag{3.19}$$

Q_m follows the chi-square distribution with the degree of freedom m. m can be selected based on how many layers are distinguishable in the image. Because many electronics require a low number of overlapping layers, m may not need to be large. We would like to mention that χ_m^2 distribution approximates Q_m very well. In the aforementioned worst-case, the false alarm probability of the test from the

simulated data with 5,000 iterations is 0.049 when we use $\chi^2_{1-0.05,m}$ as the critical value of the test. The actual false alarm probability is very close to the nominal value of 0.05.

The statistical testing can be performed as follows. First, $\mathbb{E}[\boldsymbol{\mathcal{C}}_m]$ and $\Sigma_{\boldsymbol{\mathcal{C}}_m}$ are calculated from (3.4), (3.8), and (3.10) with the printing parameters. Based on the image resolution, m, the number of distinguished layers from the image is determined. The upper control limit of the testing is set as $\chi^2_{1-\alpha,m}$ where $1-\alpha$ is the specified confidence level. Then, the coverage fractions $\mathcal{C}_k, k = 0, \ldots, m-1$, are measured from the printed pattern. When Q_m is larger than the upper control limit, the printed pattern is identified as nonuniform. Once Q_m is larger than the control limit, we know that the flakes are not uniformly distributed (with a probability of 1-(*p-value*)). This is because if the flakes were uniformly distributed, the mean and variance of the coverage fractions would be the same as the theoretical ones, and Q_m would not be significantly large under the chi-square distribution.

Figure 3.4.2 illustrates the implementation procedure to facilitate utilizing the results in this work. In Figure 3.4.2, the decisions that users need to make are represented by the boxes with dashed-line boundaries, and the other processes are represented by the boxes with solid-line boundaries. The main outcome of our work consists of two parts: the mean and covariance of the flake coverage fractions and the uniformity statistical testing. The process of obtaining the outcomes of the flake coverage model is represented by the white boxes. The uniformity statistical testing process is represented in the grey boxes.

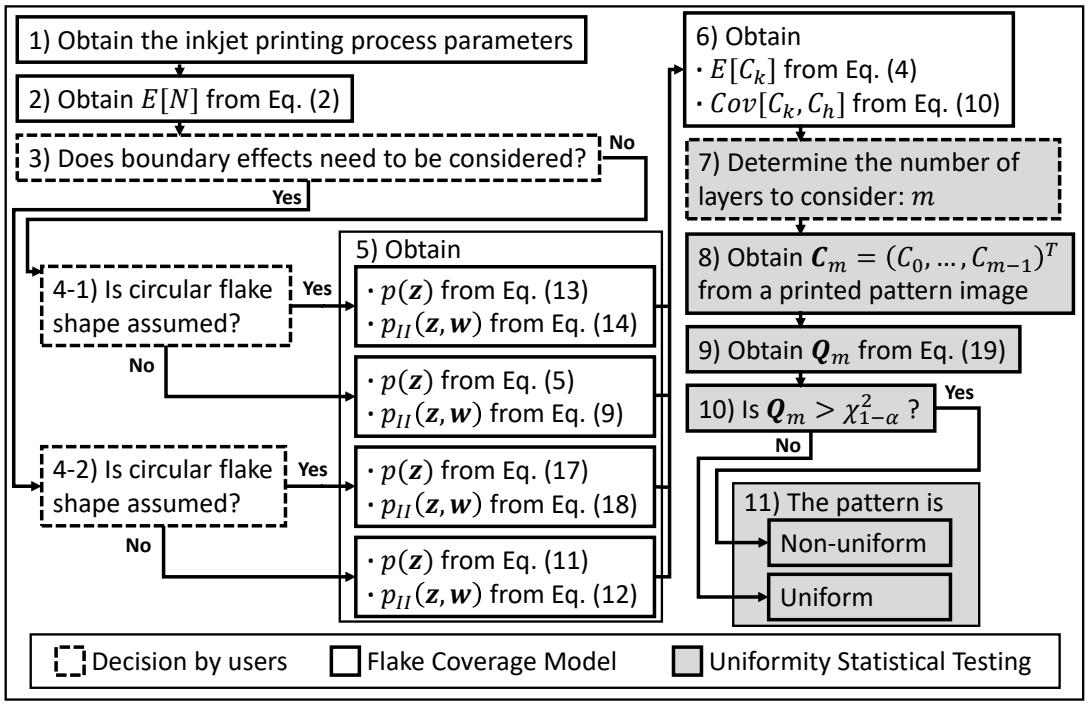


Figure 3.4.2: Summary of the implementation procedure

3.5 Numerical Study

In this section, the robustness of our proposed method is validated with different shapes of flakes (circle, ellipse, and rectangle) in various parameter settings. The parameter settings for the cases are shown in Table 3.5.1. For each case, we change one set of parameters (baseline case) to a higher level (Settings 1) or a lower level (Settings 2) while the other parameters are fixed.

Four parameters are selected to vary: (1) size of the flakes: mean and standard deviation of flake sizes ($\mathbb{E}[|F|]$ and $\mathrm{Sd}[|F|]$) are varied. The baseline is where $\mathbb{E}[|F|] = 98.17$ and $\mathrm{Sd}[|F|] = 117.87$ (which is when $\mathbb{E}[r] = 5$ and $\mathrm{Sd}[r] = 2.5$ of

Table 3.5.1: Summary of the parameters used in the numerical study

Case	Parameter	Parameter Values			Table Index	
		Baseline	Setting 1(H)	Setting 2(L)	No Boundary	Boundary
(i)	$E[\lvert F\rvert]$	98.17	392.70	3.93	Table A.3	Table A.8
	$\mathrm{Sd}[\lvert F\rvert]$	117.87	471.47	4.71		
(ii)	$E[N]\,p$	1	1.5	0.5	Table A.4	Table A.9
(iii)	R_F	150	200	100	Table A.5	Table A.10
(iv)	$Ratio(r)$	0.5	0.7	0.3	Table A.6	Table A.11

circular flakes). They are changed into a high level ($\mathbb{E}[|F|] = 392.70$ and $\mathrm{Sd}[|F|] = 471.47$ where $\mathbb{E}[r] = 10$ and $\mathrm{Sd}[r] = 5$ of circular flakes) and a low level ($\mathbb{E}[|F|] = 3.93$ and $\mathrm{Sd}[|F|] = 4.71$ where $\mathbb{E}[r] = 1$ and $\mathrm{Sd}[r] = 0.5$ of circular flakes). (2) Flakes mass concentration: $E[N]p$, indicating the flakes mass concentration, is increased to 1.5 and reduced to 0.5. (3) Radius of the flake space: we considered a circular flake space with radius of $R_F = 150$, which is changed to 200 and 100. (4) Ratio of radius: the shapes of elliptical and rectangular flakes have varied. $Ratio(r)$, the ratio between two axes of ellipse and rectangle (ratio between minor and major axes in an ellipse and ratio between width and length in a rectangle) is varied.

For each setting, we conducted the analysis both with and without the boundary effects. The outcomes from the simulation with three different flakes shapes (circle, ellipse, and rectangle) are presented along with the results calculated from the equations proposed in Section 3.3.4. The simulations are conducted with 1,000 iterations. Because of the page limit, the detailed results are presented as tables in Appendix 3.C. The table indexes can be found in Table 3.5.1. From the numerical study results, we can observe the following characteristics:

1. The analytical results obtained in this chapter (based on Equations (3.4) and

(3.8)) fit the simulation results very well. The expectations of the coverage fractions show little discrepancies among different shapes of flakes, and the outcomes based on our proposed equations are consistent with the simulated results. The flake shapes seldom affect the results. The standard deviations of the coverage fractions show larger differences than the expectations, but the discrepancies are still small. We can conclude that the expectation and standard deviation of the coverage fractions are robust to the flake shapes.

2. Without boundary effects, $\mathbb{E}[\mathcal{C}_0]$ and $\mathbb{E}[\mathcal{C}_1]$ are only determined by $E[N]p$. A larger mass concentration (indicated by large $E[N]p$) reduces the uncovered area. On the contrary, $\mathrm{Sd}[\mathcal{C}_0]$ and $\mathrm{Sd}[\mathcal{C}_1]$ are mainly affected by the relative size of the flakes to that of the printed pattern ($\mathbb{E}[|F_i|] / |\mathbb{S}_F|$) as shown in Figure 3.5.1.

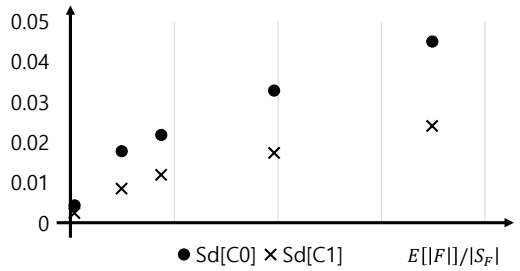


Figure 3.5.1: Changes in $\mathrm{Sd}[\mathcal{C}_k], k = 0, 1$

3. The impact of the boundary effects on $\mathcal{C}_0$ can be observed. $\mathbb{E}[\mathcal{C}_0]$s are smaller when the boundary effects are considered in comparison to those without boundary effects. Especially, larger flakes show a large difference of $\mathbb{E}[\mathcal{C}_0]$. With the boundary effects, larger flakes tend to gather inside the flake space,

leading to a reduction of $\mathbb{E}[\mathcal{C}_0]$. Inversely, a larger pattern size (larger radius of the printed pattern R_F) narrows the impacts of the boundary effects, meaning that a large printing area reduces the effects of the boundary. In general, as the flake sizes get smaller and as the size of the printed pattern R_F gets large, our proposed approximation method that considers the boundary effects becomes more accurate. Overall, $\mathbb{E}[\mathcal{C}_0]$ from the analytical equation (3.4) are smaller than those from the simulations. This is because we slightly overestimate the probability of a point being covered by a random flake.

3.6 Validation With Real Inkjet-Printed Flake Images

In this section, we show that the proposed model well describes the real flakes patterns produced by inkjet printing process, and the statistical testing scheme can detect nonuniform patterns. A flakes image is a surface topology image scanned by focused electron beams in the micrometer scale (also known as an SEM image). The original images are adopted from He and Derby (2017) and He (2016) with permission.

Six real images are presented in Figure 3.6.1. These are complete images of the dried droplets obtained from the inkjet printing processes. The physical parameters obtained from the printing process are summarized in Table 3.6.1 (He (2016)). Figure 3.6.1(a)-(c) show three uniform patterns, and Figure 3.6.1(d)-(f) show three nonuniform patterns that are widely known as the coffee-ring effect. When the coffee-ring appears, flakes are pushed forward and deposited near the boundary

of the droplet; therefore, white blanks form in the center of the dried pattern. The coffee-rings in Figure 3.6.1(d)-(e) are quite subtle to observe without careful attention; thus, we marked the center of the images to highlight the coffee-ring effects. In Figure 3.6.1(d)-(f), large portions of white blanks are clearly observed in the center of the dried droplet. Additional information on these images are presented in Appendix 3.D.1.

Table 3.6.1: Physical parameters

$\mathbb{E}[c_F]$	h_F	ρ_F	V_D	θ_c
0.5 mg/ml	1.0 nm	2200 mg/ml	0.77 nL	9.6 °

To conduct the statistical testing, the zero- and single-layer coverage fractions, $\mathcal{C}_0$ and $\mathcal{C}_1$, are extracted from the real images as follows. First, the brightness values of the pixels inside the grey-scale image are plotted as a histogram (Figure 3.6.2). From the histogram, we can decide the cut-off brightness dividing each layer coverage. We used a heuristic method to determine the cut-off thresholds in this work, and the procedure is presented in Appendix 3.D.2. The proportions of zero- and single-layer coverage are obtained accordingly.

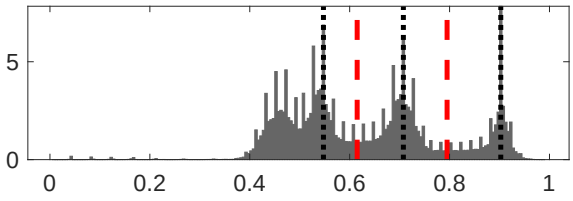
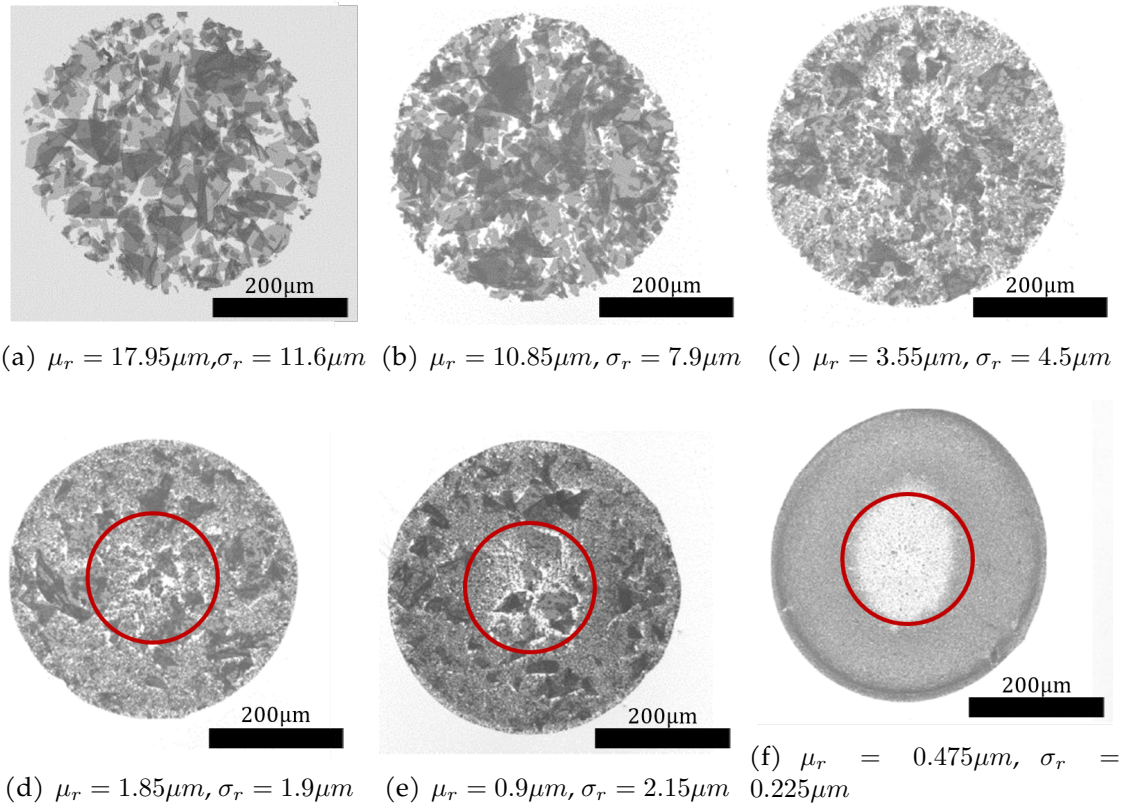


Figure 3.6.2: Histogram of the brightness of every pixel in the real inkjet-printed pattern image shown in Figure 3.6.1(a). Cut-off values for zero- and single-layer coverage are presented as vertical dashed lines.



(a) $\mu_r = 17.95\mu m$,$\sigma_r = 11.6\mu m$ (b) $\mu_r = 10.85\mu m$, $\sigma_r = 7.9\mu m$ (c) $\mu_r = 3.55\mu m$, $\sigma_r = 4.5\mu m$

(d) $\mu_r = 1.85\mu m$, $\sigma_r = 1.9\mu m$ (e) $\mu_r = 0.9\mu m$, $\sigma_r = 2.15\mu m$ (f) $\mu_r = 0.475\mu m$, $\sigma_r = 0.225\mu m$

Figure 3.6.1: Real images of the patterns printed by inkjet printer.

The extracted zero- and single-layer coverage fractions are presented in Table 3.6.2. In the table, $\mathbb{E}[r]$ and $\mathrm{Sd}[r]$ are the expectation and standard deviation of the radius of the flakes, which are provided in He and Derby (2017) and He (2016). The reduced space $\mathbb{S}_e$ is evaluated to deal with the boundary effects. $\mathcal{C}_0$ and $\mathcal{C}_1$ are the coverage fractions measured from the images. $\mathbb{E}[\mathcal{C}_k]$ and $\mathrm{Sd}[\mathcal{C}_k]$ are the expectation and standard deviation of the k-layer coverage fraction $\mathcal{C}_k$ calculated by Equations (3.4) and (3.8), and $\mathbb{C}\mathrm{ov}[\mathcal{C}_0, \mathcal{C}_1]$ is the covariance between $\mathcal{C}_0$ and $\mathcal{C}_1$ calculated by Equation (3.10). The chi-square statistics Q_2 calculated by Equation

Table 3.6.2: Case study outcome calculated by (3.4), (3.8), and (3.10) for the real images in Figure 3.6.1. For each printed pattern, statistical proportion test is conducted.

$\mathbb{E}[r]$	$\mathrm{Sd}[r]$	$\mathcal{C}_0$	$\mathcal{C}_1$	$\mathbb{E}[\mathcal{C}_0]$	$\mathbb{E}[\mathcal{C}_1]$	$\mathrm{Sd}[\mathcal{C}_0]$	$\mathrm{Sd}[\mathcal{C}_1]$	$\mathbb{C}\mathrm{ov}[\mathcal{C}_0, \mathcal{C}_1]$	Q_2	p-value
17.95	11.6	0.0977	0.2268	0.0773	0.1979	0.0542	0.0732	0.0028	0.18	0.9161
10.85	7.9	0.1039	0.2739	0.1041	0.2355	0.0414	0.0480	0.0015	1.39	0.499
3.55	4.5	0.1962	0.352	0.1081	0.2405	0.0460	0.0528	0.0021	4.49	0.1058
1.85	1.9	0.1451	0.353	0.1556	0.2895	0.0144	0.0122	0.0001	91.92	1.1×10^{-20}
0.9	2.15	0.125	0.659	0.1002	0.2305	0.0590	0.0712	0.0037	152.52	7.6×10^{-34}
0.475	0.225	0.217	0.5922	0.1768	0.3063	0.0010	0.0008	0	132,603	0

(3.19) are presented, followed by their p-values.

Based on the statistical testing, the images in Figure 3.6.1(a)-(c) are identified uniformly distributed with p-values larger than 0.1 while the images in Figure 3.6.1(d)-(f) are considered as nonuniform patterns with p-values less than 0.01. The case studies show that our proposed method can effectively catch the subtle nonuniformities of the pattern images.

The validation based on the real images suggests our proposed statistical model and the testing methods fit well with real data and can identify the nonuniform flakes dispersion pattern.

We would like to mention that the physical size of the images needs to be large enough to include a sufficient number of printed flakes. If an image is too small and includes only a limited number of flakes, the variance of the coverage fractions would be significantly large, and it will lead to a lower ability to identify the nonuniform patterns (i.e., a lower hypothesis testing power).

3.7 Conclusion

In this work, we constructed a statistical model to describe the k-layer coverage of randomly distributed two-dimensional materials. A series of results are obtained to provide not only the expectation but also the variance of coverage area. Compared with existing results, the proposed model considers the coverage with multiple overlapping layers and also provides the variance of the coverage area. To make our model more useful, boundary effects are also studied. With the boundary effects, a method to accurately evaluate the expectation and variance of k-layer coverage within a certain region is proposed. In addition, an approximated statistical testing approach is developed in this work to detect abnormal coverage patterns. The case studies based on the simulated data and real flake images obtained from the inkjet printing process show the accuracy and effectiveness of the proposed model and analysis methods. We expect the proposed model to be used to predict the functionality of the printed electronics and control the variability. The proposed statistical model may also be used in different applications. For example, our model can be used to describe the overlapping random coverage for the random wireless sensor network. Our proposed model can provide the prediction and quantification of variation where the detection ranges of the sensors are random.

We would like to mention that sometimes SEM image data might be expensive so that we may not be able to use these images for real-time quality control. However, even though the SEM image data are expensive, SEM images (or similar imaging data) are commonly available in modern microelectronic manufacturing processes, and these images can certainly be used for offline inspection and root cause analysis.

There are some interesting possible future directions. To calculate the variance of the coverage fractions in the integration, the circular shape of flakes was assumed in this work. However, if the flake shapes are far from circles, the result may not be sufficiently accurate. We can improve the accuracy of this model by using the observed flake shapes from the flake images directly onto the flake size and $p_{\amalg}(\mathbf{z}, \mathbf{w})$ estimations. For a more accurate estimate of the expected flake size, practitioners may use the flake images of patterns printed with diluted inks so that every flake is distinct and store the set of shapes of the flakes. Then, the mean flake size can be estimated based on the exact shape of the flakes, and accurate $p_{\amalg}(\mathbf{z}, \mathbf{w})$ may be estimated based on the observed shapes of flakes. Another possible direction of studying $p_{\amalg}(\mathbf{z}, \mathbf{w})$ would be the field of stochastic geometry, which has studied many aspects of the random geometry behaviors, including the boolean model. We plan to study these issues and report them in the future.

APPENDIX

3.A Mathematical Definitions

3.A.1 Mathematical Definition of Random Flakes

Each flake F_i is defined by a combination of its center point/location ϕ_i and its shape Ξ_i.

$$F_i = \phi_i + \Xi_i = \{\phi_i + \mathrm{s} | \mathrm{s} \in \Xi_i\} \tag{3.20}$$

The locations $\{\phi_1, \phi_2, \ldots\}$ follow a point process where every point is uniformly distributed in $\mathbb{S}_F \subset \mathbb{R}^2$ where $\mathbb{S}_F$ is the *flake space* or *printed space* within which flakes are deposited. The shape of the ith flake, denoted by Ξ_i, is defined by an independent and identically distributed (i.i.d.) random compact set $\Xi_i \subset \mathbb{R}^2$ whose center is on the origin. $|\Xi_i| = |F_i|$ is the size of the flake.

A random compact set is a random variable that has a compact set as a value. The definition of a random compact set can be found in Molchanov (2005, 2006): A random compact set is a random closed set X with almost surely compact values. A random closed set is a map $X : \Omega \to \mathcal{F}$ from a probability space (Ω, Σ, P) to the family $\mathcal{F}$ of closed subsets of a locally compact separable Hausdorff space E if $\{X \cap K \neq \emptyset\} \in \Sigma$ for every K from the family $\mathcal{K}$ of compact subsets of E.

3.A.2 Mathematical Definition of $flip\left(\cdot,\cdot\right)$

$flip\left(F_i,\mathbf{z}\right)$ is defined by $\mathbf{z}+\text{Ref}\left(\Xi_i\right)$ where $\text{Ref}\left(\Xi\right)=\{(x,y)|(-x,-y)\in\Xi\}$ is a point reflection function that returns the point reflection of the given geometry.

3.A.3 Mathematical Definition of $\mathbb{S}_{\phi_i}$

The restricted space in which the center point ϕ_i of flake F_i can be located, denoted by $\mathbb{S}_{\phi_i}$, is defined by $\mathbb{S}_{\phi_i}\coloneqq\{c|c+\Xi_i\subset\mathbb{S}_F,c\in\mathbb{S}_F\}\subset\mathbb{S}_F$.

3.B Detailed Derivation of the Main Results

3.B.1 Expectation of k-Layer Coverage Fraction

A detailed derivation of the expectation of the k-layer coverage fraction $\mathbb{E}\left[\mathcal{C}_k\right]$ is described. First, we show $\mathbb{E}\left[\mathcal{C}_k\right]=\mathbb{E}\left[T_k\left(\mathbf{z}\right)\right]$ based on that any point $\mathbf{z}$ in $\mathbb{S}_F$ is covered by a random flake with the same probability.

$$\mathbb{E}\left[\mathcal{C}_k\right]=\mathbb{E}\left[\frac{\int_{\mathbf{z}\in\mathbb{S}_F}T_k\left(\mathbf{z}\right)d\mathbf{z}}{\int_{\mathbf{z}\in\mathbb{S}_F}d\mathbf{z}}\right]=\frac{\int_{\mathbf{z}\in\mathbb{S}_F}\mathbb{E}\left[T_k\left(\mathbf{z}\right)\right]d\mathbf{z}}{|\mathbb{S}_F|} \tag{3.21}$$

$$=\mathbb{E}\left[T_k\left(\mathbf{z}\right)\right] \tag{3.22}$$

where $T_k\left(\mathbf{z}\right)$ is a random variable that is one if point $\mathbf{z}$ is covered by k layers of flakes, or zero, otherwise. By the law of total expectation, $\mathbb{E}\left[\mathcal{C}_k\right]=\mathbb{E}\left[\mathbb{E}\left[\mathcal{C}_k|N\right]\right]$. We

first derive $\mathbb{E}[\mathcal{C}_k|N]$.

$$\mathbb{E}[\mathcal{C}_k|N] = \mathbb{E}[T_k(\mathbf{z})|N] = \mathbb{E}_{\{p_i\}}[\mathbb{E}[T_k(\mathbf{z})|N, \{F_i\}]] \tag{3.23}$$

$$= \mathbb{E}_{\{p_i\}}[\mathbb{P}[T_k(\mathbf{z}) = 1|N, \{F_i\}]] \tag{3.24}$$

$$= \mathbb{E}_{\{p_i\}}\left[\sum_{\sum_i x_i = k} \prod_{i=1}^{N} {p_i}^{x_i} (1 - p_i)^{1 - x_i}\right] \tag{3.25}$$

$$= \sum_{\sum_i x_i = k} \prod_{i=1}^{N} \mathbb{E}[p_i]^{x_i} (1 - \mathbb{E}[p_i])^{1 - x_i} \tag{3.26}$$

$$= \binom{N}{k} p^k (1 - p)^{N-k} \tag{3.27}$$

where $p = \mathbb{E}[p_i]$. $\mathbb{E}[T_k(\mathbf{z})] = \mathbb{P}[T_k(\mathbf{z})]$ because $T_k(\mathbf{z})$ is a Bernoulli random variable conditioned on $\mathbf{z}$. The main idea is $p_i = |F_i| / |\mathbb{S}_F|$, the probability that a point is covered by the flake F_i, is an independent random variable because $|F_i|$ is an independent random variable. When the number of printed flakes N follows the Poisson distribution with mean $\mathbb{E}[N]$, $\mathbb{E}[\mathcal{C}_k]$ can be written as follows.

$$\mathbb{E}[\mathcal{C}_k] = \mathbb{E}_N[\mathbb{E}[\mathcal{C}_k|N]] = \sum_{N=k}^{\infty} \frac{\mathbb{E}[N]^N \exp(-\mathbb{E}[N])}{N!} \frac{N!}{k!(N-k)!} p^k (1 - p)^{N-k} \tag{3.28}$$

$$= \frac{(\mathbb{E}[N] p)^k \exp(-\mathbb{E}[N])}{k!} \sum_{N=k}^{\infty} \frac{\{\mathbb{E}[N](1 - p)\}^{N-k}}{(N-k)!} \tag{3.29}$$

$$= \frac{\exp(-\mathbb{E}[N] p)(\mathbb{E}[N] p)^k}{k!} \tag{3.30}$$

3.B.2 Variance of k-layer Coverage Fraction

$\mathbb{V}\mathrm{ar}\left[\mathcal{C}_k\right]$ can be represented in terms of $\mathbb{C}\mathrm{ov}\left[T_k\left(\mathbf{z}\right), T_k\left(\mathbf{w}\right)\right]$.

$$\mathbb{V}\mathrm{ar}\left[\mathcal{C}_k\right] = \mathbb{V}\mathrm{ar}\left[\frac{\int_{\mathbf{z}\in\mathbb{S}_F} T_k\left(\mathbf{z}\right) d\,\mathbf{z}}{\int_{\mathbf{z}\in\mathbb{S}_F} d\,\mathbf{z}}\right] \tag{3.31}$$

$$= \frac{1}{\left|\mathbb{S}_F\right|^2}\left\{\mathbb{E}\left[\left(\int_{\mathbf{z}\in\mathbb{S}_F} T_k\left(\mathbf{z}\right) d\,\mathbf{z}\right)^2\right] - \mathbb{E}\left[\int_{\mathbf{z}\in\mathbb{S}_F} T_k\left(\mathbf{z}\right) d\,\mathbf{z}\right]^2\right\} \tag{3.32}$$

$$= \frac{1}{\left|\mathbb{S}_F\right|^2}\left\{\left\{\iint\limits_{\mathbf{z},\mathbf{w}\in\mathbb{S}_F} \mathbb{E}\left[T_k\left(\mathbf{z}\right) T_k\left(\mathbf{w}\right)\right] d\,\mathbf{w} d\,\mathbf{z}\right\}^2 - \iint\limits_{\mathbf{z},\mathbf{w}\in\mathbb{S}_F} \mathbb{E}\left[T_k\left(\mathbf{z}\right)\right]\mathbb{E}\left[T_k\left(\mathbf{w}\right)\right] d\,\mathbf{w} d\,\mathbf{z}\right\} \tag{3.33}$$

$$= \frac{1}{\left|\mathbb{S}_F\right|^2}\int_{\mathbf{z}\in\mathbb{S}_F}\int_{\mathbf{w}\in\mathbb{S}_F} \mathbb{C}\mathrm{ov}\left[T_k\left(\mathbf{z}\right), T_k\left(\mathbf{w}\right)\right] d\,\mathbf{w} d\,\mathbf{z} \tag{3.34}$$

Further, by definition, (3.34) is the same as

$$\frac{\mathbb{V}\mathrm{ar}\left[T_k\left(\mathbf{z}\right)\right]}{\left|\mathbb{S}_F\right|^2}\int_{\mathbf{z}\in\mathbb{S}_F}\int_{\mathbf{w}\in\mathbb{S}_F} \mathbb{C}\mathrm{orr}\left[T_k\left(\mathbf{z}\right), T_k\left(\mathbf{w}\right)\right] d\,\mathbf{w} d\,\mathbf{z} \tag{3.35}$$

For $\mathbb{C}\mathrm{ov}\left[T_k\left(\mathbf{z}\right), T_k\left(\mathbf{w}\right)\right] = \mathbb{E}\left[T_k\left(\mathbf{z}\right) T_k\left(\mathbf{w}\right)\right] - \mathbb{E}\left[T_k\left(\mathbf{z}\right)\right]\mathbb{E}\left[T_k\left(\mathbf{w}\right)\right]$, we need $\mathbb{E}\left[T_k\left(\mathbf{z}\right) T_k\left(\mathbf{w}\right)\right]$, which can be obtained based on the law of total expectation: $\mathbb{E}\left[T_k\left(\mathbf{z}\right) T_k\left(\mathbf{w}\right)\right] = \mathbb{E}\left[\mathbb{E}\left[T_k\left(\mathbf{z}\right) T_k\left(\mathbf{w}\right)|N\right]\right]$. Because both $T_k\left(\mathbf{z}\right)$ and $T_k\left(\mathbf{w}\right)$ are binary variables that have zero or one as values, the event of interest is $\{T_k\left(\mathbf{z}\right) = 1 \cap T_k\left(\mathbf{w}\right) = 1\}$ and $\mathbb{E}\left[T_k\left(\mathbf{z}\right) T_k\left(\mathbf{w}\right)|N\right] = \mathbb{P}\left[T_k\left(\mathbf{z}\right) T_k\left(\mathbf{w}\right)|N\right]$. This event can be divided into $k+1$ mutually exclusive collectively exhaustive subevents: both points $\mathbf{z}$ and $\mathbf{w}$ are covered by the l same flakes and by $(k-l)$ layers with different flakes, $(l = 0, 1, \ldots, k)$ where, for each flake, points $\mathbf{z}$ and $\mathbf{w}$ are both covered by the same flake with probability $p_{\mathit{II}\,i}(\mathbf{z},\mathbf{w}) = \mathbb{P}\left[\mathbf{z}\in F_i \cap \mathbf{w}\in F_i\right]$ and $p_{\mathit{II}}(\mathbf{z},\mathbf{w}) = \mathbb{E}\left[p_{\mathit{II}\,i}(\mathbf{z},\mathbf{w})\right]$.

$\mathbb{E}\left[T_k\left(\mathbf{z}\right)T_k\left(\mathbf{w}\right)|N\right]$ can be written as follows.

$$\mathbb{E}\left[T_k\left(\mathbf{z}\right)T_k\left(\mathbf{w}\right)|N\right] = \mathbb{E}\left[\mathbb{E}\left[T_k\left(\mathbf{z}\right)T_k\left(\mathbf{w}\right)|N,\{F_i\}\right]|N\right] \tag{3.36}$$

$$= \mathbb{E}\left[\mathbb{P}\left[T_k\left(\mathbf{z}\right)=1,\,T_k\left(\mathbf{w}\right)=1|N,\{F_i\}\right]|N\right] \tag{3.37}$$

$$= \mathbb{E}\left[\prod_{|\{i\}|=l} p_{I\!I\,i}(\mathbf{z},\mathbf{w}) \prod_{\substack{|\{j\}|=k-l\\ j\neq i}} p_{I\,j}(\mathbf{z},\mathbf{w}) \prod_{\substack{|\{h\}|=k-l\\ h\neq i\\ h\neq j}} p_{I\,h}(\mathbf{z},\mathbf{w}) \prod_{\substack{v\neq i\\ v\neq j\\ v\neq h}} p_{\emptyset\,v}(\mathbf{z},\mathbf{w})\right] \tag{3.38}$$

$$= \sum_{l=0}^{k} \frac{N!}{l!\left(N-l\right)!} p_{I\!I}(\mathbf{z},\mathbf{w})^{l} \frac{\left(N-l\right)!}{\left\{N-l-2\left(k-l\right)\right\}!\left\{\left(k-l\right)!\right\}^2} p_{I}(\mathbf{z},\mathbf{w})^{2(k-l)}\, p_{\emptyset}(\mathbf{z},\mathbf{w})^{N-l-2(k-l)} \tag{3.39}$$

$$= \sum_{l=0}^{k} \frac{N!}{l!(k-l)!^2\left\{N-2k+l\right\}!} p_{I\!I}(\mathbf{z},\mathbf{w})^{l}\left\{p-p_{I\!I}(\mathbf{z},\mathbf{w})\right\}^{2(k-l)}\left\{1-2p+p_{I\!I}(\mathbf{z},\mathbf{w})\right\}^{N-2k+l} \tag{3.40}$$

where $p_{I\!I\,i}(\mathbf{z},\mathbf{w})$ is the expected probability that points $\mathbf{z}$ and $\mathbf{w}$ are both covered by the flake F_i, and $p_{I\,i}(\mathbf{z},\mathbf{w}) = p - p_{I\!I\,i}(\mathbf{z},\mathbf{w})$ and $p_{\emptyset\,i}(\mathbf{z},\mathbf{w}) = 1-2p+p_{I\!I\,i}(\mathbf{z},\mathbf{w})$ are the probabilities that one of the points is covered and that none of them is covered by the flake F_i, respectively, and those without subscript i are the expectations of them. (e.g., $p_{I\!I}(\mathbf{z},\mathbf{w}) = \mathbb{E}\left[p_{I\!I\,i}(\mathbf{z},\mathbf{w})\right]$).

Given that N is a Poisson random variable, $\mathbb{E}\left[T_k\left(\mathbf{z}\right)T_k\left(\mathbf{w}\right)\right]$ can be written as follows.

$$\mathbb{E}\left[T_k\left(\mathbf{z}\right)T_k\left(\mathbf{w}\right)\right] = \mathbb{E}\left[\mathbb{E}\left[T_k\left(\mathbf{z}\right)T_k\left(\mathbf{w}\right)|N\right]\right] \tag{3.41}$$

$$= \sum_{N=k}^{\infty} \frac{\mathbb{E}\left[N\right]^N \exp\left(-\mathbb{E}\left[N\right]\right)}{N!}\,\mathbb{P}\left[T_k\left(\mathbf{z}\right)=1,\,T_k\left(\mathbf{w}\right)=1|N\right] \tag{3.42}$$

$$= \exp\left(-\mathbb{E}\left[N\right]\right) \sum_{l=0}^{k} \sum_{N=k}^{\infty} \frac{\mathbb{E}\left[N\right]^{N}}{l!(k-l)!^{2}\left\{N-2k+l\right\}!} p_{\text{II}}(\mathbf{z},\mathbf{w})^{l} \left\{p - p_{\text{II}}(\mathbf{z},\mathbf{w})\right\}^{2(k-l)} \times \left\{1 - 2p + p_{\text{II}}(\mathbf{z},\mathbf{w})\right\}^{N-2k+l} \tag{3.43}$$

$$= \exp\left(-\mathbb{E}\left[N\right]\right) \sum_{l=0}^{k} \frac{\mathbb{E}\left[N\right]^{2k-l}}{l!(k-l)!^{2}} p_{\text{II}}(\mathbf{z},\mathbf{w})^{l} \left\{p - p_{\text{II}}(\mathbf{z},\mathbf{w})\right\}^{2(k-l)} \times \sum_{N=k}^{\infty} \frac{\left\{\mathbb{E}\left[N\right]\left(1 - 2p + p_{\text{II}}(\mathbf{z},\mathbf{w})\right)\right\}^{N-2k+l}}{\left\{N-2k+l\right\}!} \tag{3.44}$$

$$= \sum_{l=0}^{k} \frac{\mathbb{E}\left[N\right]^{2k-l} p_{\text{II}}(\mathbf{z},\mathbf{w})^{l} \left\{p - p_{\text{II}}(\mathbf{z},\mathbf{w})\right\}^{2(k-l)}}{l!\left\{(k-l)!\right\}^{2}} \exp\left\{-\mathbb{E}\left[N\right]\left(2p - p_{\text{II}}(\mathbf{z},\mathbf{w})\right)\right\} \tag{3.45}$$

$\mathbb{V}\mathrm{ar}\left[\mathcal{C}_k\right]$ can be obtained by $\mathbb{V}\mathrm{ar}\left[\mathcal{C}_k\right] = \int_{\mathbf{z}\in\mathbb{S}_F} \int_{\mathbf{w}\in\mathbb{S}_F} \mathbb{E}\left[T_k\left(\mathbf{z}\right) T_k\left(\mathbf{w}\right)\right] - \mathbb{E}\left[T_k\left(\mathbf{z}\right)\right]^2 d\mathbf{w}d\mathbf{z}/\left|\mathbb{S}_F\right|^2$.

$\mathbb{C}\mathrm{ov}\left[\mathcal{C}_k, \mathcal{C}_h\right]$ can be obtained in a similar manner through $\int_{\mathbf{z}\in\mathbb{S}_F} \int_{\mathbf{w}\in\mathbb{S}_F} \mathbb{E}\left[T_k\left(\mathbf{z}\right) T_h\left(\mathbf{w}\right)\right] - \mathbb{E}\left[T_k\left(\mathbf{z}\right)\right] \mathbb{E}\left[T_h\left(\mathbf{w}\right)\right] d\mathbf{w}d\mathbf{z}/\left|\mathbb{S}_F\right|^2$. Now, the event of interest is $\left\{T_k\left(\mathbf{z}\right) = 1 \cap T_h\left(\mathbf{w}\right) = 1\right\}$, which also can be divided into $\left(\min\left(k,h\right)+1\right)$ mutually exclusive collectively exhaustive subevents: both points $\mathbf{z}$ and $\mathbf{w}$ are covered by the l same flakes while $\mathbf{z}$ is covered by $(k-l)$ and $\mathbf{w}$ by $(h-l)$ layers with different flakes, and $l = 0, 1, \ldots, \min\left(k,h\right)$. Following the same derivation process of $\mathbb{V}\mathrm{ar}\left[\mathcal{C}_k\right]$, $\mathbb{C}\mathrm{ov}\left[\mathcal{C}_k, \mathcal{C}_h\right]$ can be written as follows.

$$\frac{1}{\left|\mathbb{S}_F\right|^2} \int_{\mathbf{z}} \int_{\mathbf{w}} \sum_{l=0}^{\min(k,h)} \frac{\mathbb{E}\left[N\right]^{k+h-l} p_{\text{II}}(\mathbf{z},\mathbf{w})^{l} \left\{p - p_{\text{II}}(\mathbf{z},\mathbf{w})\right\}^{(k+h-2l)}}{l!\left\{(k-l)!\right\}^{2}} \times \exp\left[-\mathbb{E}\left[N\right]\left\{2p - p_{\text{II}}(\mathbf{z},\mathbf{w})\right\}\right] - \mathbb{E}\left[\mathcal{C}_k\right]\mathbb{E}\left[\mathcal{C}_h\right] d\mathbf{w}d\mathbf{z} \tag{3.46}$$

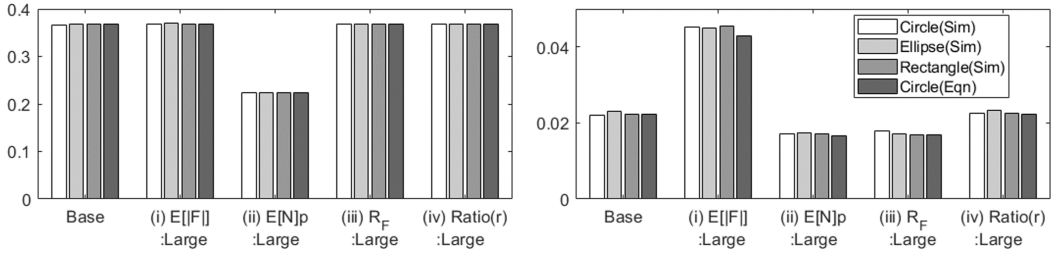
3.C Detailed Outcome of Numerical Study

In this section, we include the detailed outcome of the numerical studies. The parameter settings for the cases are shown in Table 3.C.1. For each case, we change one set of parameters (baseline case) to a higher level (Settings 1) or lower level (Settings 2) while the other parameters are fixed.

Table 3.C.1: Summary of parameters used in the numerical study

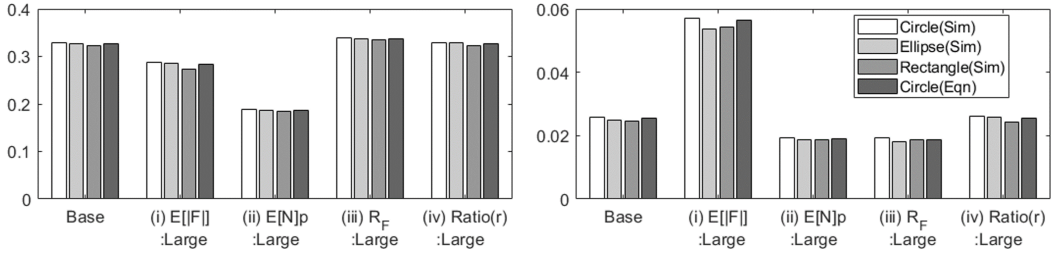
Case	Parameter	Parameter Values			Table Index	
		Baseline	Setting 1 (Large)	Setting 2 (Small)	No Boundary	Boundary
(i)	$E[\lvert F\rvert]$	98.17	392.70	3.93	Table 3.C.3	Table 3.C.8
	$\mathrm{Sd}[\lvert F\rvert]$	117.87	471.47	4.71		
(ii)	$E[N]\,p$	1	1.5	0.5	Table 3.C.4	Table 3.C.9
(iii)	R_F	150	200	100	Table 3.C.5	Table 3.C.10
(iv)	$Ratio(r)$	0.5	0.7	0.3	Table 3.C.6	Table 3.C.11

The comparison of the expectation and standard deviation of the zero- and single-layer coverage fractions ($\mathcal{C}_0$ and $\mathcal{C}_1$) between the baseline and the first setting (i.e., high setting) is presented in Figure 3.C.1 and 3.C.2, respectively.



(a) $\mathbb{E}[\mathcal{C}_0]$ without boundary effects

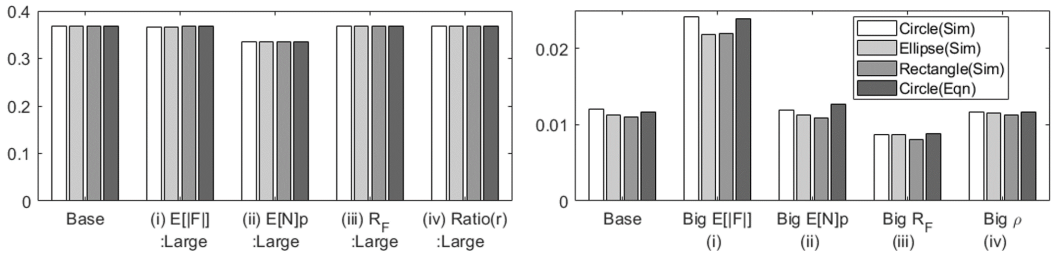
(b) $\mathrm{Sd}[\mathcal{C}_0]$ without boundary effects



(c) $\mathbb{E}[\mathcal{C}_0]$ with boundary effects

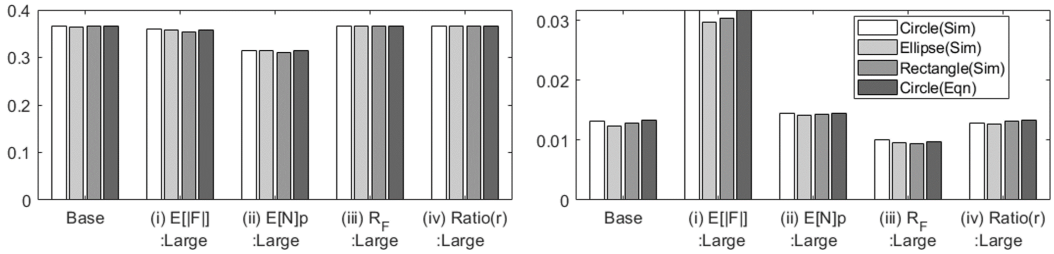
(d) $\mathrm{Sd}[\mathcal{C}_0]$ with boundary effects

Figure 3.C.1: Estimates regarding $\mathcal{C}_0$ obtained by the simulation and equations



(a) $\mathbb{E}[\mathcal{C}_1]$ without boundary effects

(b) $\mathrm{Sd}[\mathcal{C}_1]$ without boundary effects



(c) $\mathbb{E}[\mathcal{C}_1]$ with boundary effects

(d) $\mathrm{Sd}[\mathcal{C}_1]$ with boundary effects

Figure 3.C.2: Estimates regarding $\mathcal{C}_1$ obtained by the simulation and equations

Table 3.C.2: Baseline without boundary effects

Type	Shape	$\mathbb{E}[\mathcal{C}_0]$	$\mathbb{E}[\mathcal{C}_1]$	$\mathbb{E}[\mathcal{C}_2]$	$\mathrm{Sd}[\mathcal{C}_0]$	$\mathrm{Sd}[\mathcal{C}_1]$	$\mathrm{Sd}[\mathcal{C}_2]$	$\mathbb{C}\mathrm{ov}[\mathcal{C}_0, \mathcal{C}_1]$
Sim	Circle	0.3296	0.3661	0.2029	0.0258	0.0132	0.0140	$-7 \cdot 10^{-05}$
	Ellip.	0.3267	0.3648	0.2045	0.0250	0.0124	0.0137	$-6.1 \cdot 10^{-05}$
	Rect.	0.3234	0.3650	0.2062	0.0245	0.0129	0.0136	$-3.7 \cdot 10^{-05}$
Eq	Circle	0.3277	0.3656	0.2039	0.0256	0.0134	0.0142	$-6.6 \cdot 10^{-05}$

Table 3.C.3: Case(i): The expectation and standard deviation of 0-, 1-, and 2-layer coverage fractions with different flake sizes without boundary effects. Large $|F|$ is when $\mathbb{E}[|F|] = 392.70$ and $\mathrm{Sd}[|F|] = 471.47$ and small is when $\mathbb{E}[|F|] = 3.93$ and $\mathrm{Sd}[|F|] = 4.71$.

$\lvert F\rvert$	Type	Shape	$\mathbb{E}[\mathcal{C}_0]$	$\mathbb{E}[\mathcal{C}_1]$	$\mathbb{E}[\mathcal{C}_2]$	$\mathrm{Sd}[\mathcal{C}_0]$	$\mathrm{Sd}[\mathcal{C}_1]$	$\mathrm{Sd}[\mathcal{C}_2]$	$\mathbb{C}\mathrm{ov}[\mathcal{C}_0, \mathcal{C}_1]$
Large	Sim	Circle	0.369	0.367	0.1836	0.0451	0.0242	0.0244	-0.0004
		Ellip.	0.3694	0.3663	0.1841	0.045	0.0218	0.0242	-0.00036
		Rect.	0.368	0.3686	0.1838	0.0456	0.022	0.024	-0.00038
	Eq	Circle	0.3679	0.3679	0.1839	0.0428	0.0239	0.0241	0.000024
Small	Sim	Circle	0.3681	0.3678	0.1839	0.0045	0.0026	0.0026	$-4.40 \cdot 10^{-06}$
		Ellip.	0.3677	0.3679	0.184	0.0045	0.0025	0.0026	$-4.20 \cdot 10^{-06}$
		Rect.	0.368	0.3679	0.1838	0.0047	0.0025	0.0026	$-4.60 \cdot 10^{-06}$
	Eq	Circle	0.3679	0.3679	0.1839	0.0046	0.0024	0.0025	$-4.00 \cdot 10^{-06}$

Table 3.C.4: Case(ii): The expectation and standard deviation of 0-, 1-, and 2-layer coverage fractions with different $\mathbb{E}[N]\,p$ that corresponds to the mass concentration level without boundary effects. Large $\mathbb{E}[N]\,p$ is when $\mathbb{E}[N]\,p = 1.5$ and small is when $\mathbb{E}[N]\,p = 0.5$.

$\mathbb{E}[N]\,p$	Type	Shape	$\mathbb{E}[\mathcal{C}_0]$	$\mathbb{E}[\mathcal{C}_1]$	$\mathbb{E}[\mathcal{C}_2]$	$\mathrm{Sd}[\mathcal{C}_0]$	$\mathrm{Sd}[\mathcal{C}_1]$	$\mathrm{Sd}[\mathcal{C}_2]$	$\mathbb{C}\mathrm{ov}[\mathcal{C}_0, \mathcal{C}_1]$
Large	Sim	Circle	0.2232	0.3351	0.2510	0.0171	0.0120	0.0098	$4.97 \cdot 10^{-05}$
		Ellip.	0.2228	0.3346	0.2513	0.0174	0.0113	0.0098	$5.9 \cdot 10^{-05}$
		Rect.	0.2230	0.3345	0.2511	0.0170	0.0110	0.0096	$6.26 \cdot 10^{-05}$
	Eq	Circle	0.2231	0.3347	0.2510	0.0166	0.0127	0.0096	0.000086
Small	Sim	Circle	0.6061	0.3038	0.0758	0.0259	0.0172	0.0101	-0.0004
		Ellip.	0.6051	0.3039	0.0763	0.0248	0.0159	0.0102	-0.00034
		Rect.	0.6056	0.3037	0.0761	0.0259	0.0164	0.0102	-0.00038
	Eq	Circle	0.6065	0.3033	0.0758	0.0263	0.0166	0.0111	-0.00037

Table 3.C.5: Case(iii): The expectation and standard deviation of 0-, 1-, and 2-layer coverage fractions with different R_F, the radius of printed pattern without boundary effects. Large R_F is when $R_F = 200$ and small is when $R_F = 100$.

R_F	Type	Shape	$\mathbb{E}[\mathcal{C}_0]$	$\mathbb{E}[\mathcal{C}_1]$	$\mathbb{E}[\mathcal{C}_2]$	$\mathrm{Sd}[\mathcal{C}_0]$	$\mathrm{Sd}[\mathcal{C}_1]$	$\mathrm{Sd}[\mathcal{C}_2]$	$\mathbb{C}\mathrm{ov}[\mathcal{C}_0, \mathcal{C}_1]$
Large	Sim	Circle	0.3676	0.3679	0.1840	0.0179	0.0086	0.0095	$-6.6 \cdot 10^{-05}$
		Ellip.	0.3681	0.3677	0.1840	0.0172	0.0087	0.0092	$-5.6 \cdot 10^{-05}$
		Rect.	0.3687	0.3679	0.1836	0.0169	0.0081	0.0088	$-5.1 \cdot 10^{-05}$
	Eq	Circle	0.3679	0.3679	0.1839	0.0170	0.0088	0.0093	$-3.5 \cdot 10^{-05}$
Small	Sim	Circle	0.3681	0.3688	0.1836	0.0330	0.0175	0.0180	-0.00022
		Ellip.	0.3687	0.3676	0.1840	0.0337	0.0170	0.0173	-0.00024
		Rect.	0.3667	0.3683	0.1846	0.0345	0.0162	0.0188	-0.00018
	Eq	Circle	0.3679	0.3679	0.1839	0.0328	0.0176	0.0184	$-5.1 \cdot 10^{-05}$

Table 3.C.6: Case(iv): The expectation and standard deviation of 0-, 1-, and 2-layer coverage fractions with different $Ratio(r)$, the ratio between two axis of ellipse and rectangle without boundary effects. Large $Ratio(r)$ is when $Ratio(r) = 0.7$ and small is when $Ratio(r) = 0.3$.

$Ratio(r)$	Type	Shape	$\mathbb{E}[\mathcal{C}_0]$	$\mathbb{E}[\mathcal{C}_1]$	$\mathbb{E}[\mathcal{C}_2]$	$\mathrm{Sd}[\mathcal{C}_0]$	$\mathrm{Sd}[\mathcal{C}_1]$	$\mathrm{Sd}[\mathcal{C}_2]$	$\mathbb{C}\mathrm{ov}[\mathcal{C}_0, \mathcal{C}_1]$
Large	Sim	Circle	0.3679	0.3676	0.1840	0.0226	0.0117	0.0120	$-9.6 \cdot 10^{-05}$
		Ellip.	0.3681	0.3679	0.1837	0.0233	0.0115	0.0128	$-9.1 \cdot 10^{-05}$
		Rect.	0.3684	0.3681	0.1837	0.0226	0.0113	0.0119	-0.00011
	Eq	Circle	0.3679	0.3679	0.1839	0.0224	0.0117	0.0124	-0.00005
Small	Sim	Circle	0.3670	0.3682	0.1842	0.0233	0.0126	0.0122	-0.00012
		Ellip.	0.3692	0.3675	0.1835	0.0216	0.0100	0.0117	$-6 \cdot 10^{-05}$
		Rect.	0.3692	0.3678	0.1832	0.0229	0.0100	0.0120	$-7.2 \cdot 10^{-05}$
	Eq	Circle	0.3679	0.3679	0.1839	0.0224	0.0117	0.0124	-0.00005

Table 3.C.7: Baseline with boundary effects

Type	Shape	$\mathbb{E}[\mathcal{C}_0]$	$\mathbb{E}[\mathcal{C}_1]$	$\mathbb{E}[\mathcal{C}_2]$	$\mathrm{Sd}[\mathcal{C}_0]$	$\mathrm{Sd}[\mathcal{C}_1]$	$\mathrm{Sd}[\mathcal{C}_2]$	$\mathbb{C}\mathrm{ov}[\mathcal{C}_0, \mathcal{C}_1]$
Sim	Circle	0.3296	0.3661	0.2029	0.0258	0.0132	0.0140	$-7 \cdot 10^{-05}$
	Ellip.	0.3267	0.3648	0.2045	0.0250	0.0124	0.0137	$-6.1 \cdot 10^{-05}$
	Rect.	0.3234	0.3650	0.2062	0.0245	0.0129	0.0136	$-3.7 \cdot 10^{-05}$
Eq	Circle	0.3277	0.3656	0.2039	0.0256	0.0134	0.0142	$-6.6 \cdot 10^{-05}$

Table 3.C.8: Case(i): The expectation and standard deviation of 0-, 1-, and 2-layer coverage fractions with different flake sizes with boundary effects. Large $|F|$ is when $\mathbb{E}[|F|] = 392.70$ and $\mathrm{Sd}[|F|] = 471.47$ and small is when $\mathbb{E}[|F|] = 3.93$ and $\mathrm{Sd}[|F|] = 4.71$.

$\lvert F\rvert$	Type	Shape	$\mathbb{E}[\mathcal{C}_0]$	$\mathbb{E}[\mathcal{C}_1]$	$\mathbb{E}[\mathcal{C}_2]$	$\mathrm{Sd}[\mathcal{C}_0]$	$\mathrm{Sd}[\mathcal{C}_1]$	$\mathrm{Sd}[\mathcal{C}_2]$	$\mathbb{C}\mathrm{ov}[\mathcal{C}_0, \mathcal{C}_1]$
Large	Sim	Circle	0.2866	0.3604	0.2240	0.0570	0.0317	0.0315	$-4.1 \cdot 10^{-05}$
		Ellip.	0.2846	0.3573	0.2246	0.0535	0.0297	0.0298	$4.71 \cdot 10^{-05}$
		Rect.	0.2734	0.3539	0.2298	0.0541	0.0305	0.0302	0.000222
	Eq	Circle	0.2834	0.3573	0.2253	0.0563	0.0318	0.0319	0.000019
Small	Sim	Circle	0.3603	0.3680	0.1877	0.0051	0.0028	0.0029	$-5.3 \cdot 10^{-06}$
		Ellip.	0.3600	0.3678	0.1878	0.0049	0.0026	0.0027	$-4.4 \cdot 10^{-06}$
		Rect.	0.3592	0.3678	0.1883	0.0046	0.0025	0.0027	$-3.6 \cdot 10^{-06}$
	Eq	Circle	0.3601	0.3678	0.1878	0.0048	0.0024	0.0026	$-4 \cdot 10^{-06}$

Table 3.C.9: Case(ii): The expectation and standard deviation of 0-, 1-, and 2-layer coverage fractions with different $\mathbb{E}[N]\,p$ that corresponds to the mass concentration level with boundary effects. Large $\mathbb{E}[N]\,p$ is when $\mathbb{E}[N]\,p = 1.5$ and small is when $\mathbb{E}[N]\,p = 0.5$.

$\mathbb{E}[N]\,p$	Type	Shape	$\mathbb{E}[\mathcal{C}_0]$	$\mathbb{E}[\mathcal{C}_1]$	$\mathbb{E}[\mathcal{C}_2]$	$\mathrm{Sd}[\mathcal{C}_0]$	$\mathrm{Sd}[\mathcal{C}_1]$	$\mathrm{Sd}[\mathcal{C}_2]$	$\mathbb{C}\mathrm{ov}[\mathcal{C}_0, \mathcal{C}_1]$
Large	Sim	Circle	0.1895	0.3143	0.2622	0.0194	0.0145	0.0111	0.000122
		Ellip.	0.1874	0.3140	0.2627	0.0186	0.0141	0.0105	0.000118
		Rect.	0.1844	0.3109	0.2634	0.0186	0.0143	0.0103	0.000126
	Eq	Circle	0.1876	0.3139	0.2627	0.0190	0.0145	0.0110	0.000113
Small	Sim	Circle	0.5731	0.3187	0.0890	0.0304	0.0192	0.0127	-0.0005
		Ellip.	0.5706	0.3198	0.0900	0.0299	0.0184	0.0126	-0.00047
		Rect.	0.5677	0.3217	0.0906	0.0300	0.0181	0.0124	-0.00047
	Eq	Circle	0.5725	0.3193	0.0891	0.0300	0.0189	0.0127	-0.00049

Table 3.C.10: Case(iii): The expectation and standard deviation of 0-, 1-, and 2-layer coverage fractions with different R_F, the radius of printed pattern with boundary effects. Large R_F is when $R_F = 200$ and small is when $R_F = 100$.

R_F	Type	Shape	$\mathbb{E}[\mathcal{C}_0]$	$\mathbb{E}[\mathcal{C}_1]$	$\mathbb{E}[\mathcal{C}_2]$	$\mathrm{Sd}[\mathcal{C}_0]$	$\mathrm{Sd}[\mathcal{C}_1]$	$\mathrm{Sd}[\mathcal{C}_2]$	$\mathbb{C}\mathrm{ov}[\mathcal{C}_0, \mathcal{C}_1]$
Large	Sim	Circle	0.3399	0.3661	0.1980	0.0192	0.0100	0.0105	$-4.6 \cdot 10^{-05}$
		Ellip.	0.3378	0.3667	0.1988	0.0182	0.0096	0.0100	$-3.2 \cdot 10^{-05}$
		Rect.	0.3345	0.3661	0.2002	0.0187	0.0094	0.0102	$-2.9 \cdot 10^{-05}$
	Eq	Circle	0.3381	0.3666	0.1988	0.0188	0.0097	0.0103	$-4.3 \cdot 10^{-05}$
Small	Sim	Circle	0.3089	0.3631	0.2129	0.0411	0.0219	0.0221	-0.00011
		Ellip.	0.3074	0.3619	0.2135	0.0400	0.0202	0.0213	$-8.8 \cdot 10^{-05}$
		Rect.	0.2995	0.3607	0.2181	0.0384	0.0207	0.0207	$-3.9 \cdot 10^{-05}$
	Eq	Circle	0.3061	0.3624	0.2145	0.0402	0.0217	0.0226	$-8.1 \cdot 10^{-05}$

Table 3.C.11: Case(iv): The expectation and standard deviation of 0-, 1-, and 2-layer coverage fractions with different $Ratio(r)$, the ratio between two axis of ellipse and rectangle with boundary effects. Large $Ratio(r)$ is when $Ratio(r) = 0.7$ and small is when $Ratio(r) = 0.3$.

$Ratio(r)$	Type	Shape	$\mathbb{E}[\mathcal{C}_0]$	$\mathbb{E}[\mathcal{C}_1]$	$\mathbb{E}[\mathcal{C}_2]$	$\mathrm{Sd}[\mathcal{C}_0]$	$\mathrm{Sd}[\mathcal{C}_1]$	$\mathrm{Sd}[\mathcal{C}_2]$	$\mathbb{C}\mathrm{ov}[\mathcal{C}_0, \mathcal{C}_1]$
Large	Sim	Circle	0.3282	0.3656	0.2039	0.0261	0.0129	0.0144	$-6.7 \cdot 10^{-05}$
		Ellip.	0.3282	0.3653	0.2038	0.0259	0.0128	0.0144	$-5.5 \cdot 10^{-05}$
		Rect.	0.3227	0.3650	0.2064	0.0244	0.0133	0.0135	$-4.7 \cdot 10^{-05}$
	Eq	Circle	0.3277	0.3656	0.2039	0.0256	0.0134	0.0142	$-6.6 \cdot 10^{-05}$
Small	Sim	Circle	0.3289	0.3660	0.2033	0.0258	0.0129	0.0142	$-6.6 \cdot 10^{-05}$
		Ellip.	0.3233	0.3650	0.2062	0.0247	0.0116	0.0132	$-3.7 \cdot 10^{-05}$
		Rect.	0.3187	0.3641	0.2085	0.0242	0.0112	0.0131	$-1 \cdot 10^{-05}$
	Eq	Circle	0.3277	0.3656	0.2039	0.0256	0.0134	0.0142	$-6.6 \cdot 10^{-05}$

3.D Additional Information on Validation

3.D.1 Resolution of the Six Images

The resolution of the used images are as follows.

- Figure 14(a): 663×562 pixels
- Figure 14(b): 449×399 pixels
- Figure 14(c): 449×416 pixels
- Figure 14(d): 449×399 pixels
- Figure 14(e): 449×407 pixels
- Figure 14(f): 451×407 pixels

The first image resolution is distinctly different from the others because it is obtained from a different image collection setting.

3.D.2 Cut-Off Threshold Selection

There is a large body of literature on selecting the cut-off thresholds from the histogram of grey-scaled images (Otsu, 1979; Cheriet et al., 1998; Kotte et al., 2018; Huang and Wang, 2009). To determine the cut-off thresholds in this article, we used a heuristic method. First, the brightness values of the pixels inside the grey-scale image are plotted as a histogram (Figure 3.D.1). Second, the peak points of the histogram are selected, which will be the most representative brightness for each number of flake layers (dotted-lines). Then, between each pair of two neighboring peak points, the minimum point of the brightness values is selected for the thresholds (dashed-lines). The identified zero- and single-layers of the images are examined, and they look quite accurate (Figure 3.D.2).

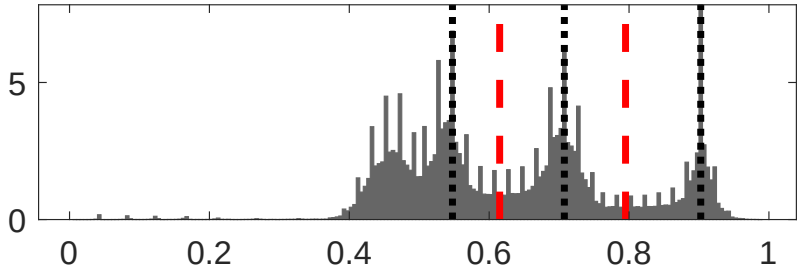
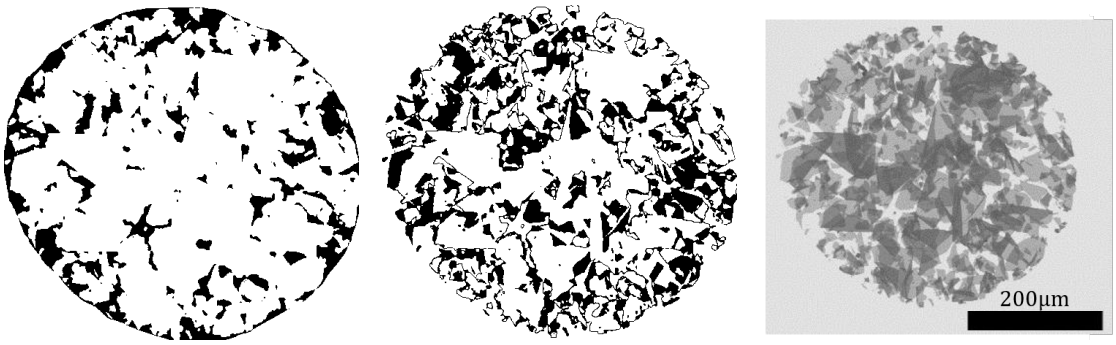


Figure 3.D.1: Histogram of the brightness of every pixel in the real inkjet-printed pattern image. The peak values are presented with dotted lines, and the cut-off values for zero- and single layer coverage are presented as dashed-lines.



(a) Zero-layer coverage area (b) Single-layer coverage area (c) Original Image

Figure 3.D.2: (a) The zero-layer coverage and (b) the single-layer coverage are extracted from (c) the inkjet printed pattern.

We provide the validation of our method by additionally conducting the case study by using Otsu's methods, which is one of the most popular histogram threshold selection techniques and proven effective (Kotte et al., 2018; Huang and Wang, 2009). The results from Otsu's method are presented in Table 3.D.1, and they are consistent with those from ours. In Table 3.D.1, the coverage fractions extracted from Otsu's method also identified Figure 14(d)-(f) as nonuniform patterns and Figure 14(a)-(c) as uniform patterns. More specifically, we used the Otsu's methods

recursively to determine the cut-offs (Otsu, 1979; Cheriet et al., 1998). First, the histogram of the brightnesses of pixels of images is constructed. Suppose we have L levels of brightness, and p_i is the relative frequency of the pixels in ith brightness level ($i = 1, \ldots, L$). Otsu's method selects the threshold t that maximizes the between-class variance σ_b^2.

$$\arg\max_t \sigma_b^2 = w_0(t)w_1(t)\left(\mu_1(t) - \mu_0(t)\right)^2 \tag{3.47}$$

where $w_0(t) = \sum_{i=1}^{t} p_i$, $w_1(t) = \sum_{i=t+1}^{L} p_i$, $\mu_0(t) = \sum_{i=1}^{t} ip_i/w_0$, and $\mu_1(t) = \sum_{i=t+1}^{L} ip_i/w_1$. Because Otsu's method makes the variance of the relative frequencies into consideration, the measurement errors of the brightness are inherently considered.

Table 3.D.1: Case study outcome with Otsu's method.

$\mathbb{E}[r]$	$\mathrm{Sd}[r]$	$\mathcal{C}_0$	$\mathcal{C}_1$	$\mathbb{E}[\mathcal{C}_0]$	$\mathbb{E}[\mathcal{C}_1]$	$\mathrm{Sd}[\mathcal{C}_0]$	$\mathrm{Sd}[\mathcal{C}_1]$	$\mathbb{C}\mathrm{ov}[\mathcal{C}_0, \mathcal{C}_1]$	Q_2	p-value
17.95	11.6	0.0829	0.2572	0.0773	0.1979	0.0542	0.0732	0.0028	1.07	0.5856
10.85	7.9	0.1015	0.2731	0.1041	0.2355	0.0414	0.048	0.0015	1.485	0.4759
3.55	4.5	0.1961	0.3444	0.1081	0.2405	0.046	0.0528	0.0021	4.049	0.132
1.85	1.9	0.1534	0.3376	0.1556	0.2895	0.0144	0.0122	0.0001	45.103	1.6×10^{-10}
0.9	2.15	0.1204	0.4275	0.1002	0.2305	0.059	0.0712	0.0037	29.11	4.8×10^{-7}
0.475	0.225	0.2426	0.2113	0.1768	0.3063	0.001	0.0008	0	18,899	0

4 LANDMARK-EMBEDDED GAUSSIAN PROCESS MODEL WITH APPLICATIONS FOR FUNCTIONAL DATA CALIBRATION *

In practice, we often need to infer the value of a target variable from functional observation data. A challenge in this task is that the relationship between the functional data and the target variable is very complex: the target variable not only influences the shape but also the location of the functional data. In addition, due to the uncertainties in the environment, the relationship is probabilistic, that is, for a given fixed target variable value, we still see variations in the shape and location of the functional data. To address this challenge, we present a landmark-embedded Gaussian process model that describes the relationship between the functional data and the target variable. A unique feature of the model is that landmark information is embedded in the Gaussian process model so that both the shape and location information of the functional data are considered simultaneously in a unified manner. Gibbs-Metropolis-Hasting algorithm is used for model parameters estimation and target variable inference. The performance of the proposed framework is evaluated by extensive numerical studies and a case study of nano-sensor calibration.

4.1 Introduction

In practice, we often need to infer the value of an unobserved variable from functional observation data. In this chapter, we call the unobserved variable "target

*This chapter is based on the paper: **Lee, J.**, C. Wang, X. Sui, S. Zhou, and J. Chen (2021). Landmark-embedded gaussian process model with applications for functional data calibration. *IISE Transactions.* https://doi.org/10.1080/24725854.2021.1974129.

variable"; specifically, we consider a continuous target variable. For example, it is common that we need to infer the tools' conditions (e.g., misplacement) from the functional sensing signals collected from the process in manufacturing process control (Lei et al., 2010a). In sensor development, we often want to infer the value of an underlying physical variable from direct sensor observations. Indeed, this work is motivated by developing a graphene-based field-effect transistor (GFET) sensor to measure the concentration of the heavy metal in water (Zhou et al., 2014). The GFET sensor is one of the widely known nano-sensors that have been developed to detect a wide range of target substances, including heavy metals (Zhou et al., 2014), Escherichia coli (E. coli) (Thakur et al., 2018), and specific proteins (Liu et al., 2012). The GFET sensors can be made cheaply through printing processes, and be portable and disposable (Maity et al., 2017). Such sensors are very useful in practice to have frequent measurements of water quality at multiple locations. Indeed, frequent wide area sensing is especially crucial in the source water quality control to prevent catastrophic water contamination (Maity et al., 2017). To utilize the data obtained from cheaply made sensors, which has large variations, we aim to predict the target variable (e.g., target substance) and its predicted interval by leveraging functional data modeling. We call such a prediction as "statistical calibration" as the statistical inference of the unobserved input variable values from the observed responses is often referred to as "statistical calibration", or simply "calibration" (Osborne, 1991). A GFET sensor consists of a gate, a drain electrode, a source electrode, a graphene channel that bridges the drain and source, and nano-particle probes scattered on the channel (Figure 4.1.1). The probes can bind with a specific target substance

in water and change the electronic properties of the graphene channel. The gate voltage also impacts on the electrical conductivity of the graphene channel. As a result, for a given sensor and a specific concentration value of the target substance, we will see a specific functional signal of drain-source current vs. gate voltage. This signal can be obtained by varying gate voltage and measuring the corresponding drain-source current. In principle, we could use physical principles to infer the concentration value from the sensor signals. However, the physical relationship between the sensor signal and the concentration is very complex. Furthermore, many sensor parameters that will be needed by the physical model will not be precisely known. Thus, the physics-based calibration approach is infeasible. As a result, we employ a data-driven method to calibrate the functional data. We first build up a statistical model linking the sensor signal and the concentration using a historical training dataset and then infer the concentration value for a new signal.

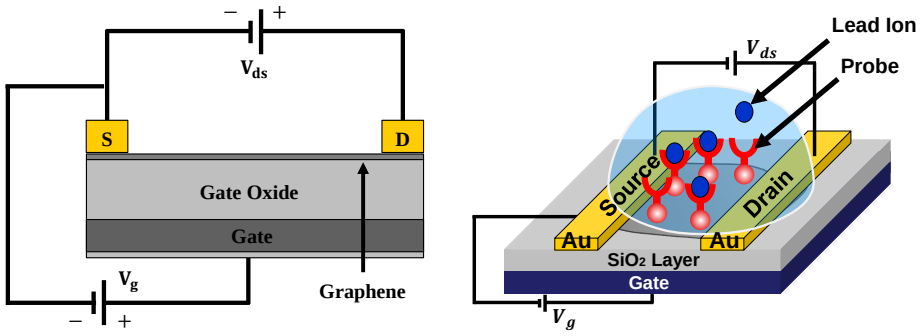


Figure 4.1.1: Field-effect transistor sensor

Figure 4.1.2 shows the sensor signals obtained from multiple GFET sensors in contaminated water with three different lead ion concentrations. From these data,

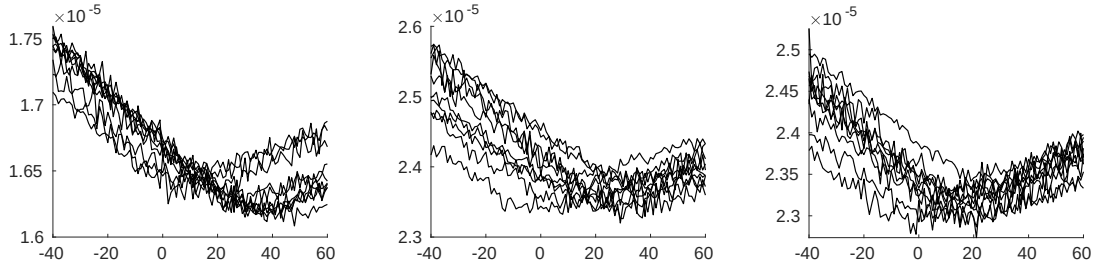
we can see some obvious characteristics:

- The signals have significant variations even under the same concentration value. The reason is that the probes in the GFET sensor can be only used once; thus, the sensor is for single use. For repeated measures of the same concentration value, we have to use multiple sensors. The variations in sensor structures will lead to significant variations in the sensor signals. In this chapter, we call the variations from each signal "same-level" variation. Similarly, we call the signal variations across different concentration values "across-level" variation; here, level refers to the value of the target variable.

- We can find a common feature among different signals in Figure 4.1.2: every signal has a V-shape. This is due to the carrier bipolar transport behavior in the graphene, and the minimum current is called the Dirac point (Wang et al., 2016). However, there are same-level and across-level variations in not only the shape of the V but also the location of the V bottom.

Clearly, to effectively model these characteristics, we need to model four main types of variations: same-level shape, same-level location, across-level shape, and across-level location variations. Due to the significant variations in the sensing signals, the predicted concentration value would inevitably involve uncertainties. Therefore, it is desirable to provide the interval estimates of the concentration values. Under the uncertainties of the predicted values, interval estimates are essential in sensor applications for the decision-makers to make better decisions and in sensor development to quantify sensor repeatability. The goal of this chapter

is to establish a comprehensive model to infer the point and interval estimates of the target substance concentration value for a new sampled signal based on the historical data. With this approach, we hope to improve the accuracy of the nanosensor for measuring the target substance concentration value.



(a) 0 ppb signals (b) 10 ppb signals (c) 20 ppb signals

Figure 4.1.2: The drain-source currents are measured with respect to a sequence of gate voltages from contaminated water with 0, 10, and 20 ppb lead ions. (Units: ampere on vertical and voltage on horizontal axes).

Statistical calibration has been widely used in predicting and improving sensor performance in various applications. For example, the camera irradiance needs to be calibrated from the charge-coupled device (CCD) images to conduct a color correction and white balancing for CCD sensors (Tsin et al., 2001). In the radio-frequency integrated circuit (RFIC) sensors, testing circuit measurements are used to calibrate the knob settings (e.g., tunable resistance or capacitance) to correct the sensor bias (Lu et al., 2015; Han et al., 2009). However, most of these calibration techniques are based on scalar-to-scalar mapping, which cannot characterize the functional measurements and variation features in our problem.

Functional regression models with a scalar response variable and functional predictors can be considered for the functional data calibration. For example,

the functional linear regression model with B-spline basis can be used for the non-periodic functional signal data to predict the concentration (Ramsay and Silverman, 2005). The functional linear regression model, however, assumes a fixed relationship between the response variable and the predictors of the population; therefore, it does not consider the random characteristics of the functional signals including the shape and location variations. Similarly, many machine learning techniques, such as deep neural networks, 1) often cannot provide quantification of the uncertainty of the prediction (Khosravi et al., 2011) and 2) require a significant amount of data (Ingrassia and Morlini, 2005); such big data may not be available in practice. Therefore, they are not suitable to address the uncertainties in our data.

The Gaussian process model is a widely used nonparametric model to characterize random functional signals (Rasmussen and Williams, 2006; Santner et al., 2003). However, the conventional Gaussian process model does not differentiate same-level and across-level shape variations and does not explicitly consider the random shifts in signal locations. Obviously, if the random shifts in signal locations are not considered, the fitted Gaussian process model will be misleading with wrong shape and exaggerated variation (Ramsay and Silverman, 2005). A straightforward idea to consider the location variation in the signals is two-steps methods. First, align the signals by vertically and horizontally shifting the signal. For example, landmark registration, which aligns the signals by superimposing signals' location features, can be applied (Gasser and Kneip, 1995; Ramsay et al., 2014). Such location features, also called 'landmarks', are defined by the features observed in the given functional dataset. For the GFET sensing data, the bottom of

the V shape is a natural choice for the landmark. Second, fit a Gaussian process model to the aligned signals with the target variable (e.g., the concentration) as a predictor and infer the value of the target variable for a sample signal based on the fitted Gaussian process model. However, in this simple two-steps method, the target variable is inferred only through the shape information of the signal, and the signal shift information is ignored. Of course, one can further build a model to link the target variable with the location shifts of the signals. Then, the target variable can be inferred based on both the signal shape and location information. However, how to combine the calibration result from shape information and that from location information is not clear. Furthermore, in this two-steps procedure, the location variation and shape variation are considered separately in the signal alignment and model fitting/inference procedure, which will lead to inefficiency and suboptimal results in both signal shape and location estimation.

In this chapter, we propose a landmark-embedded Gaussian process model to consider the above-mentioned four types of variations, namely same-level shape and location variations and across-level shape and location variations, in a unified framework. This framework has two types of latent functions: one represents the same-level variations in shape and location of the signals, and the other represents the across-level variations in shape and location. The signals' location information is characterized by the embedded landmark information in the Gaussian process. The embedded landmark ensures precise modeling of 1) individual signal' shape, 2) signal shape changes over target variable, and 3) signal location changes over the target variable. The landmarks are included as parameters in the model, and

all the parameters including the landmarks are estimated simultaneously. This integrative modeling of the whole data is very beneficial, by which we can accurately estimate the landmarks as well as other parameters. A hybrid Gibbs sampling method with the Metropolis-Hasting algorithm is developed for model parameters estimation and target variable inference. This proposed functional data calibration method is robust and effective because all the available signals are considered simultaneously. Comprehensive numerical and real-world case studies demonstrate that the calibration performance is significantly better than various competing methods including the above two-steps method.

We would like to mention that our work is distinct from the 'computation model'-based Bayesian calibration (Kennedy and O'Hagan, 2001; Higdon et al., 2013; Goh et al., 2013; Plumlee et al., 2016; Gramacy et al., 2015; Lee et al., 2020; Farmanesh et al., 2020). First, the Bayesian calibration methods aim to estimate the *population-level* 'calibration parameters' that make the computational model best *fitted* to the data set. The ideal values of the calibration parameters to be found are fixed and correspond to the whole data set and model. On the other hand, we aim to *predict* the target variable (e.g., contamination level) value that is *specific* to one signal based on the historical dataset. Therefore, the model structure is quite different. Second, the responses of the models in the Bayesian computational calibration literature are scalar variables; only the relationship between the response and the calibration parameters are designed by functional models, such as Gaussian processes. However, the responses in our data are functional data. In Plumlee et al. (2016), the calibration parameter is of functional form, but the responses are still

scalar. In Guo et al. (2017), to handle functional response, the authors estimated the intermediate scalar variables by the physics-based mathematical model. Then, they used these estimated scalar values as the responses of the Gaussian process and performed calibration by using a typical Bayesian calibration process. Our proposed method is data-driven and directly uses the functional responses to predict the value of the associated target variable in our model. Third, we have complex randomness in the functional type of responses. Individual signals are randomly shaped and shifted, which cannot be characterized by the conventional Gaussian processes used in the literature.

The rest of the chapter is organized as follows. Section 4.2 presents the structure of the proposed landmark-embedded Gaussian process model. Section 4.3 describes MCMC sampling methods that estimate the parameters in the proposed method and infer the target variable given newly observed signals based on the fitted model; In Section 4.3.1, we first provide a roadmap of using the Gibbs sampling for the parameter estimations and target variable calibration. Then, the specific Gibbs sampling methods are described for parameter estimation in Section 4.3.2 and calibration in Section 4.3.3. Section 4.4 demonstrates the performance of the proposed calibration method with the proposed model by extensive numerical studies. Case study results are presented to show the effectiveness of our model in Section 4.5. Section 4.6 draws a conclusion and shows a brief discussion.

4.2 Proposed model structure

Suppose functional data are obtained in M groups, and each group corresponds to a distinct value of the target variable denoted by $t_i \in \mathbb{R}$, $i = 1, \ldots, M$. For group i, we have N_i functional samples, denoted by $\{\mathbf{x}_{ij}, \mathbf{y}_{ij}\}$, $j = 1, \ldots, N_i$, where $\mathbf{y}_{ij} = \left(y_{ij1}, \ldots, y_{ijL_{ij}}\right)^{\mathsf{T}} \in \mathbb{R}^{L_{ij}}$ is the response and $\mathbf{x}_{ij} = \left(x_{ij1}, \ldots, x_{ijL_{ij}}\right)^{\mathsf{T}} \in \mathbb{R}^{L_{ij}}$ is the predictor. Further, we denote a landmark for each functional sample as $(\lambda_{ij}^H, \lambda_{ij}^V)$, where λ_{ij}^H is the "horizontal landmark" and λ_{ij}^V is the "vertical landmark"; λ_{ij}^H and λ_{ij}^V are the position of the landmark on the horizontal and vertical axes, respectively. Landmarks are defined by the common features of the functional data. We can choose different features as the landmark for different applications (e.g., maximum point, or point when the signal passing a certain value). For the problem of GFET sensor calibration, the minimum point of the functional sample (i.e., the V bottom) can be selected as its landmark. For notation simplicity, we assume the number of functional samples in each group and the length of each functional sample are identical, i.e., $N_i = N$, $L_{ij} = L$, $\mathbf{x}_{ij} = \mathbf{x} = (x_1, \ldots, x_L)^{\mathsf{T}}$. This assumption is only for notation simplicity and can be easily relaxed in the proposed framework. Thus, we denote the training data as $\mathcal{D} = \{(\mathbf{y}_{ij}, \mathbf{x}, t_i)\}$, $i = 1, \ldots, M$ and $j = 1, \ldots, N$. Please note that the landmarks are included as parameters and to be estimated with other parameters. To describe this data, we propose a Gaussian process model in a special structure as follows.

$$\mathbf{y}_{ij} = \boldsymbol{f}_{ij}^G(\mathbf{x}, t_i) + \boldsymbol{f}_{ij}^I(\mathbf{x}) + \lambda_{ij}^V \mathbb{1}_{[L]} + \boldsymbol{\epsilon}_{ij} \tag{4.1}$$

$$\boldsymbol{f}_{ij}^I(\mathbf{x}) \sim \mathcal{N}\left(\mathbf{0}, \boldsymbol{\Omega}_{(ij,i'j')}^I\left(\mathbf{x}; \lambda_{ij}^H\right)\right) \tag{4.2}$$

$$\boldsymbol{f}_{ij}^{G}(\mathbf{x}, t_i) \sim \mathcal{N}\left(\mathbf{0}, \boldsymbol{\Omega}_{(ij,i'j')}^{G}\left(\mathbf{x}, t_i; \lambda_{ij}^{H}\right)\right) \tag{4.3}$$

$\mathbf{y}_{ij}$ is the observed functional signal and represented by the summation of measurement noises $\boldsymbol{\epsilon}_{ij}$ and the denoised signal $(\boldsymbol{f}_{ij}^{G}(\mathbf{x}, t_i) + \boldsymbol{f}_{ij}^{I}(\mathbf{x}) + \lambda_{ij}^{V}\mathbb{1}_{[L]})$. $\boldsymbol{f}_{ij}^{G}$ and $\boldsymbol{f}_{ij}^{I}$ are two independent Gaussian process latent functions that have two distinct roles. $\boldsymbol{f}_{ij}^{G}(\mathbf{x}, t_i) = \left(f_{ij}^{G}(x_1, t_i), \ldots, f_{ij}^{G}(x_L, t_i)\right)^{\mathsf{T}} \in \mathbb{R}^L$ is defined over the predictor $\mathbf{x}$ and target variable t and captures the across-level variations in shape and location. In particular, f_{ij}^{G} characterizes the common shape of the function at the target variable t_i. $\boldsymbol{f}_{ij}^{I}(\mathbf{x}) = \left(f_{ij}^{I}(x_1), \ldots, f_{ij}^{I}(x_L)\right)^{\mathsf{T}} \in \mathbb{R}^L$ is a function of only $\mathbf{x}$ and captures the same-level variations in shape and location. $\boldsymbol{f}_{ij}^{I}$ additionally accounts for the random difference of the individual signal's shape from the common shape, caused from the individual sensor-to-sensor variations. The vertical landmark $\lambda_{ij}^{V}\mathbb{1}_{[L]}$ is the mean of $\mathbf{y}_{ij}$, where $\mathbb{1}_{[L]}$ is a column vector with L rows whose every component is one, and $\boldsymbol{\epsilon}_{ij}$ is additive measurement noise whose component is an independent normal random variable with mean zero and variance σ_{ε}^2. This proposed Gaussian process model in (4.1)-(4.3) is quite flexible and captures the four types of variations of the data mentioned in the introduction section.

The Gaussian process model is flexible and effective to model functional data, where functional samples are viewed as the realizations from a multivariate normal distribution. In the specifications of a Gaussian process model, the covariance function plays a critical role in determining the model characteristics. Here, we

propose the following covariance terms,

$$\mathbf{\Omega}^{I}_{(ij,i'j')}(\mathbf{x}) = \delta_{ii'}\delta_{jj'}\sigma_I^2 k_I\Big(\mathbf{x} - \lambda^H_{ij}, \mathbf{x} - \lambda^H_{i'j'}\Big) \tag{4.4}$$

$$\mathbf{\Omega}^{G}_{(ij,i'j')}(\mathbf{x}, t_i) = \sigma_G^2 k_G\Big(\Big(\mathbf{x} - \lambda^H_{ij}, t_i\Big), \Big(\mathbf{x} - \lambda^H_{i'j'}, t_{i'}\Big)\Big) \tag{4.5}$$

where δ_{kh} is the Kronecker delta function, σ_I^2 and σ_G^2 are the prior vertical variations and $k_I(\cdot,\cdot)$ and $k_G(\cdot,\cdot)$ are the kernel functions of $\boldsymbol{f}^I_{ij}(\cdot,\cdot)$ and $\boldsymbol{f}^G_{ij}(\cdot,\cdot)$. The detailed information of the kernels used in this work is presented in Appendix A in the supplementary material. The horizontal landmark λ^H_{ij} is integrated into the covariance function for precise characterization of the functional shape from the aligned signals. With the above covariance function definition, the covariance matrix $\mathbf{\Omega}_{\boldsymbol{Y}} = \mathbb{C}\mathrm{ov}\left(\boldsymbol{Y}, \boldsymbol{Y}\right)$ of the overall data $\boldsymbol{Y} = \left(\mathbf{y}_{11}^{\mathsf{T}}, \ldots, \mathbf{y}_{MN}^{\mathsf{T}}\right)^{\mathsf{T}}$ is

$$\mathbf{\Omega}_{\boldsymbol{Y}} = \begin{bmatrix} \mathbf{\Omega}^G_{11} + \mathbf{\Omega}^I_{11} & \mathbf{\Omega}^G_{(11,12)} & \cdots & \mathbf{\Omega}^G_{(11,ij)} & \cdots & \mathbf{\Omega}^G_{(11,MN)} \\ \mathbf{\Omega}^G_{(12,11)} & \mathbf{\Omega}^G_{12} + \mathbf{\Omega}^I_{12} & \cdots & \mathbf{\Omega}^G_{(12,ij)} & \cdots & \mathbf{\Omega}^G_{(12,MN)} \\ \vdots & \vdots & \ddots & \vdots & & \vdots \\ \mathbf{\Omega}^G_{(ij,11)} & \mathbf{\Omega}^G_{(ij,12)} & \cdots & \mathbf{\Omega}^G_{ij} + \mathbf{\Omega}^I_{ij} & \cdots & \mathbf{\Omega}^G_{(ij,MN)} \\ \vdots & \vdots & & \vdots & \ddots & \vdots \\ \mathbf{\Omega}^G_{(MN,11)} & \mathbf{\Omega}^G_{(MN,12)} & \cdots & \mathbf{\Omega}^G_{(MN,ij)} & \cdots & \mathbf{\Omega}^G_{MN} + \mathbf{\Omega}^I_{MN} \end{bmatrix} + \sigma_\varepsilon^2 I_{[MNL]} \tag{4.6}$$

where $I_{[MNL]}$ is the identity matrix with dimension $MNL \times MNL$. $\mathbf{\Omega}^G_{(ij,i'j')} = \mathbb{C}\mathrm{ov}\left(\boldsymbol{f}^G_{ij}, \boldsymbol{f}^G_{i'j'}\right)$ represents the covariance between different signals $\mathbf{y}_{ij}$ and $\mathbf{y}_{i'j'}$. $\mathbf{\Omega}^I_{ij} = \mathbf{\Omega}^I_{(ij,ij)} = \mathbb{C}\mathrm{ov}\left(\boldsymbol{f}^I_{ij}, \boldsymbol{f}^I_{ij}\right)$ and $\mathbf{\Omega}^G_{ij} = \mathbf{\Omega}^G_{(ij,ij)} = \mathbb{C}\mathrm{ov}\left(\boldsymbol{f}^G_{ij}, \boldsymbol{f}^G_{ij}\right)$ on the diagonal

components of $\boldsymbol{\Omega}_{\boldsymbol{Y}}$ represent covariance structure within each functional sample $\mathbf{y}_{ij}$.

For the probabilistic relationship between the landmarks and the target variable, we adopt the Gaussian process prior for the landmarks with the target variable as the predictor; this enables us to infer the target variable value based on the location information of the signals in calibration. We denote the entire horizontal and vertical landmarks by $\boldsymbol{\lambda}^H = \left(\lambda_{11}^H, \ldots, \lambda_{MN}^H\right)^\mathsf{T} \in \mathbb{R}^{MN}$ and $\boldsymbol{\lambda}^V = \left(\lambda_{11}^V, \ldots, \lambda_{MN}^V\right)^\mathsf{T} \in \mathbb{R}^{MN}$, and their corresponding target variable vector by $\boldsymbol{t}_\lambda = \left(t_1 \mathbb{1}_{[N]}^\mathsf{T}, \ldots, t_M \mathbb{1}_{[N]}^\mathsf{T}\right)^\mathsf{T} \in \mathbb{R}^{MN}$. Then, $\boldsymbol{\lambda}^H$ and $\boldsymbol{\lambda}^V$ follow the multivariate normal distributions.

$$\boldsymbol{\lambda}^H \sim \mathcal{N}\left(\mathbf{0}, \sigma_H^2 k_H(\boldsymbol{t}_\lambda, \boldsymbol{t}_\lambda)\right) + \boldsymbol{\epsilon}_H \tag{4.7}$$

$$\boldsymbol{\lambda}^V \sim \mathcal{N}\left(\mathbf{0}, \sigma_V^2 k_V(\boldsymbol{t}_\lambda, \boldsymbol{t}_\lambda)\right) + \boldsymbol{\epsilon}_V \tag{4.8}$$

where $\boldsymbol{\epsilon}_H \sim \mathcal{N}\left(\mathbf{0}, \sigma_{\varepsilon H}^2 I_{[MN]}\right)$ and $\boldsymbol{\epsilon}_V \sim \mathcal{N}\left(\mathbf{0}, \sigma_{\varepsilon V}^2 I_{[MN]}\right)$. The noises $\boldsymbol{\epsilon}_H$ and $\boldsymbol{\epsilon}_V$ account for the variations on the landmarks at a given target variable. The kernel function $k_H(\cdot, \cdot)$ and $k_V(\cdot, \cdot)$ are specified in Appendix A in the supplementary material. Equations (4.1)-(4.3), (4.7), and (4.8) provide the complete model specification.

The kernel functions used for the Gaussian processes often involve the hyperparameters (e.g., lengthscale parameter for Gaussian kernels). The kernel functions and the hyperparameters constitute the Gaussian process prior, and they need to be selected based on our *a priori* knowledge on the functional signals; the hyperparameters are determined based on the functional signals change as the predictor

values change (Plumlee et al., 2016). The hyperparameters used in this work is discussed in Appendix A.3 in the supplementary material.

Figure 4.2.1 illustrates how our proposed model can successfully address the challenges from both shape and location variations to achieve accurate point and interval estimates. To precisely model the functional signals with significant random variations, the proposed method facilitates characterizing both the common shape and the individual differences in the shape of the functional signals for a given value of the target variable. The location shifts of the signals against the target variable are captured by the Gaussian process priors (presented in Equations (4.7) and (4.8)) as illustrated in Figure 4.2.1(c). Figure 4.2.1(d) depicts the intermediate results of our proposed methods fitted to the data. Horizontal and vertical landmarks are used to align the functional data by moving the signals along the x-axis and y-axis, respectively, so that the landmarks are located at the origin. The Gaussian process model is based on the aligned functional data; therefore, the estimated global mean function has a similar shape with individual signals, and the estimated variances (and the confidence intervals) are small. f^G characterizes the common shape of aligned signals for a given target variable value. Then, the unexplained parts of the signals are accounted for by f_{ij}^I. In Figure 4.2.1(e), each signal is represented by the common shape by f^G (solid line) and the remained unexplained parts are further accounted for by f_{ij}^I (dashed line), which facilitate precisely representing the individual signals. In contrast, the conventional Gaussian process presented in Figure 4.2.1(b) provides a generic shape of one mean function for all the signals obtained from the same value of the target variable. To achieve accurate interval

estimates from calibration, we also need to precisely model the uncertainties in the signals. Our proposed model provides precise shapes for the fitted mean functions and narrow confidence intervals tailored for each signal, as shown in Figure 4.2.1(f). The precise mean estimation and narrow confidence interval facilitate accurate prediction. The conventional Gaussian process model without alignment, on the other hand, shows a wide confidence interval in Figure 4.2.1(b) because the signals have different shapes at the same predictor values.

We would like to point out that the use of the landmarks is not confined to the V-shape features of functional data. For example, landmarks can also be defined as the points where the functions passing through a specified value or as the local maximum or minimum. Such features are commonly observed in functional data in various applications because many functional data have a local maximum or minimum point (Lei et al., 2010b; Reza et al., 2020). When we have multiple local maximums and/or minimums, one of which can be specified as a landmark. In Mosesova (2007), landmarks are defined by the first point where the first differences of the functional data become positive. A brief discussion on potential identifiability issues on the landmark specification is provided in Appendix B.

4.3 Model parameter estimation and calibration

We first estimate the model parameters, denoted by $\Theta = \{\sigma_\varepsilon^2, \sigma_I^2, \sigma_G^2, \boldsymbol{\lambda}^H, \boldsymbol{\lambda}^V, \sigma_H^2, \sigma_V^2, \sigma_{\varepsilon H}^2, \sigma_{\varepsilon V}^2\}$ and then infer the target variable t based on the estimated parameters $\widehat{\Theta}$ by using the Markov chain Monte Carlo sampling method. In section 4.3.1, we provide

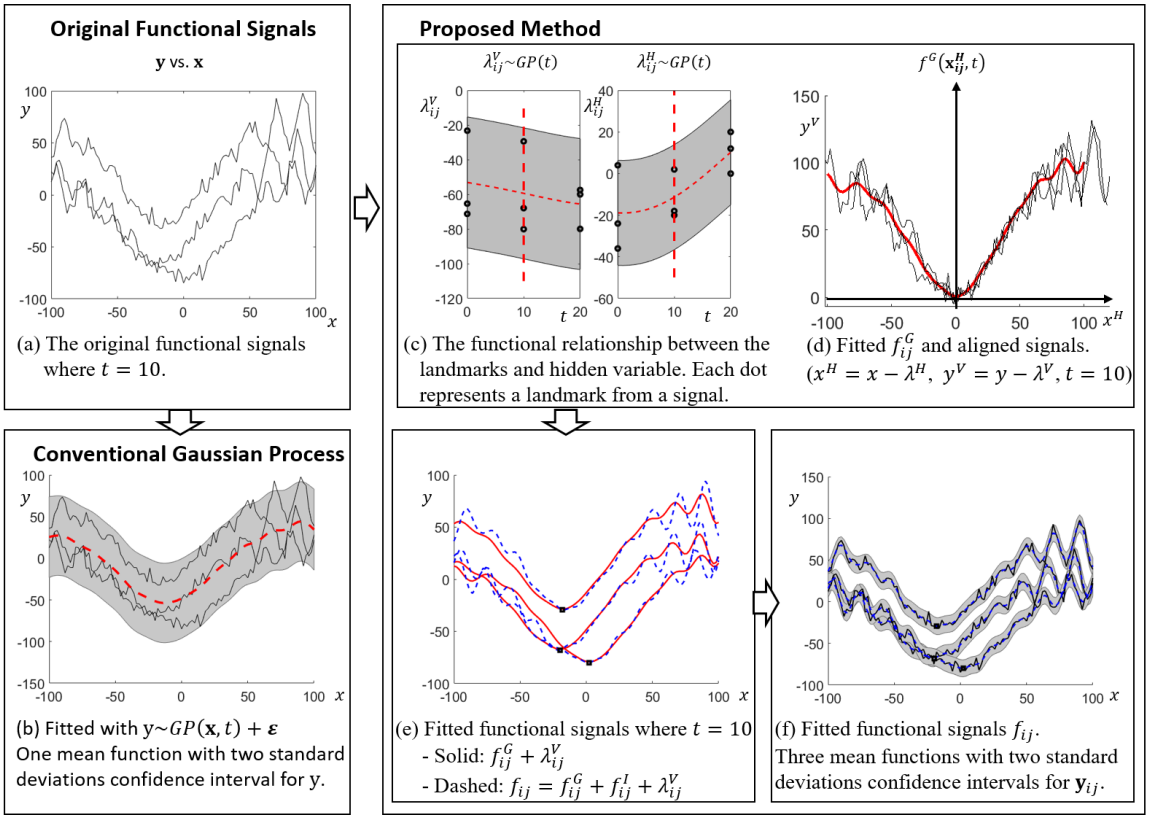


Figure 4.2.1: The conventional Gaussian process and our proposed model are fitted to the dataset that includes nine functional signals where three signals are collected from each of three target variable values (i.e., t is one of $(0, 10, 20)$). Two standard deviation confidence intervals are presented. (a) The original functional signals where $t = 10$. (b) The conventional Gaussian process fitted to the functional signals. The mean and confidence intervals at $t = 10$ are presented. (c) The relationship between the landmarks (i.e., functional signals' locations) and the target variables accounted for by the Gaussian process priors (presented in (4.7) and (4.8)). (d) The fitted f_{ij}^G, aligned signals, and the confidence intervals for the aligned signals at $t = 10$ are presented. (e) Solid lines ($\boldsymbol{f}_{ij}^G(\mathbf{x}, t_i) + \lambda_{ij}^V$) show the predicted functional signals with the estimated common shapes ($\boldsymbol{f}_{ij}^G$), and the estimated landmarks ($\lambda_{ij}^H, \lambda_{ij}^V$) are presented in square dots. The difference in individual signal' shape is adjusted by $\boldsymbol{f}_{ij}^I(\mathbf{x})$; therefore, the dashed lines have similar shapes to the original signals but smoother because the measurement noises are removed. (f) The complete model, denoted by f_{ij}, fitted to the original signals. The mean functions (dashed lines) and confidence intervals are presented along with the original signals (solid lines).

a general structure of the Gibbs sampler that we employ, and the specific sampling processes for the model parameter estimations and calibration are described in Section 4.3.2 and Section 4.3.3, respectively.

4.3.1 Roadmap of parameter estimation and calibration

Based on the training dataset $\mathcal{D}$, we want to estimate $\Theta = \{\sigma_\varepsilon^2, \sigma_I^2, \sigma_G^2, \boldsymbol{\lambda}^H, \boldsymbol{\lambda}^V, \sigma_H^2, \sigma_V^2, \sigma_{\varepsilon H}^2, \sigma_{\varepsilon V}^2\}$ in parameters estimation and infer $\left(t_{new}, \lambda_{new}^V, \lambda_{new}^H\right)$ in calibration for a sampled functional signal. For simplicity, here, we describe the sampling process for Θ, but the same idea applies to calibration, instead of sampling Θ, by sampling $\left(t_{new}, \lambda_{new}^V, \lambda_{new}^H\right)$ conditioning on $\widehat{\Theta}$. We employ the Gibbs sampling, which is an iterative procedure to sample from the joint posterior distribution, $\mathbb{P}\left(\Theta|\mathcal{D}\right)$, and the Metropolis-Hasting algorithm is used to draw samples from the non-trivial probability distributions; how to draw a sample using the Metropolis-Hasting algorithm can be found in Appendix C in the supplementary material. The samples from nth iterations are denoted by $\Theta^{(n)} = \left\{\theta_1^{(n)}, \ldots, \theta_P^{(n)}\right\}$ where $\theta_k^{(n)}$ is a kth parameter in $\Theta^{(n)}$; for example, θ_1 is the first parameter σ_ε^2 and θ_9 is $\sigma_{\varepsilon V}^2$. Gibbs sampling is a specific case of the Metropolis-Hasting algorithm. The Metropolis-Hasting algorithm constructs a Markov chain that is irreducible and aperiodic and samples from the chain; such chain ensures the sampling distribution converges to the stationary posterior distribution (Gelman et al., 2013a). Brief description of the sampling method is presented in Appendix D of the supplementary material. We briefly discussed the computational complexity of the proposed method in Appendix G in the supplementary material. The detailed cyclic sampling processes specific for the

model estimation and calibration are described in the following Section 4.3.2 and 4.3.3.

4.3.2 Model parameter estimation

The model parameters Θ are estimated by the Gibbs sampler described in Section 4.3.1. In particular, we sample the landmarks $\boldsymbol{\lambda}^H$ and $\boldsymbol{\lambda}^V$ and all the variances in the model from the full conditional distribution. Landmarks are obtained from the posterior mean of the functional signal denoised from $\mathbf{y}_{ij}$, denoted by f_{ij} (i.e., $f_{ij} = f_{ij}^G + f_{ij}^I + \lambda_{ij}^H$), rather than from the observations $\mathbf{y}_{ij}$ directly to achieve robustness and accuracy by removing the effects of measurement noises. The types of the landmark can be specified based on the features of the given functional data. The technical details are presented in Appendix E in the supplementary material. The iterative cyclic sampling process to nth sample $\Theta^{(n)}$ is presented below.

- **Step 1**: *sample horizontal landmarks* $\boldsymbol{\lambda}^{H\,(n)} = \left(\lambda_{11}^{H\,(n)}, \ldots, \lambda_{MN}^{H\,(n)}\right)^{\mathsf{T}}$*, and vertical landmarks* $\boldsymbol{\lambda}^{V\,(n)} = \left(\lambda_{11}^{V\,(n)}, \ldots, \lambda_{MN}^{V\,(n)}\right)^{\mathsf{T}}$ given the sampled parameters in the previous step and the training set $\mathcal{D}$ including the whole signals $\boldsymbol{Y} = \left(\mathbf{y}_{11}^{\mathsf{T}}, \ldots, \mathbf{y}_{MN}^{\mathsf{T}}\right)^{\mathsf{T}}$:
  - For each $i = 1, \ldots, M$, $j = 1, \ldots, N$
    1. *Obtain posterior mean of* $f_{ij}(x, t_i)$:

$$\boldsymbol{\Omega}_{f_{ij}\boldsymbol{Y}}^{(n-1)} \left(\boldsymbol{\Omega}_{\boldsymbol{Y}}^{(n-1)}\right)^{-1} \left(\boldsymbol{Y} - \boldsymbol{\lambda}_{\boldsymbol{Y}}^{V\,(n-1)}\right) + \lambda_{ij}^{V\,(n-1)} \tag{4.9}$$

where $\boldsymbol{\Omega}_{\boldsymbol{Y}}^{(n-1)}$ is the prior covariance matrix of $\boldsymbol{Y}$ and $\boldsymbol{\Omega}_{f_{ij}\boldsymbol{Y}}^{(n-1)}$ is the prior joint covariance matrix between f_{ij} and $\boldsymbol{Y}$, constructed with the $(n-1)$th sampled parameters ${\boldsymbol{\lambda}^H}^{(n-1)}$, ${\sigma_G^2}^{(n-1)}$, ${\sigma_I^2}^{(n-1)}$, and ${\sigma_\varepsilon^2}^{(n-1)}$. $\boldsymbol{\lambda}_{\boldsymbol{Y}}^{V\,(n-1)} = \left({\lambda_{11}^V}^{(n-1)}\mathbb{1}_{[L]}^\mathsf{T}, \ldots, {\lambda_{MN}^V}^{(n-1)}\mathbb{1}_{[L]}^\mathsf{T}\right)^\mathsf{T}$ is a $(n-1)$th sampled vertical landmarks vector corresponding to $\boldsymbol{Y}$. The detailed derivation of (4.9) is presented in Appendix E.1 in the supplementary material.

2. *Obtain landmarks*: the minimum point of the posterior mean of the signal is used in this work.

$$\lambda_{ij}^{H(n)} = \arg\min_x \mathbb{E}\left[f_{ij}(x,t_i)\middle|\mathcal{D},\Theta^{(n-1)}\right] \tag{4.10}$$

$$\lambda_{ij}^{V(n)} = \min_x \mathbb{E}\left[f_{ij}(x,t_i)\middle|\mathcal{D},\Theta^{(n-1)}\right] \tag{4.11}$$

where $\mathbb{E}\left[f_{ij}(x,t_i)\middle|\mathcal{D},\Theta^{(n-1)}\right]$ is obtained in (4.9).

- **Step 2**: *sample* ${\sigma_G^2}^{(n)}$ *and* $\sigma_I^{2(n)}$ (*variances of the latent functions* f_{ij}^G *and* f_{ij}^I) *and* $\sigma_\varepsilon^{2(n)}$ (*measurement noise*), *and* ${\sigma_H^2}^{(n)}$ *and* ${\sigma_V^2}^{(n)}$ (*variance of the Gaussian process for the landmarks*) *and* ${\sigma_{\varepsilon H}^2}^{(n)}$ *and* ${\sigma_{\varepsilon V}^2}^{(n)}$ (*corresponding errors*) from the full conditional probabilistic distribution given the previously sampled parameters, by using the Metropolis-Hasting algorithm with the acceptance rates presented in Appendix E.2 and E.3 in the supplementary material.

Our method does not restrict the landmark only to the minimum points of signals. When a different definition of the landmarks is used, the required change

in the algorithm is minimal, remaining other parts the same: Only the Equations (4.10) and (4.11) need to be modified instead of using *min* function.

4.3.3 Functional data calibration

The calibration is formulated to infer the posterior distribution of t given a new functional sample $\mathbf{y}_{new}$ at predictors $\mathbf{x} \in \mathbb{R}^L$, i.e., $\mathbb{P}\left(t|\mathbf{y}_{new}, \mathbf{x}, \widehat{\Theta}\right)$, where $\widehat{\Theta}$ is the estimated parameter set from Section 4.3.2. More specifically, $\mathbb{P}\left(t|\mathbf{y}_{new}, \mathbf{x}, \widehat{\Theta}\right)$ is obtained by marginalizing the joint distribution $\mathbb{P}\left(t, \lambda^V_{new}, \lambda^H_{new}, |\mathbf{y}_{new}, \mathbf{x}, \widehat{\Theta}\right)$, where λ^V_{new} and λ^H_{new} are the landmarks for $\mathbf{y}_{new}$. The iterative cyclic sampling process to sample nth samples of $\left(t, \lambda^V_{new}, \lambda^H_{new}\right)$ is presented below.

- **Step 1**: *sample horizontal landmark* ${\lambda^H_{new}}^{(n)}$ *of signal* $\mathbf{y}_{new}$ from the full conditional distribution $\mathbb{P}\left(\lambda^H_{new}\middle|\mathbf{y}_{new}, \mathbf{x}, \widehat{\Theta}, t^{(n-1)}\right)$ by using the Metropolis-Hasting algorithm with the acceptance rates presented in Appendix F.1.1 in the supplementary material.

- **Step 2**: *sample vertical landmark* ${\lambda^V_{new}}^{(n)}$ *of signal* $\mathbf{y}_{new}$ from the normal distribution $\mathcal{N}\left({\mu_{\lambda^V_{new}}}^{(n)}, {\sigma^2_{\lambda^V_{new}}}^{(n)}\right)$ where

$$
{\sigma^2_{\lambda^V_{new}}}^{(n)} = \frac{1}{\mathbb{1}^{\mathsf{T}}_{[L]}\left(\mathbf{\Omega}^{(n-1)}_{\mathbf{y}_{new}}\right)^{-1}\mathbb{1}_{[L]}} + \frac{1}{\widehat{\mathbf{\Omega}}_{\lambda^V_{new}} - \widehat{\mathbf{\Omega}}_{\lambda^V_{new}\boldsymbol{\lambda}^V}\left(\widehat{\mathbf{\Omega}}_{\boldsymbol{\lambda}^V}\right)^{-1}\widehat{\mathbf{\Omega}}_{\boldsymbol{\lambda}^V\lambda^V_{new}}} \tag{4.12}
$$

and $\mu_{\lambda^V_{new}}{}^{(n)}$ is

$$\left\{ \mathbb{1}_{[L]}^{\mathsf{T}} \left(\mathbf{y}_{new} - \lambda^V_{new}{}^{(n-1)} \right) \left(\mathbf{\Omega}^{(n-1)}_{\mathbf{y}_{new}} \right)^{-1} + \frac{\widehat{\mathbf{\Omega}}_{\lambda^V_{new}\boldsymbol{\lambda}^V} \left(\widehat{\mathbf{\Omega}}_{\boldsymbol{\lambda}^V} \right)^{-1} \widehat{\boldsymbol{\lambda}^V}}{\widehat{\mathbf{\Omega}}_{\lambda^V_{new}} - \widehat{\mathbf{\Omega}}_{\lambda^V_{new}\boldsymbol{\lambda}^V} \left(\widehat{\mathbf{\Omega}}_{\boldsymbol{\lambda}^V} \right)^{-1} \widehat{\mathbf{\Omega}}_{\boldsymbol{\lambda}^V \lambda^V_{new}}} \right\} \sigma^2_{\lambda^V_{new}}{}^{(n)} \tag{4.13}$$

where $\mathbf{\Omega}^{(n-1)}_{\mathbf{y}_{new}}$ is the covariance matrix of $\mathbf{y}_{new}$ calculated by (4.6) with $\lambda^H_{new}{}^{(n-1)}$ and $\widehat{\Theta}$, and $\widehat{\mathbf{\Omega}}_{\boldsymbol{\lambda}^V} = \widehat{\sigma^2_V} k_V(\boldsymbol{t}_\lambda, \boldsymbol{t}_\lambda) + \widehat{\sigma^2_{\varepsilon^V}} I_{[MN]}$, $\widehat{\mathbf{\Omega}}_{\lambda^V_{new}\boldsymbol{\lambda}^V} = \widehat{\sigma^2_V} k_V(t, \boldsymbol{t}_\lambda)$, and $\widehat{\mathbf{\Omega}}_{\lambda_{new}} = \widehat{\sigma^2_V} k_V(t,t) + \widehat{\sigma^2_{\varepsilon^V}}$.

- **Step 3**: *sample target variable* $t^{(n)}$ from $\mathbb{P}\left(t|\lambda^V_{new}{}^{(n)}, \lambda^H_{new}{}^{(n)}, \mathbf{y}_{new}, \mathbf{x}, \widehat{\Theta}\right)$ defined in (4.14) by using the Metropolis-Hasting algorithm.

The detailed derivation of the acceptance rates for the Metropolis-Hasting algorithms (step 1) and the posterior distribution $\mathcal{N}\left(\mu_{\lambda^V_{new}}, \sigma^2_{\lambda^V_{new}}\right)$ (step 2) are presented in Appendix F in the supplementary material. In step 3, the full conditional distribution of the target variable is not a trivial probability distribution, and we decompose it as follows:

$$\mathbb{P}\left(t|\lambda^V_{new}, \lambda^H_{new}, \mathbf{y}_{new}, \mathbf{x}, \widehat{\Theta}\right) \propto \pi(t)\, \pi_H\left(t|\lambda^H_{new}\right) \pi_V\left(t|\lambda^V_{new}\right) \pi_{\mathbf{y}_{new}}\left(t|\mathbf{y}_{new}, \lambda^H_{new}, \lambda^V_{new}\right) \tag{4.14}$$

where $\pi(t)$ is the prior distribution of t, and $\pi_H\left(t|\lambda^H_{new}\right) = \mathbb{P}\left(\lambda^H_{new}\middle|\widehat{\boldsymbol{\lambda}^H}, \boldsymbol{t}_\lambda, t\right)$, $\pi_V\left(t|\lambda^V_{new}\right) = \mathbb{P}\left(\lambda^V_{new}\middle|\widehat{\boldsymbol{\lambda}^V}, \boldsymbol{t}_\lambda, t\right)$, and $\pi_{\mathbf{y}_{new}}\left(t|\mathbf{y}_{new}, \lambda^H_{new}, \lambda^V_{new}\right) = \mathbb{P}\left(\mathbf{y}_{new}\middle|\mathbf{x}, \widehat{\Theta}, \lambda^H_{new}, \lambda^V_{new}, t\right)$ are the likelihood of t given λ^H_{new}, λ^V_{new}, and $\mathbf{y}_{new}$, respectively. Specifically, π_H and π_V are the posterior predictions of Gaussian process for λ^H_{new} and λ^V_{new}, respectively,

given the estimated landmarks ($\widehat{\boldsymbol{\lambda}^H}$ and $\widehat{\boldsymbol{\lambda}^V}$) of the training signals. $\pi_{\mathbf{y}_{new}}$ is the multivariate normal distribution with mean $\lambda^V_{new}\mathbb{1}_{[L]}$ and covariance $\boldsymbol{\Omega}_{\mathbf{y}_{new}}$ defined in (4.6) given $\widehat{\Theta}$. This decomposition allows us to fully utilize the information on the random shifts and shapes during the calibration process. This is illustrated in Figure 4.3.1, a new sampled signal is evaluated by the three likelihoods: horizontal landmark, vertical landmark, and shape. The horizontal and vertical landmarks (λ^H_{new} and λ^V_{new}) are evaluated by the likelihood π_V and π_H over the target variable. Once conditioning on the landmarks, every signal is aligned by overlapping its landmark with the origin, and the shapes of the aligned signals are evaluated by the likelihood $\pi_{\mathbf{y}_{new}}$. Then, by aggregating all three likelihoods with the prior probability of the target variable, the posterior probability is obtained. In this manner, this process evaluates the new signal $\mathbf{y}_{new}$ by all three aspects, horizontal and vertical locations and shape in terms of t. This detailed probabilistic model structure improves calibration accuracy. For example, suppose we have functional signals with large vertical shifting variations and small horizontal shifting variations as in Figure 4.3.1. Vertical landmarks of signals would provide weak information on the target variable, which produce widely distributed likelihoods π_V over t. In contrast, horizontal landmarks will provide strong information on the target variable and produce densely distributed likelihoods π_H of t around the true target variable. Then, (4.14) reflects the strengths of the pieces of evidence during calibration by multiplying the three likelihoods. The resulting posterior distribution would be dense around the true target variable by taking advantage of the strong information.

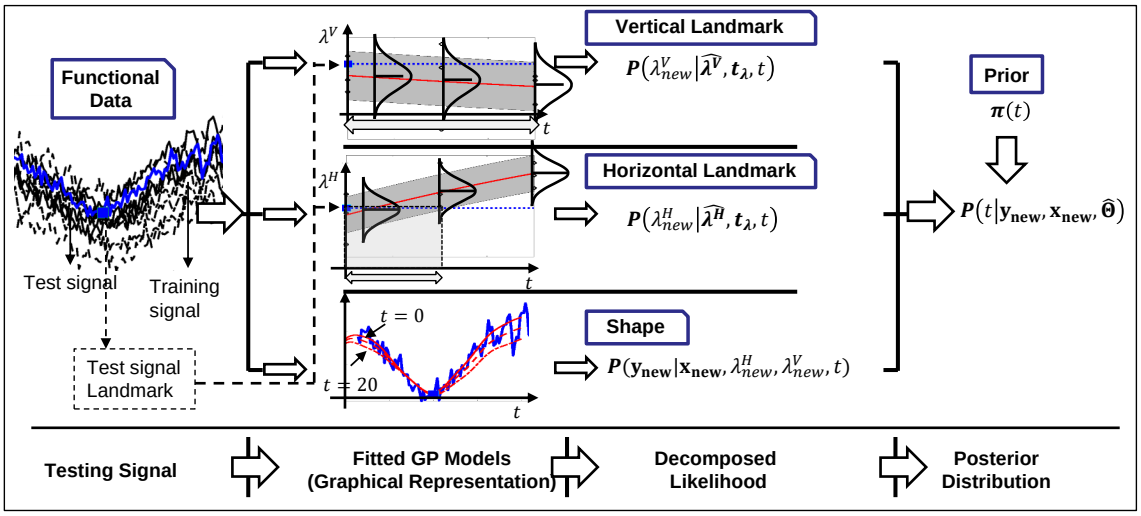


Figure 4.3.1: Illustration of the calibration process

4.4 Numerical study

The performance of our proposed calibration method with the landmark-embedded Gaussian process model is validated by extensive numerical studies. In Section 4.4.1, the evaluation criteria and the benchmark settings are described. The results of the numerical studies are reported in Section 4.4.2.

4.4.1 Evaluation criteria and benchmark settings

We conducted numerical studies with multiple different scenarios. For each scenario, ten different training sets ($D = 10$) are generated. Each training set includes four groups ($M = 4$, $(t_1, t_2, t_3, t_4) = (0, 10, 20, 30)$), and ten functional signals ($N = 10$) for each group. Each functional signal $\mathbf{y}_{ij}$ is measured at a sequence of predictors from -100 to 100 equally spaced by 4 ($L = 101$, $\mathbf{x} = (-100, -96, \ldots, 100)^{\mathsf{T}}$),

and the signals, predictors, and target variable t_i are observed. A test set is generated for each dataset. In each test set, ten test signals ($R = 10$) are created at each of nine values of the target variable (t_t^{test}) from 2 to 26 with increment of 3 ($T = 9$, $(t_1^{test}, t_2^{test}, \ldots, t_9^{test}) = (2, 5, \ldots, 26)$). In total, 900 test signals are calibrated ($(D = 10$ datasets$) \times (R = 10$ test signals at each group$) \times (T = 9$ test groups$)$) in each scenario.

The proposed calibration method is based on MCMC sampling, and the calibration result is a histogram of the sampled target variables. As a result, we use the median value (point estimate) and the 90%-posterior interval (interval estimate) of the histogram to evaluate the calibration performance. The overall accuracy of the prediction is evaluated by the *point prediction* (i.e., one single predicted value) More specifically, we assess the performances of point estimates in terms of accuracy and performance stability. The distribution of the absolute prediction error (APE) of the point estimates

$$APE_{dtr}^{P} = \left|\widehat{t}_{dtr} - t_t^{test}\right| \tag{4.15}$$

will be reported by box plots, where $\widehat{t}_{dtr}$ is the point estimate from rth test signal from the target variable t_t^{test} with dth training set. Notice that the contamination values used for the test signals are selected between 0 and C, where C is defined by the largest value of the target variable used in the training set; therefore, a reasonably calibrated point estimate would have an APE less than C. We define a set of reasonable point estimates' indexes $\mathcal{R} = \left\{(d, t, r)\,|APE_{dtr}^{P} < C\right\}$. Then, we report the mean squared prediction error (MSPE) only with the predicted point

estimate $\widehat{t}_{dtr}$ whose (d, t, r) is in $\mathcal{R}$ to evaluate the overall calibration accuracy after removing the outliers' influence.

$$MSPE = \frac{\sum_{(d,t,r)\in\mathcal{R}} \left(\widehat{t}_{dtr} - t_t^{test}\right)^2}{|\mathcal{R}|} \tag{4.16}$$

Furthermore, the proportion of the number of the calibrated results that failed to be included in $\mathcal{R}$ over the total number of calibrated results, denoted by %Fail, is also reported to show performance stability.

$$\%Fail = 1 - \frac{|\mathcal{R}|}{D \times T \times R} \tag{4.17}$$

A lower %Fail indicates higher performance stability.

We also report the accuracy of the interval estimates. In Bayesian statistics, posterior interval (also known as credible interval) is often used for interval estimates (Gelman et al., 2013a). $\alpha\%$-posterior interval, or briefly $\alpha\%$-interval, of a target variable is defined by the range containing the true target variable value with the probability of $\alpha\%$. For each signal calibration, we calculate the 90%-interval, defined by the range between 5th and 95th percentiles of the posterior target variable values sampled from the MCMC. The accuracy of the interval estimates is evaluated by the accuracy of the proportion that the predicted intervals contain the true target values. Ideally, 90% of the '90%-intervals' should include the true values. Accordingly, the errors are calculated as follows: We counted the number of intervals that include the true target variable values for each test dataset; the proportions of the counts that the predicted intervals include the true values (over

the total $T \times R$ times calibrations) are calculated for each dataset.

$$\widehat{P}_d = \frac{\sum_{t=1}^{T} \sum_{r=1}^{R} \mathbb{1}\left(t_t^{test} \in \widehat{\mathcal{I}}_{dtr}\right)}{T \times R} \tag{4.18}$$

where $\widehat{\mathcal{I}}_{dtr}$ denotes the interval estimated from rth test signal from the target variable t_t^{test} with dth training set, and $\mathbb{1}(\cdot)$ is one if it satisfies the condition in the bracket or zero. Then, the absolute prediction errors as the difference between the calculated proportion and the true inclusion probability, 0.9, for each dataset

$$APE_d^I = |\widehat{P}_d - 0.9| \tag{4.19}$$

are reported in box plots.

We benchmark the performance of our proposed calibration method, namely landmark-embedded Gaussian process, denoted by *LEGP*, against other three alternative models. The kernel functions and their hyperparameters that we used are presented in Appendix A in the supplementary material.

- *LR*$(\cdot)$: two-steps landmark registration approaches; **Step 1**: landmark registration. landmarks are obtained from the minimum point of the posterior mean function of the individually fitted Gaussian process $f_{ij}^I(x) \sim \mathcal{GP}(0, \sigma_I^2 k_I(x,x))$, and the signals are aligned by overlapping the landmarks. **Step 2**: calibration. a conventional Gaussian process is fitted to the aligned functional data, and the target variable is inferred from a new sample signal based on the fitted Gaussian process model. Notice that these methods do not consider the same-level variations of the functional signal shapes. Two types of land-

mark registration approach methods are considered with different calibration methods in the second step:

  - *LR*(*Aln*): calibration is conducted only with the Gaussian process model $f^G(x,t) \sim \mathcal{GP}(0, \sigma_G^2 k_G((x,t),(x,t)))$ fitted to the aligned functional data. In this method, the location information is not used in calibration. To be specific, in this manner, we aligned all the signals in the training dataset (denoted by $\mathcal{D}'$) and test signals (denoted by $\mathbf{y}'_{new}$) by overlapping the landmarks to the origin. $\mathbb{P}(t|\mathbf{y}'_{new}, \mathcal{D}')$ is estimated by the likelihood function $\mathbb{P}(\mathbf{y}'_{new}|t, \mathcal{D}')$.
  - *LR*(*Lm*): calibration is conducted with both the Gaussian process models of the aligned signals $f^G(x,t) \sim \mathcal{GP}(0, \sigma_G^2 k_G((x,t),(x,t)))$ and landmarks $\lambda^H \sim \mathcal{GP}(0, \sigma_H^2 k_H(t,t))$ and $\lambda^V \sim \mathcal{GP}(0, \sigma_V^2 k_V(t,t))$. The calibration result
    $\mathbb{P}\left(t|\mathbf{y}'_{new}, \mathcal{D}', \lambda^H_{new}, \lambda^V_{new}, \boldsymbol{\lambda}\right)$ is inferred by the likelihood function decomposed by (4.14). This calibration method assumes that the landmarks and aligned signals are given as fixed predictors, and the shape and location of the test signal are considered independently in model fitting and calibration, while our proposed method considers the estimated landmark distribution and integrates the estimation uncertainty into the calibration step in a unified manner.

- *FLR*: functional linear regression that models the concentration as a scalar response variable and the sensing signals as the functional predictors (Ramsay

and Silverman, 2005). B-spline basis is used for the functional predictors, and the number of basis functions and the penalty variable are selected by the cross-validation. The prediction interval is obtained by the parameter bootstrapping with 1,000 samples. Functional linear regression assumes a fixed population model to represent the relationship between the scalar variable and the functional signals. Hence, the random variations of the functional signals are not addressed with this method.

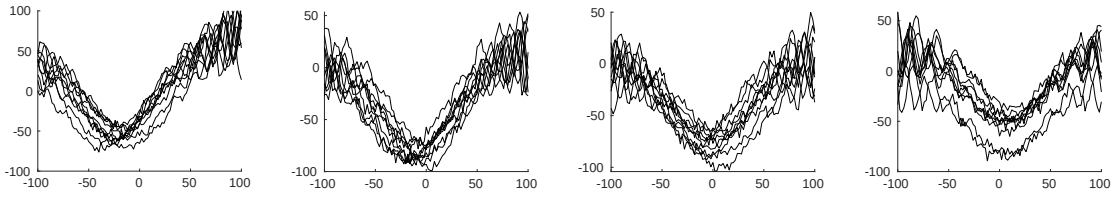
4.4.2 Numerical study result

The proposed model and the four benchmarks are compared in this section using numerical simulations. The random signal $\mathbf{y}_{ij} = (y_{ij1}, \ldots, y_{ijL})^{\mathsf{T}}$ is created from jth sensor at t_i as follows.

$$y_{ijl} = (t_i - 100)\left[\exp\left(-\left\{\left(1+\delta_{ij}^{S}/50\right)\left(x_l - \delta_{ij}^{H}\right)\right\}^2/2500\right) + 5\exp\left\{\left(x_l - 5t_i\right)/50\right\}\right] + \sin\left((x_l - \delta_{ij}^{H})/10\pi\right)\left\{\left(x_l - \delta_{ij}^{H}\right)/20\right\}^2 + \delta_{ij}^{V} + \varepsilon_{ijl} \tag{4.20}$$

The general shapes of the simulated signals are illustrated in Figure 4.4.1. The functional form of the signals is assumed unknown to any of the used methods. The signals have four characteristics: 1) non-trivial shapes for modeling (asymmetric and wavy), 2) random shifting (horizontally and vertically) and random shape presented in each individual signal, 3) common trends of the signals' random shifting and shapes across the target variables, and 4) measurement noises. The

random features of horizontal and vertical shifts, signals' shape, and noises are denoted by $\delta_{ij}^{S} \sim \mathcal{N}\left(0, \sigma_S^2\right)$ and $\delta_{ij}^{H} \sim \mathcal{N}\left(t_i - 20, \sigma_H^2\right)$ and $\delta_{ij}^{V} \sim \mathcal{N}\left(0.1\left(t_i - 20\right)^2, \sigma_V^2\right)$ and $\varepsilon \sim \mathcal{N}\left(0, \sigma_\varepsilon^2\right)$, respectively. σ_H^2, σ_V^2, σ_S^2, and σ_ε^2 parameterize the magnitude of the randomness in the horizontal shifts, vertical shifts, shape of the signals, and noises, respectively. We consider five different cases to validate the benefits of our proposed model, where each case has different values for the parameters of σ_H^2, σ_V^2, σ_S^2, σ_ε^2 and N. The parameters used in each case are summarized in Table 4.4.1.



(a) $t = 0$ (b) $t = 10$ (c) $t = 20$ (d) $t = 30$

Figure 4.4.1: Simulated signals with $t = 0$, $t = 10$, $t = 20$, $t = 30$ for Case I.

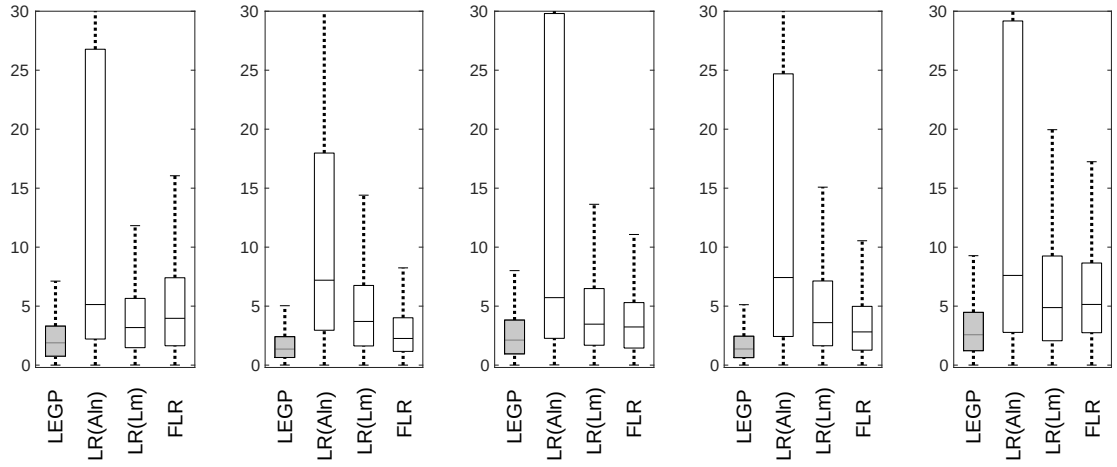
For each of the training dataset, the statistical parameters are first estimated. Then, the unknown target variable is inferred for each test signal based on the estimated model parameters. The initial parameters for calibrating the test signal are set empirically according to the training model estimates. Due to the customized initial values, within 100 burn-in, the sample parameters tended to be stationary. The samples from the burn-in period were disregarded. After burning-in, Q =300 samples are collected from the posterior distribution of the target variable. The convergence of the MCMC sampling was checked both in parameter estimation and calibration to determine if the iteration process can be completed. For the convergence threshold ratio, 0.05 was used.

Table 4.4.1: Five designed cases considered in the numerical study

Case	σ_H	σ_V	σ_S	σ_ε	N	Comment
I	7	10	5	3	10	Baseline: all the randomness are included
II	**0**	**20**	5	3	10	No random horizontal but large vertical random shifting
III	7	10	5	**10**	10	Large noises
IV	7	10	5	3	**3**	Small sample size
V	**10**	**20**	**0**	3	10	Large location variations with no shape variations

Figure 4.4.2 presents the distribution of APEs (defined in (4.15)), and Table 4.4.2 summarizes 1) MSPE calculated only with the predicted point estimates whose APEs are less than 30 (defined in (4.16)) and 2) the proportion of the number of calibrated point estimates whose APEs are larger than 30 (defined in (4.17)). The results indicate that our proposed model clearly outperforms the others. In every case, our proposed model shows the best accuracy with the smallest MSPE and the best stability (zero number of the APEs larger than 30 in the table). The results confirm that our proposed method effectively utilizes the information of the signals' location and shape. The MSPE of *LEGP* from Case II is much smaller than its MSPE from Case I and than the MSPEs from the other methods in Case II. Notice that, in Case II, the relationship between signals' horizontal locations and target variable involves no randomness. By leveraging the location information over the target variable, *LEGP* facilitates the accurate inference of the target variable values even if there is a significant random vertical shifting in signals. Our proposed model also enables accurate calibration with a smaller sample size (Case IV) while all the other methods show quite bad performances in both accuracy and performance stability. The outperformance of $LR(Lm)$ over $LR(Aln)$ in accuracy and performance stability of point estimates confirms the importance of utilizing the signals' shifted

location in calibration. *FLR* performed quite well in the point estimates although it noticeably underperformed *LEGP*. In most cases (Case II-V), *FLR* outperforms $LR(Lm)$ and $LR(Aln)$.



(a) Case I (b) Case II (c) Case III (d) Case IV (e) Case V

Figure 4.4.2: Absolute prediction error with point estimated target variable for given five cases.

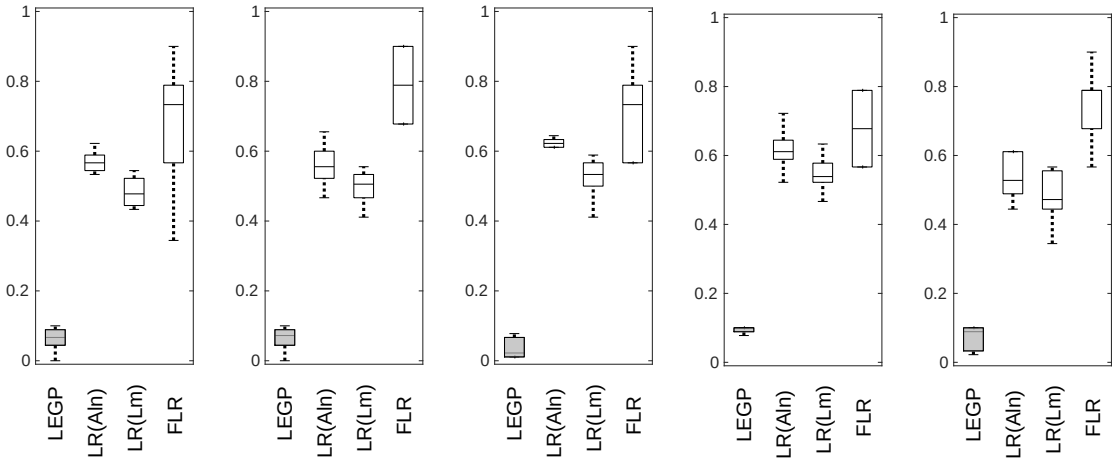
Table 4.4.2: The MSPE from the calibrated point estimates whose absolute prediction errors are less than 30 (defined in (4.16)) and the proportion of the number of the calibrated point estimates whose absolute prediction errors are larger than 30 in bracket (defined in (4.17)).

	LEGP	LR(Aln)	LR(Lm)	FLR
Case I	10 (0%)	83.4 (22.2%)	31 (1.4%)	56 (1%)
Case II	6.5 (0%)	102.1 (16.9%)	47.3 (1.8%)	14.5 (0%)
Case III	12.2 (0%)	82.4 (16.9%)	38.9 (1.6%)	29.3 (0%)
Case IV	4.9 (0%)	135.6 (16.4%)	54.5 (2.6%)	36 (0.1%)
Case V	16.6 (0%)	106.4 (16.4%)	67.4 (5%)	65.8 (1.7%)

Figure 4.4.3 exhibits the absolute errors of the interval estimates (defined in (4.19)) from all the models. The interval estimates of our proposed method signifi-

cantly more accurate than those from the reference methods. The absolute errors of the interval estimates from *LEGP* are between 0 and 0.1 in every case. On the other hand, the median errors of the two-steps approaches ($LR(Aln)$ and $LR(Lm)$) are between 0.45 and 0.65 with ranges between 0.1 and 0.2. The absolute errors from *FLR* are the worst: the medians of the errors are between 0.7 and 0.8 and their ranges are longer than the others, with between 0.2 and 0.6. Such inaccuracy of the interval estimates from *FLR* is because *FLR* does not properly characterize the randomness of the given functional data. *FLR* assumes a 'population-level' fixed relationship between the response and functional predictors; therefore, it cannot precisely account for the randomness involved in the signals. The imprecise modeling of the randomness in the signals resulted in overconfidence in the interval estimates and failed in including the true target variable values in most cases. A brief discussion about the overconfidence is provided in Appendix H.

Overall, *LR* models showed poor performance in both point and interval predictions. The accuracy of the interval estimates is distinctly lower than our proposed *LEGP* model due to the distinct disadvantages of *LR* models over *LEGP* model: First, in *LR* models, the landmarks are less precise because the GP models used for landmark estimations are different from the GP models used for calibration, which incurs discrepancy. Also, individually fitted GP models in landmark estimations cannot consider the correlation between signals, which provide less precise fitting and accordingly less precise landmarks. Second, *LR* models use the conventional GP models to the aligned signals and for calibration, which do not consider the individual shape differences of signals. The individual random shape differences



(a) Case I (b) Case II (c) Case III (d) Case IV (e) Case V

Figure 4.4.3: Absolute prediction error of the 90%-interval estimates for each training set, defined in (4.19). For each dataset, we calculate the proportion of the number of tests of which estimated intervals cover the true target variable values over the number of total calibrated results (defined in (4.18)), and the absolute difference between the calculated proportion and 0.9 is reported.

are incorrectly accounted for by the independent measurement errors. Third, uncertainties in the estimated landmarks and the functional signals are not precisely quantified because landmarks are assumed to be determined as constants after the landmark estimation process. In *LEGP*, on the other hand, landmarks are included as parameters in the model, and all the parameters including the landmarks are estimated simultaneously by a Bayesian method. The results confirms that this integrative modeling approach is very effective, by which we can accurately estimate the landmarks as well as other parameters.

4.5 Case study

In this section, we demonstrate the functional data calibration based on the GFET sensor signal data, which have been illustrated in Figure 4.1.2. The GFET sensor signals are obtained from a complex physical model of the GFET sensor (Wang et al., 2021a), which has been validated using real sensing data. In the dataset, the drain-source currents are obtained corresponding to the gate voltages with given sensor physical and geometrical parameter settings, including the length of the graphene channel and drain-source voltages. The gate voltages are equally spaced from -40V to 60V by 0.5V. We use the sensor physical and geometrical settings obtained from Wang et al. (2021a). Because the random variations exist in sensor settings, we can observe significant variations in the sensing signal as seen in Figure 4.1.2.

As described earlier in the introduction section, the training set consists of three groups of contamination levels: 0, 10, 20 ppb. In each training set, ten signals are included for each group of contamination levels; therefore, in total, thirty functional signals are in a training set. The test set consists of ten groups of contamination levels from 1 ppb to 19 ppb by the increment of 2 ppb, and each group has ten signals. Hence, each test set has a hundred signals. The proposed landmark-embedded Gaussian process model is first estimated using the training dataset, and then the calibration is conducted with every test signal using the estimated model. To make reliable performance evaluation, we repeat 10 times of model training and signal calibrations. Thus, in total, we have 10 training datasets and corresponding 10 testing datasets. The concentration levels in the training sets are known. On

the other hand, the concentration levels in the test sets are assumed unknown in calibration, and the calibration results are evaluated by being compared with the true concentration levels.

The evaluation criteria and benchmarks used in the numerical studies are also used in the case study. Figure 4.5.1() shows the distribution of the APEs of the point estimates (defined in (4.15)) in box plots where only APEs less than 20 are displayed. Table 4.5.1 reports 1) MSPE calculated with the calibrated point estimates whose APEs are less than 20 for the accuracy (defined in (4.16)) , and 2) the proportion of the number of the calibrated point estimates whose APEs are larger than 20 for the stability (defined in (4.17)). Figure 4.5.1(a) is the box plot of the absolute errors of the posterior interval coverage, defined in (4.19). Similar to the numerical study results in Section 4.4, the results indicate that our proposed model (*LEGP*) clearly outperforms the reference methods. *LEGP* shows the smallest median and variance of the prediction errors with respect to both point estimates and interval estimates. Especially, the interval estimates of *LEGP* are significantly more accurate than those of the reference methods. The median of the absolute error of the interval estimates' coverages is less than 0.1, on the other hand, those from the reference methods are larger than 0.3 with large variations. Although the point estimates from *FLR* are quite accurate, their interval estimates are the worst with median of 0.8.

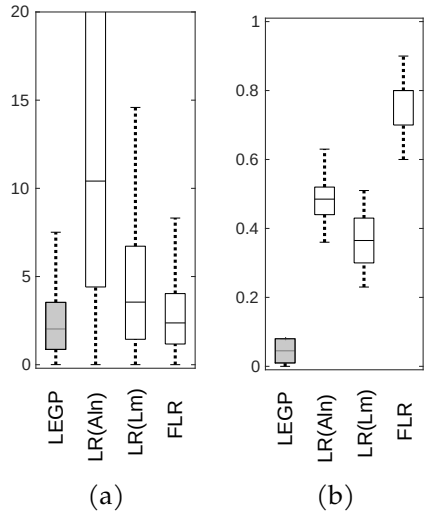


Figure 4.5.1: Absolute prediction errors of the (a) point estimates (defined in Equation (4.15)) and (b) 90%-posterior interval in the case studies (defined in Equation (4.19)).

Table 4.5.1: Summary results of the case study. MSPE is calculated from the calibrated point estimates whose APEs are smaller than 20. %Fail is the proportion of the calibrated point estimates whose APEs are larger than 20 (Equation (4.17)).

	LEGP	LR(Aln)	LR(Lm)	FLR
MSPE(%Fail)	10.8(0%)	90.9(27.9%)	40.3(3.9%)	13.8(0%)

4.6 Conclusion

In this chapter, we establish a data-driven non-parametric statistical functional model, namely landmark-embedded Gaussian process to account for the functional signals with four types of variations: same-level shape, same-level location, across-level shape, and across-level location variations. The individual random shifts of the functional signals are characterized by the landmarks embedded in the Gaussian

process model. The specially tailored structure with two latent functions of the Gaussian processes is constructed to account for the same-level variations of signals while capturing the across-level variations shared among signals. Our proposed method is a unified model where the alignment and Gaussian process parameter estimations are conducted simultaneously. The landmarks are estimated from the fitted Gaussian process, not from the noisy raw data. Our proposed model provides a comprehensive view over all the observed signals, automatically accounting for their four random variations. Therefore, we can fully utilize the available information for the alignment and statistical calibration. We developed Markov chain Monte Carlo (MCMC) algorithms to estimate the model parameters and infer the probability distribution of the unknown target variable. The performance of the calibration is evaluated in the numerical studies with simulated data and a case study with GFET sensor signal data.

Statistical functional data calibration framework in this chapter can be applied in various applications besides sensor calibration, such as profile data monitoring and diagnosis from the manufacturing domain. In many manufacturing systems, the machine operating conditions and environments change dynamically. The uncertainties in the environments often induce random shifts and shape changes in the functional signals (Lei et al., 2010a; Zhang et al., 2018). The proposed framework can be applied to those profile data to estimate the unobserved working condition. The estimated working condition can be used for root-cause identification of the manufacturing process and for anomaly detection.

The proposed method successfully account for the random shifts and random

shapes of the functional data in calibration by using the landmark defined by the user. We can perform the prediction in the calibration stage fairly quickly despite the higher computational burden on the model parameters estimation stage. In order to further improve the scalability, one can use one of the approximation techniques to make the Gaussian process parameters estimation scalable. Many approximation techniques have been developed for scalable Gaussian process, for example, by introducing inducing points in the functional data (Herbrich et al., 2003). GPyTorch uses graphics processing unit (GPU) to make the Gaussian process fitting and prediction further scalable (Gardner et al., 2018). There also have been efforts to reduce the computational burdens in hyperparameter estimation of the Gaussian processes (Chen et al., 2020). In addition, the MCMC sampling procedure in the parameters estimation and calibration process in this work can also be easily parallelized with multiple chains to speed up the Gaussian process estimation time.

We would like to point out that to utilize the proposed method, we need to pre-specify certain landmarks for the signal. These landmarks can often be identified based on the physical meanings of the signal, for example, GFET sensor signals (Wang et al., 2016) and tonnage signals (Jin and Shi, 1999a). However, there may be some cases where it is hard to select a meaningful landmark for alignment. In such case, the proposed method cannot be applied.

One interesting future research direction would be an algorithmic method of landmark specification: an algorithm that can automatically define good landmarks by investigating the feature of the given functional data. Such algorithm might be especially useful when we need to choose the best landmark among multiple

candidates, such as multiple local maxima and minima. One possible approach is a feature selection method.

Another interesting future research direction is the application on censored functional data. When the functional data are censored, a simple approach is to discard the censored functional data and use the remained data for parameter estimation. This treatment is reasonable only when the censored data size is small. When we have a large amount of the data being censored, we may not simply ignore the censored data. One possible approach is to infer and recover the full shape of such functional signals based on the historical data and use our proposed method. To deal with censored functional data, one may build a complete model that can simultaneously recover the unseen shapes of the functional signals as well as their landmarks for calibration.

APPENDIX

4.A Prior Specification

4.A.1 Kernel functions

Four kernel functions are presented in this article for the Gaussian processes: $k_I(x, x')$, $k_G((x, t), (x', t'))$, $k_H(t, t')$, and $k_V(t, t')$. We used the Gaussian kernel for all the kernel functions because the Gaussian kernel is flexible and able to capture various features of the functional data Kontar et al. (2018). A univariate Gaussian kernel is defined as follows.

$$k(x, x') = \exp\left(-\frac{(x - x')^2}{2l_1^2}\right) \tag{4.21}$$

For bivariate Gaussian processes, we used

$$k((x, t), (x', t')) = \exp\left(-\frac{(x - x')^2}{2l_1^2} - \frac{(t - t')^2}{2l_2^2}\right) \tag{4.22}$$

where l_1 and l_2 are the length-scale parameters. The length-scale parameters are tuning parameters that are specified by users according to the characteristics of the signals. The length-scale parameters determine the length of the wiggles of the prior functional curve. In this article, $k(\mathbf{x}, \mathbf{x}')$ is defined as a matrix whose (l, l')-component is $k(x_l, x'_{l'})$ where x_l and $x'_{l'}$ are lth and l'th component of $\mathbf{x}$ and $\mathbf{x}'$, respectively. $k((\mathbf{x}, t), (\mathbf{x}', t'))$ is defined as a matrix whose (l, l')-component is $k((x_l, t), (x'_{l'}, t'))$.

4.A.2 Prior Distributions

For variance of the normal distributions, people often use the inverse gamma prior distribution.

$$\sigma^2 \sim \pi(\alpha, \beta) \tag{4.23}$$

The α and β are the hyper-parameters that are pre-specified according to the empirical domain knowledge about the variance. They can be selected so that their mean and variance are equals to those we already know as prior information. Another way of selecting these hyper-parameters are setting them zero and make the prior uninformative. This implies that the prior probability densities of any values are the same within its support. The uninformative priors are widely used if specific knowledge about parameters is not available, or ones do not want to put some specific knowledge so that let the data speak. In this work, we used noninformative priors for all the variances (equivalently, $\alpha = 0, \beta = 0$ of the inverse gamma prior distribution).

For the target variable, the noninformative prior with positive values is used. Specifically, for all $t > 0$,

$$\pi(t) \propto 1 \tag{4.24}$$

4.A.3 Lengthscales for the Gaussian Kernels

Lengthscale of the Gaussian kernels can be selected based on our prior knowledge on the functional signals. The lengthscale can be explained by how far we need to move (along a particular axis) in input space for the function values to become uncorrelated (Williams and Rasmussen, 2006). In the application of GFET sensors, we set the lengthscale l_I for f^I as 10. This is because the 10 voltages range around the turning point of each functional signal can provide detailed current readings for describing the function turning behavior. The turning point is an important characteristic in our model, and facilitate strong correlation around the turning point can help remove noise in identifying the landmark locations. Based on the same intuition, we set the l_G for f^G as 40. This is because the global lengthscale should cover the turning point behavior in not only the each individual signal, but all the functional signals under different t's. The lengthscale for t is also set as 40, and it facilitates the correlation among contamination intensities within 40 ppb difference. As a result, the l_H and l_V are also set as 40 to keep consistent in describing the correlation impacts resulting from t.

Please note that the lengthscales l_I, l_G, l_H and l_V can also be estimated by maximizing the log likelihood function of the Gaussian process model. However, large scale Gaussian process often suffers numerical issues in parameter estimation (Yi et al., 2011). The lengthscales can also be treated as unknown random variables and integrated into the MCMC parameter estimation procedure we introduced in Section 3. However, this will add another layer of the MCMC procedure and make the whole algorithm inefficient.

In summary, below are the lengthscales used in this work for both numerical and case studies.

- $l_I = 10$ is used for the lengthscale of kernel $k_I(x, x')$.
- $l_G = (40, 40)$ is used for the lengthscale for kernel $k_G((x, t), (x', t'))$.
- $l_H = 40$ is used for the lengthscale for kernel $k_H(t, t')$.
- $l_V = 40$ is used for the lengthscale for kernel $k_V(t, t')$.

4.B Landmark specification

There may be an identifiability issue for the linear functional signals when we have both horizontal and vertical landmarks in the proposed model because we cannot tell if a linear functional signal shifted horizontally or vertically: vertical shift ($(y-d_1) = ax+b$) always can be represented by the horizontal shift ($y = a(x-d_2)+b$) with some d_2. Regardless, our proposed method can be used for the linear functional signals by defining different landmarks. For example, we can define the landmark as a point where the functions passing T as landmark, and then, the landmark will be defined as $\lambda_{ij}^{H(n)} = \arg\min_x \left| \mathbb{E}\left[f_{ij}(x, t_i) \middle| \mathcal{D}, \Theta^{(n-1)} \right] - T \right|$ and $\lambda_{ij}^{V(n)} = T$; these will be substituted for the Equations (4.10) and (4.11) in the main text instead of using *min* function; this is the only required modification in the method in response to different landmark specification. Under this definition of landmark, we choose a fixed vertical landmark, and the horizontal shifts of the linear function are accounted for by the horizontal landmarks.

4.C Metropolis-Hasting Algorithm

In this article, the Metropolis-Hasting algorithm is used for parameter estimation and calibration. Here, we briefly describe the algorithm.

4.C.1 Proposal Distribution

The Metropolis-Hasting algorithm draws samples from proposal distributions. The normal distribution is widely used for the proposal distribution to sample the random variables whose supports are the real line. The random variables that only have positive values also can be sampled from the normal distribution after log transformation. We used the normal distribution for the proposal distributions.

$$\tilde{\theta} \sim \mathcal{N}\left(\theta^{(n-1)}, \sigma_{\mathcal{J}}^2\right) \tag{4.25}$$

The parameters that only can have positive values (e.g. variance) are sampled after log transformation. For instance, log-variance is sampled from the normal distribution.

$$\log\left(\tilde{\sigma^2}\right) \sim \mathcal{N}\left(\log\left(\sigma^{2^{(n-1)}}\right), \sigma_{\mathcal{J}}^2\right) \tag{4.26}$$

where $\sigma_{\mathcal{J}}^2$ is the variance of the proposal distribution. The proposal distribution is denoted by $\mathcal{J}$ hereafter, therefore, for example, the proposal distribution for σ^2 above is $\mathcal{J}\left(\tilde{\sigma^2}\middle|\sigma^{2^{(n-1)}}\right) = \mathcal{N}\left(\log\left(\sigma^{2^{(n-1)}}\right), \sigma_{\mathcal{J}}^2\right)$

4.C.2 Sampling Process of the Metropolis-Hasting algorithm

The process of the Metropolis-Hasting algorithm is briefly explained. Suppose we want to sample an arbitrary parameter θ from the Metropolis-Hasting algorithm. Given the proposal probability distribution $\mathcal{J}\left(\tilde{\theta}\middle|\theta^{(n-1)}\right)$ and the acceptance rate $\mathcal{A}\left(\tilde{\theta},\theta^{(n-1)}\right)$, the Metropolis-Hasting algorithm is conducted as follows.

1. Draw $\tilde{\theta}\sim\mathcal{J}\left(\tilde{\theta}\middle|\theta^{(n-1)}\right)$
2. Draw $u\sim\mathcal{U}\left(0,1\right)$
3. If $u\leq\mathcal{A}\left(\tilde{\theta},\theta^{(n-1)}\right)$, $\theta^{(n)}=\tilde{\theta}$, or $\theta^{(n)}=\theta^{(n-1)}$

where $\mathcal{U}\left(0,1\right)$ is the uniform distribution between zero and one, and the acceptance rate is calculated by

$$\mathcal{A}\left(\tilde{\theta},\theta^{(n-1)}\right)=\min\left(1,\frac{\mathbb{P}\left(\tilde{\theta}\right)\mathcal{J}\left(\theta^{(n-1)}\middle|\tilde{\theta}\right)}{\mathbb{P}\left(\theta^{(n-1)}\right)\mathcal{J}\left(\tilde{\theta}\middle|\theta^{(n-1)}\right)}\right) \tag{4.27}$$

For simplicity, throughout the appendix, we denote the above sampling process by $\theta^{(n)}\sim\mathcal{MH}\Big(\mathcal{J}\left(\tilde{\theta}\middle|\theta^{(n-1)}\right),\mathcal{A}\left(\tilde{\theta},\theta^{(n-1)}\right)\Big)$.

4.D Cyclic Sampling

4.D.1 Sampling process

Here is the sampling process; first, the initial values of the parameters $\Theta^{(0)}$ are set, and the burn-in period, the number of iterations for one round (Q), the maximum

number of iterations, and a convergence threshold are specified by users. Then, the parameters are sampled iteratively by following the *cyclic sampling process*, and the generic cyclic sampling process is described below. To ensure the samples obtained are from the posterior distribution, it is suggested to have burn-in sampling periods from which samples are removed before collecting samples. After disregarding the samples from the burin-in period, parameters are sampled by iterating the cyclic sampling Q times, and the convergence of the samples are checked. The iterative estimation process is repeated until the convergence criterion is satisfied or until it reaches the maximum number of iterations. The general iterative *cyclic sampling* process is as followed.

- *Cyclic sampling*: for $k = 1, \ldots, P$,
  - **Step k**: draw a sample $\theta_k^{(n)}$ given the previously sampled parameters, $\{\theta_h\}_{h=1,\ldots,k-1}^{(n)}$ and $\{\theta_h\}_{h=k+1,\ldots,P}^{(n-1)}$ from the full conditional probability distribution $\mathbb{P}\left(\theta_m \middle| \{\theta_h\}_{h=1,\ldots,k-1}^{(n)}, \{\theta_h\}_{h=k+1,\ldots,P}^{(n-1)}, \mathcal{D}\right)$. When this distribution is non-trivial to sample from, the Metropolis-Hasting algorithm is used for sampling.

Our convergence criterion is presented in Appendix 4.D.2 in the supplementary material.

4.D.2 Cyclic Sampling Convergence criterion

Here is the convergence check criterion that we used in this work; every Qth iteration, the convergence of the samples is checked. For each parameter, calculate the mean of the sampled parameter with the first 2/3 and the last 2/3 of the samples (excluding the samples from the burn-in period). If the difference ratio between two means is larger than a convergence threshold (e.g. 0.05), continue the iterative cyclic sampling. If the samples converged or did not converge within the user-specified maximum iteration, finish the iteration.

4.E Model Parameter Estimation

4.E.1 Posterior Distribution of Signals for Landmark Sampling

The nth samples of the landmarks $\lambda_{ij}^{H(n)}$ and $\lambda_{ij}^{V(n)}$ are obtained from the conditional distribution of the individual signal function $f_{ij}^I(x,t_i)$ given $\boldsymbol{Y}$. $f_{ij}^I(x,t_i)$ is predicted from the entire observed signals in training set $\mathcal{D}$ by constructing the multivariate normal distributions for $\boldsymbol{Y}$ and $f_{ij}^I(x,t_i)$ given $(n-1)$th sampled parameters $\Theta^{(n-1)}$.

$$\begin{bmatrix} \boldsymbol{Y} \\ f_{ij}^I(x,t_i) \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \boldsymbol{\lambda}_{\boldsymbol{Y}}^{V\,(n-1)} \\ \lambda_{ij}^{V(n-1)} \mathbb{1}_{[L]} \end{bmatrix}, \begin{bmatrix} \boldsymbol{\Omega}_{\boldsymbol{Y}}^{(n-1)} & \boldsymbol{\Omega}_{\boldsymbol{Y} f_{ij}}^{(n-1)} \\ \boldsymbol{\Omega}_{f_{ij}\boldsymbol{Y}}^{(n-1)} & \boldsymbol{\Omega}_{f_{ij}f_{ij}}^{(n-1)} \end{bmatrix} \right) \tag{4.28}$$

where $\boldsymbol{\lambda}_{\boldsymbol{Y}}^V = \left(\lambda_{11}\mathbb{1}_{[L]}^\mathsf{T}, \ldots, \lambda_{MN}\mathbb{1}_{[L]}^\mathsf{T}\right)^\mathsf{T}$ is the vertical landmarks vector corresponding to $\boldsymbol{Y}$. $\boldsymbol{\Omega}_{\boldsymbol{Y}}^{(n-1)}$, $\boldsymbol{\Omega}_{f_{ij}\boldsymbol{Y}}^{(n-1)}$, and $\boldsymbol{\Omega}_{f_{ij}f_{ij}}^{(n-1)}$ are the prior (joint) covariance matrices given parameters $\boldsymbol{\lambda}^{H(n-1)}$, $\sigma_G^{2\,(n-1)}$, $\sigma_I^{2(n-1)}$, and $\sigma_\varepsilon^{2(n-1)}$. $\boldsymbol{\Omega}_{\boldsymbol{Y}}$ is defined in (4.6). $\boldsymbol{\Omega}_{f_{ij}\boldsymbol{Y}}^{(n-1)} =$

$\left[\mathbf{\Omega}^{(n-1)}_{f_{ij}\mathbf{y}_{11}}, \ldots, \mathbf{\Omega}^{(n-1)}_{f_{ij}\mathbf{y}_{MN}}\right]$ where $\mathbf{\Omega}^{(n-1)}_{f_{ij}\mathbf{y}_{i'j'}}$ is

$$\begin{aligned}\sigma_G^{2\,(n-1)} k_G\Big(\Big(x - \lambda_{ij}^{H(n-1)}, t_i\Big), \Big(\mathbf{x} - \lambda^H_{i'j'}, t_{i'}\Big)\Big) + \\ \delta_{ii'}\delta_{jj'}\sigma_I^{2(n-1)} k_I\Big(x - \lambda_{ij}^{H(n-1)}, \mathbf{x} - \lambda^{H\ (n-1)}_{i'j'}\Big)\end{aligned} \tag{4.29}$$

where δ_{kh} is a Kronecker delta function. The conditional distribution of the individual signal function of jth sensor at t_i is as follows.

$$f_{ij}^{I\,(n)}(x)\,|\boldsymbol{Y}, \Theta^{(n-1)} \sim \mathcal{N}\left(\mu_{ij}^{I|\boldsymbol{Y}^{(n)}}(x), \sigma^{2\,I|\boldsymbol{Y}^{(n)}}_{ij}(x)\right) \tag{4.30}$$

$$\mu_{ij}^{I|\boldsymbol{Y}^{(n)}}(x) = \mathbf{\Omega}^{(n-1)}_{f_{ij}\boldsymbol{Y}}\left(\mathbf{\Omega}^{(n-1)}_{\boldsymbol{Y}}\right)^{-1}\left(\boldsymbol{Y} - \boldsymbol{\lambda}_{\boldsymbol{Y}}^{V\,(n-1)}\right) + \lambda_{ij}^{V(n-1)} \tag{4.31}$$

$$\sigma^{2\,I|\boldsymbol{Y}^{(n)}}_{ij}(x) = \mathbf{\Omega}^{(n-1)}_{f_{ij}f_{ij}} - \mathbf{\Omega}^{(n-1)}_{f_{ij}\boldsymbol{Y}}\left(\mathbf{\Omega}^{(n-1)}_{\boldsymbol{Y}}\right)^{-1}\mathbf{\Omega}^{(n-1)\mathsf{T}}_{f_{ij}\boldsymbol{Y}} \tag{4.32}$$

4.E.2 Acceptance rate for variance sampling

The variances of global and individual Gaussian processes and the environment noises σ_G^2, σ_I^2, σ_ε^2 are sampled by the Metropolis-Hasting algorithm using the property that the full conditional distribution of one given the others is proportional to

$$\mathbb{P}\left(\boldsymbol{Y}\middle|\sigma_G^2, \sigma_I^2, \sigma_\varepsilon^2, \boldsymbol{\lambda}^H_{\boldsymbol{Y}}, \boldsymbol{\lambda}^V_{\boldsymbol{Y}}\right)\pi\left(\sigma_G^2\right)\pi\left(\sigma_I^2\right)\pi\left(\sigma_\varepsilon^2\right) \tag{4.33}$$

where $\pi(\cdot)$ is a prior distribution. $\mathbb{P}\left(\boldsymbol{Y}\middle|\sigma_G^2, \sigma_I^2, \sigma_\varepsilon^2, \boldsymbol{\lambda}^H_{\boldsymbol{Y}}, \boldsymbol{\lambda}^V_{\boldsymbol{Y}}\right)$ is the normal distribution with mean $\boldsymbol{\lambda}^V_{\boldsymbol{Y}}$ and covariance matrix $\mathbf{\Omega}_{\boldsymbol{Y}}$ embedded with $\boldsymbol{\lambda}^H_{\boldsymbol{Y}}$.

The acceptance rate for the full conditional distribution of the global latent

function variance given the other variances sampled in the previous step $\mathcal{A}\left(\tilde{\sigma_G^2}, {\sigma_G^2}^{(n-1)} \middle| {\sigma_I^2}^{(n-1)}, {\sigma_\varepsilon^2}^{(n-1)}, {\boldsymbol{\lambda}_{\boldsymbol{Y}}^H}^{(n)}, {\boldsymbol{\lambda}_{\boldsymbol{Y}}^V}^{(n)}, \mathcal{D}\right)$ is

$$\frac{\mathbb{P}\left(\tilde{\sigma_G^2} \middle| {\sigma_I^2}^{(n-1)}, {\sigma_\varepsilon^2}^{(n-1)}, {\boldsymbol{\lambda}_{\boldsymbol{Y}}^H}^{(n)}, {\boldsymbol{\lambda}_{\boldsymbol{Y}}^V}^{(n)}, \mathcal{D}\right) \mathcal{J}\left({\sigma_G^2}^{(n-1)} \middle| \tilde{\sigma_G^2}\right)}{\mathbb{P}\left({\sigma_G^2}^{(n-1)} \middle| {\sigma_I^2}^{(n-1)}, {\sigma_\varepsilon^2}^{(n-1)}, {\boldsymbol{\lambda}_{\boldsymbol{Y}}^H}^{(n)}, {\boldsymbol{\lambda}_{\boldsymbol{Y}}^V}^{(n)}, \mathcal{D}\right) \mathcal{J}\left(\tilde{\sigma_G^2} \middle| {\sigma_G^2}^{(n-1)}\right)} \tag{4.34}$$

$\mathbb{P}\left(\tilde{\sigma_G^2} \middle| {\sigma_I^2}^{(n-1)}, {\sigma_\varepsilon^2}^{(n-1)}, {\boldsymbol{\lambda}_{\boldsymbol{Y}}^H}^{(n)}, {\boldsymbol{\lambda}_{\boldsymbol{Y}}^V}^{(n)}, \mathcal{D}\right)$ is proportional to

$$\mathbb{P}\left(\boldsymbol{Y} \middle| \tilde{\sigma_G^2}, {\sigma_I^2}^{(n-1)}, {\sigma_\varepsilon^2}^{(n-1)}, {\boldsymbol{\lambda}_{\boldsymbol{Y}}^H}^{(n)}, {\boldsymbol{\lambda}_{\boldsymbol{Y}}^V}^{(n)}\right) \pi\left(\tilde{\sigma_G^2}\right) \pi\left({\sigma_I^2}^{(n-1)}\right) \pi\left({\sigma_\varepsilon^2}^{(n-1)}\right) \tag{4.35}$$

where $\mathbb{P}\left(\boldsymbol{Y} \middle| \tilde{\sigma_G^2}, {\sigma_I^2}^{(n-1)}, {\sigma_\varepsilon^2}^{(n-1)}, {\boldsymbol{\lambda}_{\boldsymbol{Y}}^H}^{(n)}, {\boldsymbol{\lambda}_{\boldsymbol{Y}}^V}^{(n)}\right)$ is the likelihood function of $\tilde{\sigma_G^2}$ given $\boldsymbol{Y}$, ${\sigma_I^2}^{(n-1)}$, and ${\sigma_\varepsilon^2}^{(n-1)}$, and $\pi\left(\sigma^2\right)$ is the prior distribution of σ^2. Therefore, the acceptance rate in (4.34) is

$$\frac{\mathcal{N}\left(\boldsymbol{Y} \middle| \boldsymbol{\lambda}_{\boldsymbol{Y}}^V, \boldsymbol{\Omega}_{\boldsymbol{Y}}\left(\tilde{\sigma_G^2}, {\sigma_I^2}^{(n-1)}, {\sigma_\varepsilon^2}^{(n-1)}, {\boldsymbol{\lambda}_{\boldsymbol{Y}}^H}^{(n)}\right)\right) \pi\left(\tilde{\sigma_G^2}\right) \log \mathcal{N}\left({\sigma_G^2}^{(n-1)} \middle| \log\left(\tilde{\sigma_G^2}\right), \sigma^2_{\mathcal{J}(G)}\right)}{\mathcal{N}\left(\boldsymbol{Y} \middle| \boldsymbol{\lambda}_{\boldsymbol{Y}}^V, \boldsymbol{\Omega}_{\boldsymbol{Y}}\left({\sigma_G^2}^{(n-1)}, {\sigma_I^2}^{(n-1)}, {\sigma_\varepsilon^2}^{(n-1)}, {\boldsymbol{\lambda}_{\boldsymbol{Y}}^H}^{(n)}\right)\right) \pi\left({\sigma_G^2}^{(n-1)}\right) \log \mathcal{N}\left(\tilde{\sigma_G^2} \middle| \log\left({\sigma_G^2}^{(n-1)}\right), \sigma^2_{\mathcal{J}(G)}\right)} \tag{4.36}$$

where

- $\mathcal{N}\left(\boldsymbol{Y} \middle| \boldsymbol{\lambda}_{\boldsymbol{Y}}^V, \boldsymbol{\Omega}_{\boldsymbol{Y}}\left(\sigma_G^2, \sigma_I^2, \sigma_\varepsilon^2, \boldsymbol{\lambda}_{\boldsymbol{Y}}^H\right)\right)$ is the probability density function of the multivariate normal distribution likelihood function given $\boldsymbol{Y}$ with mean $\boldsymbol{\lambda}_{\boldsymbol{Y}}^V$ and variance $\boldsymbol{\Omega}_{\boldsymbol{Y}}\left(\sigma_G^2, \sigma_I^2, \sigma_\varepsilon^2, \boldsymbol{\lambda}_{\boldsymbol{Y}}^H\right)$, which is as follows.

$$\left|\boldsymbol{\Omega}_{\boldsymbol{Y}}\left(\sigma_G^2, \sigma_I^2, \sigma_\varepsilon^2, \boldsymbol{\lambda}_{\boldsymbol{Y}}^H\right)\right|^{-\frac{1}{2}} \exp\left\{-\frac{1}{2}\left(\boldsymbol{Y} - \boldsymbol{\lambda}_{\boldsymbol{Y}}^V\right)^{\mathsf{T}} \left(\boldsymbol{\Omega}_{\boldsymbol{Y}}\left(\sigma_G^2, \sigma_I^2, \sigma_\varepsilon^2, \boldsymbol{\lambda}_{\boldsymbol{Y}}^H\right)\right)^{-1} \left(\boldsymbol{Y} - \boldsymbol{\lambda}_{\boldsymbol{Y}}^V\right)\right\} \tag{4.37}$$

$\boldsymbol{\Omega}_{\boldsymbol{Y}}\left(\sigma_G^2, \sigma_I^2, \sigma_\varepsilon^2, \boldsymbol{\lambda}_{\boldsymbol{Y}}^H\right)$ denotes the covariance matrix of $\boldsymbol{Y}$ given σ_G^2, σ_I^2, and σ_ε^2.

$$\boldsymbol{\Omega}_{\boldsymbol{Y}}\left(\sigma_G^2, \sigma_I^2, \sigma_\varepsilon^2, \boldsymbol{\lambda}_{\boldsymbol{Y}}^H\right) = \boldsymbol{\Omega}_{\boldsymbol{Y}}^I + \boldsymbol{\Omega}_{\boldsymbol{Y}}^G + \sigma_\varepsilon^2 I_{[MNL]} \tag{4.38}$$

$\boldsymbol{\Omega}_{\boldsymbol{Y}}^I$ and $\boldsymbol{\Omega}_{\boldsymbol{Y}}^G$ are defined as follows.

$$\boldsymbol{\Omega}_{\boldsymbol{Y}}^I = \sigma_I^2 \text{diag}\Big(k_I\left(\mathbf{x}_{11} - \lambda_{11}^H, \mathbf{x}_{11} - \lambda_{11}^H\right), \ldots, k_I\left(\mathbf{x}_{MN} - \lambda_{MN}^H, \mathbf{x}_{MN} - \lambda_{MN}^H\right)\Big) \tag{4.39}$$

$$\boldsymbol{\Omega}_{\boldsymbol{Y}}^G = \sigma_G^2 k_G\Big(\left(\boldsymbol{X} - \boldsymbol{\lambda}_{\boldsymbol{Y}}^H, \boldsymbol{C}\right), \left(\boldsymbol{X} - \boldsymbol{\lambda}_{\boldsymbol{Y}}^H, \boldsymbol{C}\right)\Big) \tag{4.40}$$

where $\boldsymbol{X}$, $\boldsymbol{\lambda}_{\boldsymbol{Y}}^H$, and $\boldsymbol{C}$ are the predictors, horizontal landmarks, and target variables, corresponding to $\boldsymbol{Y}$, respectively.

- $\pi\left(\sigma^2\right)$ is the prior distribution of σ^2.
- $\log \mathcal{N}\left(\tilde{\sigma_G^2} \middle| \log\left(\sigma_G^{2\,(n-1)}\right), \sigma_{\mathcal{J}(G)}^2\right)$ is the likelihood of the log-normal distribution with mean parameter $\log\left(\sigma_G^{2\,(n-1)}\right)$ and variance parameter $\sigma_{\mathcal{J}(G)}^2$, which is written as.

$$\log \mathcal{N}\left(\tilde{\sigma_G^2} \middle| \log\left(\sigma_G^{2\,(n-1)}\right), \sigma_{\mathcal{J}(G)}^2\right) = \frac{1}{\sigma_{\mathcal{J}(G)}\sqrt{2\pi}} \exp\left[-\frac{\left\{\log\left(\tilde{\sigma_G^2}\right) - \log\left(\sigma_G^{2\,(n-1)}\right)\right\}^2}{2\sigma_{\mathcal{J}(G)}^2}\right] \tag{4.41}$$

The acceptance rates for the full conditional distributions of the variance σ_I^2 and σ_ε^2 can be derived in the same manner.

$\mathcal{A}\left(\tilde{\sigma_I^2}, {\sigma_I^2}^{(n-1)} \middle| {\sigma_G^2}^{(n)}, {\sigma_\varepsilon^2}^{(n-1)}, {\boldsymbol{\lambda}_{\boldsymbol{Y}}^H}^{(n)}, {\boldsymbol{\lambda}_{\boldsymbol{Y}}^V}^{(n)}\right)$ is

$$\frac{\mathcal{N}\left(\boldsymbol{Y}\middle|{\boldsymbol{\lambda}_{\boldsymbol{Y}}^V}^{(n)}, \boldsymbol{\Omega}_{\boldsymbol{Y}}\left({\sigma_G^2}^{(n)}, \tilde{\sigma_I^2}, {\sigma_\varepsilon^2}^{(n-1)}, {\boldsymbol{\lambda}_{\boldsymbol{Y}}^H}^{(n)}\right)\right)\pi\left(\tilde{\sigma_I^2}\right)\log\mathcal{N}\left({\sigma_I^2}^{(n-1)}\middle|\log\left(\tilde{\sigma_I^2}\right), \sigma^2_{\mathcal{J}(I)}\right)}{\mathcal{N}\left(\boldsymbol{Y}\middle|{\boldsymbol{\lambda}_{\boldsymbol{Y}}^V}^{(n)}, \boldsymbol{\Omega}_{\boldsymbol{Y}}\left({\sigma_G^2}^{(n)}, {\sigma_I^2}^{(n-1)}, {\sigma_\varepsilon^2}^{(n-1)}, {\boldsymbol{\lambda}_{\boldsymbol{Y}}^H}^{(n)}\right)\right)\pi\left({\sigma_I^2}^{(n-1)}\right)\log\mathcal{N}\left(\tilde{\sigma_I^2}\middle|\log\left({\sigma_I^2}^{(n-1)}\right), \sigma^2_{\mathcal{J}(I)}\right)} \tag{4.42}$$

$\mathcal{A}\left(\tilde{\sigma_\varepsilon^2}, {\sigma_\varepsilon^2}^{(n-1)} \middle| {\sigma_I^2}^{(n)}, {\sigma_G^2}^{(n)}, {\boldsymbol{\lambda}_{\boldsymbol{Y}}^H}^{(n)}, {\boldsymbol{\lambda}_{\boldsymbol{Y}}^V}^{(n)}\right)$ is

$$\frac{\mathcal{N}\left(\boldsymbol{Y}\middle|{\boldsymbol{\lambda}_{\boldsymbol{Y}}^V}^{(n)}, \boldsymbol{\Omega}_{\boldsymbol{Y}}\left({\sigma_G^2}^{(n)}, {\sigma_I^2}^{(n)}, \tilde{\sigma_\varepsilon^2}, {\boldsymbol{\lambda}_{\boldsymbol{Y}}^H}^{(n)}\right)\right)\pi\left(\tilde{\sigma_\varepsilon^2}\right)\log\mathcal{N}\left({\sigma_\varepsilon^2}^{(n-1)}\middle|\log\left({\sigma_\varepsilon^2}^{(n-1)}\right), \sigma^2_{\mathcal{J}(\varepsilon)}\right)}{\mathcal{N}\left(\boldsymbol{Y}\middle|{\boldsymbol{\lambda}_{\boldsymbol{Y}}^V}^{(n)}, \boldsymbol{\Omega}_{\boldsymbol{Y}}\left({\sigma_G^2}^{(n)}, {\sigma_I^2}^{(n)}, {\sigma_\varepsilon^2}^{(n-1)}, {\boldsymbol{\lambda}_{\boldsymbol{Y}}^H}^{(n)}\right)\right)\pi\left({\sigma_\varepsilon^2}^{(n-1)}\right)\log\mathcal{N}\left(\tilde{\sigma_\varepsilon^2}\middle|\log\left(\tilde{\sigma_\varepsilon^2}\right), \sigma^2_{\mathcal{J}(\varepsilon)}\right)} \tag{4.43}$$

4.E.3 Sampling Variances of the Gaussian Process for Landmarks with Metropolis-Hasting algorithm

The variances of the Gaussian process functions for the landmarks σ_H^2 and $\sigma_{\varepsilon H}^2$ can be sampled by the Metropolis-Hasting algorithm using the property that $\mathbb{P}\left(\sigma_H^2|\sigma_{\varepsilon V}^2, \boldsymbol{\lambda}^H, \boldsymbol{t}_{\boldsymbol{\lambda}}\right)$ is proportional to

$$\mathbb{P}\left(\boldsymbol{\lambda}^H\middle|\sigma_H^2, \sigma_{\varepsilon H}^2, \boldsymbol{t}_{\boldsymbol{\lambda}}\right)\pi\left(\sigma_H^2\right)\pi\left(\sigma_{\varepsilon H}^2\right) \tag{4.44}$$

where $\boldsymbol{\lambda}^H = \left(\lambda_{11}^H, \ldots, \lambda_{MN}^H\right)^\mathsf{T}$ and $\boldsymbol{t}_{\boldsymbol{\lambda}} = \left(t_1\mathbb{1}_{[N]}^\mathsf{T}, \ldots, t_M\mathbb{1}_{[N]}^\mathsf{T}\right)^\mathsf{T} \in \mathbb{R}^{MN}$ is a vector of the target variable corresponding to $\boldsymbol{\lambda}^H$. $\mathbb{P}\left(\boldsymbol{\lambda}^H\middle|\sigma_H^2, \sigma_{\varepsilon H}^2, \boldsymbol{t}_{\boldsymbol{\lambda}}\right)$ is the probability distribution of the normal distribution with mean $\mathbf{0}$ and covariance matrix $\sigma_H^2 k_H(\boldsymbol{t}_{\boldsymbol{\lambda}}, \boldsymbol{t}_{\boldsymbol{\lambda}}) + \sigma^2_{\sigma^2_{\varepsilon H}} I_{[MN]}$. σ_V^2 and $\sigma_{\varepsilon V}^2$ are sampled in the same manner.

The acceptance rates for the full distribution of σ_H^2 and $\sigma_{\varepsilon H}^2$ are described. The

acceptance rate of σ_V^2 and $\sigma_{\varepsilon V}^2$ can be derived in the same manner. Let's denote $\boldsymbol{\lambda}^H = \left(\lambda_{11}^H, \ldots, \lambda_{MN}^H\right)^\mathsf{T}$ and $\boldsymbol{t}_{\boldsymbol{\lambda}} = \left(t_1 \mathbb{1}_{[N]}^\mathsf{T}, \ldots, t_M \mathbb{1}_{[N]}^\mathsf{T}\right)^\mathsf{T}$. $\mathcal{A}\left(\sigma_H^2, {\sigma_H^2}^{(n-1)} \middle| {\sigma_{\varepsilon H}^2}^{(n-1)}, \boldsymbol{\lambda}^{H(n)}\right)$ is

$$\frac{\mathcal{N}\left(\boldsymbol{\lambda}^{H(n)} \middle| \mathbf{0}, \boldsymbol{\Omega}_H\left(\tilde{\sigma_H^2}, {\sigma_{\varepsilon H}^2}^{(n-1)}\right)\right) \pi\left(\tilde{\sigma_H^2}\right) \log \mathcal{N}\left(\log\left({\sigma_H^2}^{(n-1)}\right) \middle| \tilde{\sigma_H^2}, \sigma_{\mathcal{J}(\lambda)}^2\right)}{\mathcal{N}\left(\boldsymbol{\lambda}^{H(n)} \middle| \mathbf{0}, \boldsymbol{\Omega}_H\left({\sigma_H^2}^{(n-1)}, {\sigma_{\varepsilon H}^2}^{(n-1)}\right)\right) \pi\left({\sigma_H^2}^{(n-1)}\right) \log \mathcal{N}\left(\tilde{\sigma_H^2} \middle| \log\left({\sigma_H^2}^{(n-1)}\right), \sigma_{\mathcal{J}(\lambda)}^2\right)} \tag{4.45}$$

where $\mathcal{N}\left(\boldsymbol{\lambda}^{H(n)} \middle| \mathbf{0}, \boldsymbol{\Omega}_H(\sigma_H^2, \sigma_{\varepsilon H}^2)\right)$ is the probability density function of the multivariate normal distribution with mean zero and covariance matrix

$$\boldsymbol{\Omega}_H\left(\sigma_H^2, \sigma_{\varepsilon H}^2\right) = \sigma_H^2 k_\lambda(\boldsymbol{t}_{\boldsymbol{\lambda}}, \boldsymbol{t}_{\boldsymbol{\lambda}}) + \sigma_{\varepsilon H}^2 I_{[MN]} \tag{4.46}$$

In the same manner, $\mathcal{A}\left(\sigma_{\varepsilon H}^2, {\sigma_{\varepsilon H}^2}^{(n-1)} \middle| {\sigma_H^2}^{(n-1)}, \boldsymbol{\lambda}^{H(n)}\right)$ is

$$\frac{\mathcal{N}\left(\boldsymbol{\lambda}^{H(n)} \middle| \mathbf{0}, \boldsymbol{\Omega}_H\left({\sigma_H^2}^{(n-1)}, \tilde{\sigma_{\varepsilon H}^2}\right)\right) \pi\left(\tilde{\sigma_{\varepsilon H}^2}\right) \log \mathcal{N}\left(\log\left({\sigma_{\varepsilon H}^2}^{(n-1)}\right) \middle| \tilde{\sigma_{\varepsilon H}^2}, \sigma_{\mathcal{J}(\varepsilon\lambda)}^2\right)}{\mathcal{N}\left(\boldsymbol{\lambda}^{H(n)} \middle| \mathbf{0}, \boldsymbol{\Omega}_H\left({\sigma_H^2}^{(n-1)}, {\sigma_{\varepsilon H}^2}^{(n-1)}\right)\right) \pi\left({\sigma_{\varepsilon H}^2}^{(n-1)}\right) \log \mathcal{N}\left(\tilde{\sigma_{\varepsilon H}^2} \middle| \log\left({\sigma_{\varepsilon H}^2}^{(n-1)}\right), \sigma_{\mathcal{J}(\varepsilon\lambda)}^2\right)} \tag{4.47}$$

4.F Calibration

4.F.1 Full conditional distributions of the landmarks

The full conditional distribution of the test signal's landmark (denoted by λ_{new} representing either horizontal or vertical landmark) $\mathbb{P}\left(\lambda_{new} \middle| \mathbf{y}_{new}, \mathbf{x}_{new}, \widehat{\Theta}, t\right)$ is pro-

portional to

$$\mathbb{P}\left(\mathbf{y}_{new}\middle|\mathbf{x}_{new},\widehat{\Theta},\lambda^{H}_{new},\lambda^{V}_{new},t\right)\mathbb{P}\left(\lambda_{new}\middle|\widehat{\Theta},t\right) \tag{4.48}$$

$\mathbb{P}\left(\mathbf{y}_{new}\middle|\mathbf{x}_{new},\widehat{\Theta},\lambda^{H}_{new},\lambda^{V}_{new},t\right)$ is the multivariate normal distribution with mean λ^{V}_{new} and covariance $\mathbf{\Omega}_{\mathbf{y}_{new}}$ given the estimated parameters $\widehat{\Theta}$. $\mathbb{P}\left(\lambda_{new}\middle|\widehat{\Theta},t\right)$ is the posterior prediction of the Gaussian process of λ_{new} based on the estimated parameters, which can be expanded as.

$$\mathcal{N}\left(\widehat{\mathbf{\Omega}}_{\lambda_{new}\boldsymbol{\lambda}}\left(\widehat{\mathbf{\Omega}}_{\boldsymbol{\lambda}}\right)^{-1}\widehat{\boldsymbol{\lambda}},\ \widehat{\mathbf{\Omega}}_{\lambda_{new}}-\widehat{\mathbf{\Omega}}_{\lambda_{new}\boldsymbol{\lambda}}\left(\widehat{\mathbf{\Omega}}_{\boldsymbol{\lambda}}\right)^{-1}\widehat{\mathbf{\Omega}}_{\boldsymbol{\lambda}\lambda_{new}}\right) \tag{4.49}$$

where $\boldsymbol{\lambda}=(\lambda_{11},\ldots,\lambda_{MN})^{\mathsf{T}}$ and $\boldsymbol{t}_{\lambda}$ is a vector of target variables t corresponding to $\boldsymbol{\lambda}$.

$\widehat{\mathbf{\Omega}}_{\boldsymbol{\lambda}}=\widehat{\sigma^2_{\lambda}}k_{\lambda}(\boldsymbol{t}_{\lambda},\boldsymbol{t}_{\lambda})+\widehat{\sigma^2_{\varepsilon\lambda}}I_{[MN]}$, $\widehat{\mathbf{\Omega}}_{\lambda_{new}\boldsymbol{\lambda}}=\widehat{\sigma^2_{\lambda}}k_{\lambda}(t,\boldsymbol{t}_{\lambda})$, and $\widehat{\mathbf{\Omega}}_{\lambda_{new}}=\widehat{\sigma^2_{\lambda}}k_{\lambda}(t,t)+\widehat{\sigma^2_{\varepsilon\lambda}}$ where σ^2_{λ}, $\sigma^2_{\varepsilon\lambda}$, and $k_{\lambda}(\cdot,\cdot)$ will be substituted with σ^2_{H}, $\sigma^2_{\varepsilon H}$, and $k_{H}(\cdot,\cdot)$ for the horizontal landmarks and σ^2_{V}, $\sigma^2_{\varepsilon V}$, and $k_{V}(\cdot,\cdot)$ for the vertical landmarks. Because λ^{V}_{new} is the mean of the normal distribution and its prior distribution is also Gaussian in (4.49), the posterior distribution of λ^{V}_{new} is the normal distribution, from which can be easily sampled. The full conditional distribution of λ^{V}_{new} is presented in the following Appendix 4.F.1.2. However, the posterior distribution of λ^{H}_{new} is not following a well-known distribution family because the covariance matrix of the likelihood function $\mathbf{\Omega}_{\mathbf{y}_{new}}$ is a function of λ^{H}_{new}. Hence, the Metropolis-Hasting algorithm is used to sample λ^{H}_{new} from the full conditional posterior distribution. The acceptance rate for the Metropolis-Hasting algorithm for λ^{V}_{new} is presented in

following Appendix 4.F.1.1.

4.F.1.1 Acceptance rate for landmark for the horizontal landmarks

The acceptance rate for the full conditional distribution of λ^V_{new}, $\mathcal{A}\left(\tilde{\lambda}^H_{new}, {\lambda^H_{new}}^{(n-1)}\right)$, is as follows.

$$\frac{\mathcal{N}\left(\mathbf{y}_{new}\middle|\lambda^V_{new},\, \mathbf{\Omega}_{\mathbf{y}_{new}}\left(\tilde{\lambda}^H_{new}\right)\right)\mathcal{N}\left(\tilde{\lambda}^H_{new}\middle|\hat{\boldsymbol{\mu}}_{new},\, \hat{\Sigma}_{new}\right)}{\mathcal{N}\left(\mathbf{y}_{new}\middle|\lambda^V_{new},\, \mathbf{\Omega}_{\mathbf{y}_{new}}\left({\lambda^H_{new}}^{(n-1)}\right)\right)\mathcal{N}\left({\lambda^H_{new}}^{(n-1)}\middle|\hat{\boldsymbol{\mu}}_{new},\, \hat{\Sigma}_{new}\right)} \tag{4.50}$$

where $\hat{\boldsymbol{\mu}}_{new} = \widehat{\mathbf{\Omega}}_{\lambda_{new}\boldsymbol{\lambda}}\left(\widehat{\mathbf{\Omega}}_{\boldsymbol{\lambda}}\right)^{-1}\hat{\boldsymbol{\lambda}}$, $\hat{\Sigma}_{new} = \widehat{\mathbf{\Omega}}_{\lambda_{new}} - \widehat{\mathbf{\Omega}}_{\lambda_{new}\boldsymbol{\lambda}}\left(\widehat{\mathbf{\Omega}}_{\boldsymbol{\lambda}}\right)^{-1}\widehat{\mathbf{\Omega}}_{\boldsymbol{\lambda}\lambda_{new}}$, and $\mathbf{\Omega}_{\mathbf{y}_{new}}\left(\lambda^H_{new}\right) = \sigma^2_I k_I\left(\mathbf{x}_{new} - \lambda^H_{new}, \mathbf{x}_{new} - \lambda^H_{new}\right) + \sigma^2_G k_G((\mathbf{x}_{new} - \lambda^H_{new}, t_{new}), (\mathbf{x}_{new} - \lambda^H_{new}, t_{new})) + \sigma^2_\varepsilon I_{[L_{new}, L_{new}]}$ and L_{new} is the size of $\mathbf{x}_{new}$ and $\mathbf{y}_{new}$.

4.F.1.2 Posterior mean and variance of the vertical landmarks

The full conditional posterior distribution of $\mathbb{P}\left(\lambda^V_{new}\middle|\mathbf{y}_{new}, \mathbf{x}_{new}, \widehat{\Theta}, t\right)$ is as follows.

$$\mathbb{P}\left(\lambda^V_{new}\middle|\mathbf{y}_{new}, \mathbf{x}_{new}, \widehat{\Theta}, t\right) \propto \mathbb{P}\left(\mathbf{y}_{new}\middle|\mathbf{x}_{new}, \widehat{\Theta}, \lambda^H_{new}, \lambda^V_{new}, t\right)\mathbb{P}\left(\lambda^V_{new}\middle|\widehat{\Theta}, t\right) \tag{4.51}$$

$\mathbb{P}\left(\mathbf{y}_{new}\middle|\mathbf{x}_{new}, \widehat{\Theta}, \lambda^H_{new}, \lambda^V_{new}, t\right)$ is the probability density of the multivariate normal distribution

$$\mathcal{N}\left(\mathbf{y}_{new}\middle|\lambda^V_{new}\mathbb{1}_{[L_{new}]},\, \mathbf{\Omega}_{\mathbf{y}_{new}}\left(\lambda^H_{new}\right)\right) \tag{4.52}$$

and $\mathbb{P}\left(\lambda_{new}^{V}\middle|\mathcal{D},\widehat{\Theta},t_{new}\right)$ is

$$\mathcal{N}\left(\lambda_{new}^{V}\middle|\widehat{\boldsymbol{\Omega}}_{\lambda_{new}\boldsymbol{\lambda}}\left(\widehat{\boldsymbol{\Omega}}_{\boldsymbol{\lambda}}\right)^{-1}\widehat{\boldsymbol{\lambda}},\ \widehat{\boldsymbol{\Omega}}_{\lambda_{new}}-\widehat{\boldsymbol{\Omega}}_{\lambda_{new}\boldsymbol{\lambda}}\left(\widehat{\boldsymbol{\Omega}}_{\boldsymbol{\lambda}}\right)^{-1}\widehat{\boldsymbol{\Omega}}_{\boldsymbol{\lambda}\lambda_{new}}\right) \tag{4.53}$$

Therefore, the full conditional posterior distribution of $\mathbb{P}\left(\lambda_{new}^{V}{}^{(n)}\middle|\mathbf{y}_{new},\mathbf{x}_{new},\widehat{\Theta},t^{(n-1)}\right)$ is

$$\mathcal{N}\left(\mu_{\lambda_{new}^{V}}{}^{(n)},\sigma^{2}_{\lambda_{new}^{V}}{}^{(n)}\right) \tag{4.54}$$

where

$$\sigma^{2}_{\lambda_{new}^{V}}{}^{(n)}=\frac{1}{\mathbb{1}_{[L_{new}]}^{\mathsf{T}}\left(\boldsymbol{\Omega}_{\mathbf{y}_{new}}^{(n-1)}\right)^{-1}\mathbb{1}_{[L_{new}]}}+\frac{1}{\sigma^{2}_{\lambda_{new}^{V}|\boldsymbol{\lambda}^{V}}{}^{(n)}} \tag{4.55}$$

$$\mu_{\lambda_{new}^{V}}{}^{(n)}=\left\{\mathbb{1}_{[L_{new}]}^{\mathsf{T}}\left(\mathbf{y}_{new}-\lambda_{new}^{V}{}^{(n-1)}\right)\left(\boldsymbol{\Omega}_{\mathbf{y}_{new}}^{(n-1)}\right)^{-1}+\frac{\mu_{\lambda_{new}^{V}|\boldsymbol{\lambda}^{V}}{}^{(n)}}{\sigma^{2}_{\lambda_{new}^{V}|\boldsymbol{\lambda}^{V}}{}^{(n)}}\right\}\sigma^{2}_{\lambda_{new}^{V}}{}^{(n)} \tag{4.56}$$

and

$$\mu_{\lambda_{new}^{V}|\boldsymbol{\lambda}^{V}}{}^{(n)}=\widehat{\boldsymbol{\Omega}}_{\lambda_{new}^{V}\boldsymbol{\lambda}^{V}}\left(\widehat{\boldsymbol{\Omega}}_{\boldsymbol{\lambda}^{V}}\right)^{-1}\widehat{\boldsymbol{\lambda}^{V}} \tag{4.57}$$

$$\sigma^{2}_{\lambda_{new}^{V}|\boldsymbol{\lambda}^{V}}{}^{(n)}=\widehat{\boldsymbol{\Omega}}_{\lambda_{new}^{V}}-\widehat{\boldsymbol{\Omega}}_{\lambda_{new}^{V}\boldsymbol{\lambda}^{V}}\left(\widehat{\boldsymbol{\Omega}}_{\boldsymbol{\lambda}^{V}}\right)^{-1}\widehat{\boldsymbol{\Omega}}_{\boldsymbol{\lambda}^{V}\lambda_{new}^{V}} \tag{4.58}$$

4.G Computational complexity and computed time for numerical studies

The most computationally intensive part of our algorithm is the inverse operation of the Gaussian process prediction in the iterative operations of parameter estimation (i.e., training) and calibration (i.e., testing), which are $\mathcal{O}(n_{train}^3)$ and $\mathcal{O}(n_{new}^3)$ where n_{train} and n_{new} are the number of data points in the training and test data, respectively. Accordingly, the training is computationally intensive but will be conducted offline. On the other hand, in the test stage, we focus on a single signal to be calibrated while using the parameters estimated in the training set; therefore, the calibration computation complexity is fairly low. We provide the average computation times for the parameter estimation of each data set and the calibration times of each signal in the test set in Table 4.G.1. The numerical and case studies were performed in Matlab 2019b, except the functional linear regression that was run in R, on Ubuntu Linux. The computer used in the experiments has Intel(R) Core(TM) i9-9900K CPU @ 3.60GHz and 128 GB RAM memory. In principle, if we conduct parallel computing to collect samples from Markov chains on n cores, we can further reduce the computation time by cutting it into n times shorter.

	Case I	Case II	Case III	Case IV	Case V
Param. Est.	9.38 min	10.91 min	10.92 min	0.45 min	15.58 min
Calibration	1.08 min	2.04 min	1.32 min	0.22 min	1.70 min

Table 4.G.1: Parameter Estimation time of each training data set and calibration computation time for each signal in the numerical studies

4.H Discussion of the interval estimates in numerical and case studies

The length of the interval estimate implies the quantified uncertainties caused by the randomness in the data. A wide interval estimate does not mean poor performance of a method; it rather may imply that the prediction involves higher uncertainties with a wide probability distribution due to the randomness in the original data. In other words, accuracy of predicted interval estimate, rather than the simple length of the interval estimate, is important. Too narrow estimated interval lengths will always fail in including the true values due to their conservative view on the uncertainties ("the true values would be within such a narrow interval"). On the other hand, too wide interval lengths would always include the true values, which also incur deviation from the target significance level (i.e., 90% in this case). Indeed, in both numerical and case studies in the main text, all the reference methods failed in providing accurate interval estimates; their interval estimates only include the true target variable with much less probability than 90% (large errors) due to the too narrow interval lengths. On the other hand, our proposed model provided predicted interval estimates that include the true value with probability very close to 90% (small errors).

Table 4.H.1 and 4.H.2 display the median length of the estimated intervals (the difference between 95th and 5th percentiles of the posterior samples of the target variable) of the numerical and case studies. The table indicates that the other methods produce too conservative views on the uncertainties and fail in

accurately accounting for the uncertainties involved in prediction; in fact, the lengths of their interval estimates need to be longer. Notice that *FLR*, in most cases, has a longer prediction intervals than $LR(Lm)$ but the accuracy is generally worse. In Case V, where the signals involve the most significant randomness (i.e., significant uncertainties), *FLR* even misled to shorter lengths of the intervals than those in Case I. This confirms that Bayesian calibration with Gaussian processes are more suitable for characterizing the uncertainties in the functional data.

Table 4.H.1: Median length of the estimated 90%-intervals from numerical study

	LEGP	LR(Aln)	LR(Lm)	FLR
Case I	10.8	7	5.3	8.3
Case II	10.3	7.9	5.7	7.7
Case III	10.2	6.2	4.9	5.2
Case IV	12.3	6.1	4.1	8
Case V	20.5	9.8	7.5	6.4

Table 4.H.2: Median length of the estimated 90%-intervals from case study

	LEGP	LR(Aln)	LR(Lm)	FLR
Case study	9.3	13.8	6.8	3.5

5 ROBUST PARAMETER DESIGN ON DUAL STOCHASTIC RESPONSE MODELS WITH CONSTRAINED BAYESIAN OPTIMIZATION *

In engineering system design, minimizing the variations of the quality measurements while guaranteeing their overall quality up to certain levels, namely the robust parameter design (RPD), is crucial. Recent works have dealt with the design of a system whose response-control variables relationship is a deterministic function with a complex shape and function evaluation is expensive. In this work, we propose a Bayesian optimization method for the RPD of stochastic functions. Dual stochastic response models are carefully designed for stochastic functions. The heterogeneous variance of the sample mean is addressed by the predictive mean of the log variance surrogate model in a two-step approach. We establish an acquisition function that favors exploration across the feasible and optimality-improvable regions to effectively and efficiently solve the stochastic constrained optimization problem. The performance of our proposed method is demonstrated by the extensive numerical and case studies.

5.1 Introduction

We often want to design an engineering system so that its quality/performance measurements (also called "response") have small random variations while achieving a satisfactory overall quality/performance. For example, in manufacturing design,

*This chapter is based on the paper: **Lee, J.**, S. Zhou, and J. Chen. (2022) Robust parameter design on dual stochastic response models with constrained Bayesian optimization. *submitted*

reducing manufacturing process variations is crucial to minimize the uncertainty of the product quality. Also, in sensor design, finding the sensor structure that creates the minimum variations in the output signals is very important. Robust parameter design (RPD) is a statistical method that finds the optimal setting of the control variables where the resulting responses are insensitive to the variations created from noise variables, which are the sources of the undesirable uncertainty in the response (Taguchi, 1986). RPD has been significantly studied for the past couple of decades. RPD methods can be used in automatic process control, in which RPD methods automatically find the best robust settings with automated engineering systems or simulators (Zhong et al., 2010). Conventional RPD minimizes the expected loss function where the minimization leads to reducing both the variance of the response and the distance between the mean and a target (or minimizing/maximizing the mean depending on the loss function we use). In other words, the loss function presents the trade-off between the variance and the mean. However, in practice, we often want to explicitly constrain the overall quality of the process/products (i.e., the mean of the response) while minimizing the variance of the response; we call this constrained robust parameter design (CRPD).

Many Conventional RPD works assume that noise variables are observable and controllable in the experimentation stage but uncontrollable in the real world. Under such assumptions, RPD methods have also been studied for deterministic computer simulators. However, in many types of real-world data, there are various unobserved and uncontrollable factors that create variations in responses. For example, in multistage manufacturing processes, every operation involves process

variations that often cannot be directly measured, and such variations propagate to the downstream sensing measurements of the final product quality (Lee et al., 2020). Some computer simulators are inherently stochastic (Al-Aomar, 2006). In such stochastic environments, with the same values of the control variables, the output response will have different values with variations. Naturally, considering stochastic environments broadens the range of applications of the RPD.

Recently, researchers have studied RPD methods for expensive data where the relationships between the response and control variables are unknown and complex. For instance, many simulations take a long time to run a single round of experiment where the relationship between the response surface and the control variables is not known. Under stochastic environments, mean and variance of the response are functions of the control variables, of which mathematical representations are often complex and unknown; therefore, such functions cannot be expressed by parametric expressions, such as quadratic forms.

It is very important to establish a CRPD method that addresses the aforementioned challenges: 1) data are generated from stochastic environments; 2) data are expensive to obtain; 3) the underlying mean and variance are unknown and complex functions of control variables. Time-consuming experiments call for a sequential optimization method that can leverage the historical data to efficiently find the optimal solution within a small number of experiments. In particular, we establish a new stochastic constrained Bayesian optimization method. Bayesian optimization consists of two parts: surrogate models and acquisition function. *Surrogate models* predict the unobserved functions and quantify the uncertainty;

therefore, the Gaussian processes are commonly used for the surrogate models; an *acquisition function* serves as an evaluation criterion of the control variables to be used for the next experiment based on the prediction and quantified uncertainties obtained from the surrogate models. To deal with *stochastic* response, instead of directly modeling the response surface, its mean and variance can be represented by surrogate models as functions of the control variables. In particular, two Gaussian processes surrogate models are constructed for the unknown and complex mean and variance functions. Bayesian optimization methods trade-off between exploitation and exploration. It exploits its prediction to search for a precise solution while exploring the control variable space by selecting a control variable with high uncertainty. For effective exploration, precise modeling of variance across the control variable space is important. However, it is challenging to model the variance of the errors in the mean surrogate model because sample means have heterogeneous variance across control variable values, and the exact functional form of the variance is unknown. Besides, a desirable acquisition function should encourage exploration sufficiently while exploiting the prediction results. Through our proposed stochastic constrained Bayesian optimization method, we addressed the aforementioned technical challenges. In particular, our contributions can be summarized in two aspects.

- First, we establish stochastic dual response surrogate models of the mean and log variance of the response by the Gaussian processes. Specifically, the sample mean and log sample variance are used as the observations of the surrogate models. The heteroskedasticity of the variance of the errors in the

mean surrogate model is addressed by using a two-step approach. Particularly, we predict the variance of the sample mean by the log variance surrogate model.

- Second, we propose an acquisition function that explicitly encourages exploration while exploiting the predicted feasibility and optimality; both exploration and exploitation are performed based on the estimated surrogate models. It selects the next experimental setting in unexplored regions with high uncertainty that are likely to be feasible and have better objective values. The current best optimal point is obtained from the predicted functions.

Our proposed method quickly and successfully finds the optimal robust design that has the minimum variance while satisfying the mean constraint where the data are generated from unknown complex expensive-to-evaluate stochastic functions. We demonstrate the advantageous features of our proposed method with challenging synthetic problems in the numerical study. In the case study, we show our proposed method works well and outperforms the reference methods in the real world data by using the graphene field-effect transistor nanosensor data.

The rest of the chapter is organized as follows. Relevant literature is reviewed in Section 5.2. Section 5.3 describes our proposed method. In Section 5.3.1, the dual stochastic surface response surrogate models are presented. The proposed acquisition function for the constrained optimization on the stochastic functions is detailed in Section 5.3.2. The performance of our proposed method is demonstrated by the numerical study in Section 5.4 and by the case study in Section 5.5. Section 5.6 gives some discussions and draws conclusions.

5.2 Literature Review

Conventional RPD conducts a 'batch' of experiments and obtains a 'batch' of data at once, and the optimal robust setting is found based on the given data (Taguchi, 1986). Because whole experiments are conducted at once not leveraging the information of historical data, conventional RPD generally need large size of data. Besides, conventional RPD minimizes a single loss function, such as the quadratic loss function (Taguchi, 1986; Tan and Wu, 2012; Ouyang et al., 2019; Vanli and Castillo, 2009; Shen, 2017; Zhong et al., 2010), and the weighted sum of mean and variance (or standard deviation) (Apley et al., 2005; Shen, 2017). Such loss functions characterize the trade-off between a lower variance and a better mean (e.g., 'better' can be defined as either 'lower', 'larger', or 'close to a target' mean value). However, such methods cannot constrain the mean of the responses. In addition, many conventional RPD works considered factor variables (i.e., binary or ordinal variables). Others who considered continuous control variables often used parametric regressions (mainly, linear regressions in quadratic expressions); such models cannot address the stochastic data generated from unknown/complex mean and variance functions.

Dual response models have been studied for CRPD by fitting the mean and variance of the response in terms of control variables (Vining and Myers, 1990; Copeland and Nelson, 1996; Castillo et al., 1997; Fan, 2000; Köksoy and Doganaksoy, 2003; Yanıkoğlu et al., 2015). The dual response models can be used for stochastic response data where the noise variables are not observable nor controllable. However, most dual response models have common strict settings: 1) conducting

a batch of experiments; 2) parametric linear models for each mean and variance (mostly quadratic, but sometimes with a higher-order polynomial expression (Giovagnoli and Romano, 2008)); 3) assuming independent errors with homogeneous variances; 4) optimization only using the fitted mean curves (i.e, only exploitation). These four conventional settings limit the use of such methods for CRPD.

Recent RPD works have dealt with complex shape black-box functions of data, which are mostly motivated by the advent of computer simulations (Tan and Wu, 2012; Tan, 2015; Shen, 2017; Lehman et al., 2004). Most RPD literature on computer simulations assumes *deterministic* computer simulators where noise variables are assumed to be observable and controllable in experimentation stage; the true response value will be known with certainty after an experiment. The simulation outputs are modeled by the typical noiseless deterministic Gaussian process in terms of both control and noise variables; such deterministic Gaussian process models cannot be used for stochastic functions with differing mean and variance functions.

Space-filling design, such as Latin hypercube sampling (McKay et al., 1979) and uniform design (Yang et al., 2021; Fang et al., 2000), is also often used where points far from each other are selected for experiments (Bektas et al., 2017). However, space-filling design only favors exploration across the variable space and thus requires many experiments.

To conduct a smaller number of experiments with the costly simulators, model-based sequential optimization techniques have been used (Shen, 2017; Lehman et al., 2004). Such sequential optimization methods select a setting to be used

for the next experiment by utilizing the knowledge obtained from the historical data. Shen (2017) used a stochastic approximation technique to optimize a loss function, which is a gradient-based approach. Gradient calculation requires many experiments with orthogonal designs on deterministic functions and is difficult to perform on stochastic functions. Moreover, the gradient-based approach tends to find a local optimum. Lehman et al. (2004) used the weighted expected improvement for deterministic simulators. They calculated the mean and variance by assuming the noise variables and their probability distribution are known to solve the constrained RPD. These methods can only be used for the deterministic data generation functions whose noise variables are observable and controllable and cannot be used in stochastic settings.

One may consider a global derivative-free optimization method generally used for stochastic functions, such as the pattern search (Hooke and Jeeves, 1961) by using the sample mean and sample variance as the observations. Although such a method can handle a stochastic function in a complex unknown shape, such a method is not desirable when an experiment is expensive to conduct because such method does not consider the uncertainty of the samples; it generally requires a large size of data.

Bayesian optimization has been shown efficient and effective for sequential optimization and widely used in the field of model hyperparameter tuning for machine learning algorithms (Jones et al., 1998; Snoek et al., 2012). In the Bayesian optimization literature, several acquisition functions have been developed for the constrained optimization on the deterministic functions (Zhan and Xing, 2020),

including the weighted expected improvement (WEI) (Gardner et al., 2014). However, such methods cannot be directly applied to stochastic functions. Recently, Letham et al. (2019) developed the noisy weighted expected improvement (NWEI), using a quasi-Monte Carlo sampling technique to use the weighted expected improvement for stochastic functions. NWEI is inefficient because of its sampling procedure at each iteration. Furthermore, WEI (accordingly, NWEI) is a greedy technique because WEI multiplies two greedy terms: the expectation improvement (Jasrasaria and Pyzer-Knapp, 2018) and the probability of feasibility (the cumulative density functions are known to be greedy (Brochu et al., 2010; Berk et al., 2019)).

Lehman et al. (2004) used the Bayesian optimization, in particular, WEI, for the constrained RPD with a deterministic simulator. They calculated the mean and variance from the predicted noiseless Gaussian process function fitted to the *deterministic* simulator output surface and the *assumed-to-be-known* probability mass functions of noise variables; therefore, this method cannot be used for a stochastic function. Furthermore, even for a deterministic simulator, this method has several limitations, especially when the number of noise variables is large: 1) As mentioned above, WEI is a greedy method. 2) This method requires many samples across the noise variables for variance estimation. As the number of noise variables increases, the area of the space to explore grows exponentially, which demands an extensive amount of experiments. 3) this method assumes the discrete noise variables. One may discretize the continuous noise variables, but how to discretize will significantly affect the performance and computational complexity.

5.3 Constrained Bayesian Optimization on Robust Parameter Design

We assume that data are generated from a stochastic black-box function. *Response* $y(\mathbf{x})$ is the output obtained by a 'function evaluation' (i.e., by conducting an experiment). We want to find the optimal control variable value $\mathbf{x}$ that minimizes the variance and guarantees the mean of the response. For example, in manufacturing, one may want to find the setting of process parameters that minimize the variation of the product dimensional quality while maximizing its mean. Response $y(\mathbf{x})$ is a function of $\mathbf{x}$, and can be decomposed by the mean function $\mu_y(\mathbf{x})$ and error $\varepsilon(\mathbf{x})$.

$$y(\mathbf{x}) = \mu_y(\mathbf{x}) + \varepsilon(\mathbf{x}) \,, \tag{5.1}$$

where $\varepsilon(\mathbf{x})$ is a normal random error with mean zero and variance function $\sigma_y^2(\mathbf{x})$ (i.e., $\mathbb{V}\mathrm{ar}\, y(\mathbf{x}) = \mathbb{V}\mathrm{ar}\, \varepsilon(\mathbf{x}) = \sigma_y^2(\mathbf{x})$). Given a fixed $\mathbf{x}$, the response has constant underlying mean and variance while the mean and variance are functions of $\mathbf{x}$; that is, the variance is heterogeneous over different $\mathbf{x}$ values. Our goal is to minimize the log variance of the response $\sigma_y^2(\mathbf{x})$ while satisfying the constraint on the mean $\mu_y(\mathbf{x})$:

$$\begin{aligned} \min \quad & \log \sigma_y^2(\mathbf{x}) \\ \text{s.t.} \quad & \mu_y(\mathbf{x}) \leq u_y \end{aligned} \tag{5.2}$$

where u_y is the upper bound of the mean function. Solving (5.2) is challenging because we can only observe $y(\mathbf{x})$ but not $\mu_y(\mathbf{x})$ nor $\sigma_y^2(\mathbf{x})$; both $\mu_y(\mathbf{x})$ and $\sigma_y^2(\mathbf{x})$ need to be inferred from the observed values of $y(\mathbf{x})$. This type of constrained optimization formulation has been commonly used for the robust parameter design with dual response models because this formulation is intuitive and practical (Vining and Myers, 1990; Copeland and Nelson, 1996; Castillo et al., 1997; Fan, 2000; Köksoy and Doganaksoy, 2003). Without loss of generality, we use a constraint with an upper bound, but one can easily replace it by a lower bound or use both lower and upper bounds.

In the following sections, we propose a stochastic constrained Bayesian optimization method that finds the optimal solution of (5.2) only with a small number of experiments conducted. The Bayesian optimization method is a sequential optimization technique. The scheme initially starts with observed outputs at some $\mathbf{x}$. At each iteration, first, the proposed *surrogate models* (introduced in Section 5.3.1) are fitted to predict the mean $\mu_y(\mathbf{x})$ and variance $\sigma_y^2(\mathbf{x})$ based on the responses of $y(\mathbf{x})$ and quantify the uncertainties in the predictions. Then, a point to be used for the next experiment is suggested based on the proposed *acquisition function* (introduced in Section 5.3.2) leveraging the prediction and quantified uncertainty by the surrogate models. Then, after conducting experiments, the described iteration is repeated.

5.3.1 Mean and Variance Surrogate Models

We propose nonparametric dual stochastic response models that represent the mean (i.e, $\mu_y(\mathbf{x})$) and log variance (i.e., $\log \sigma_y^2(\mathbf{x})$) of the response $y(\mathbf{x})$. Because $\mu_y(\mathbf{x})$ and $\log \sigma_y^2(\mathbf{x})$ are not observable, they are modeled by the Gaussian processes where the *sample mean* and *log sample variance* of $y(\mathbf{x})$ are used for their observations. The conventional dual surface models, on the other hand, use two independent polynomial linear regressions to model the mean and variance functions; the strict assumptions of a quadratic form, homogeneous variance, and independent errors cannot account for the complex uncertainties in the surrogate models (Vining and Myers, 1990; Copeland and Nelson, 1996; Castillo et al., 1997; Fan, 2000; Köksoy and Doganaksoy, 2003). Besides, our method does not need to observe the noise variables nor their probability distribution to address the stochasticity in the response; In contrast, Lehman et al. (2004) modeled the *deterministic* simulators by a noiseless deterministic Gaussian process model with assumptions that the noise variables are observable and their probability mass function is known.

We denote $m(\mathbf{x})$ and $v(\mathbf{x})$ by the sample mean and log sample variance (i.e., $\log s^2$ where s^2 is sample variance) calculated by N sampled responses $y(\mathbf{x})$. $m(\mathbf{x})$ and $v(\mathbf{x})$ are modeled by the summation of the mean μ and error term δ that has zero mean and positive variance:

$$v(\mathbf{x}) = \mu_v(\mathbf{x}) + \delta_v \tag{5.3}$$

$$m(\mathbf{x}) = \mu_m(\mathbf{x}) + \delta_m(\mathbf{x}) \tag{5.4}$$

where $\mu_v(\mathbf{x})$ and $\mu_m(\mathbf{x})$ are the mean functions of $v(\mathbf{x})$ and $m(\mathbf{x})$, and δ_v and $\delta_m(\mathbf{x})$ are the normal random variables with mean zero. Because m is a unbiased estimator (i.e., $\mu_m(\mathbf{x}) = \mathbb{E}\, m(\mathbf{x}) = \mu_y(\mathbf{x})$) and the bias of v to $\log \sigma_y^2(\mathbf{x})$ is constant (i.e., $\mu_v(\mathbf{x}) = \mathbb{E}\, v(\mathbf{x}) = \log \sigma_y^2(\mathbf{x}) + b$; specifically, the bias b is a function of the sample size N; details can be found in Appendix 5.A.2.) Optimization formulation (5.2) is equivalent to $\min \mu_v(\mathbf{x})$ subject to $\mu_m(\mathbf{x}) \leq u_y$. However, $\mu_v(\mathbf{x})$ and $\mu_m(\mathbf{x})$ are not observable; therefore, they are modeled by the Gaussian process priors.

$$\mu_v(\mathbf{x}) \sim \mathcal{GP}(\mu_v^o(\mathbf{x}), \mathrm{cov}(\mathbf{x}, \mathbf{x}'; \boldsymbol{\theta}_v)) \tag{5.5}$$

$$\mu_m(\mathbf{x}) \sim \mathcal{GP}(\mu_m^o(\mathbf{x}), \mathrm{cov}(\mathbf{x}, \mathbf{x}'; \boldsymbol{\theta}_m)) \tag{5.6}$$

where $\mu_v^o(\mathbf{x})$ and $\mu_m^o(\mathbf{x})$ are the prior means of $\mu_v(\mathbf{x})$ and $\mu_m(\mathbf{x})$. $\mathrm{cov}(\cdot, \cdot; \boldsymbol{\theta})$ is a covariance function with the hyperparameters $\boldsymbol{\theta}$, and $\mathrm{cov}(\cdot, \cdot; \boldsymbol{\theta})$ is represented with a kernel function. We used the Gaussian kernel, which is one of the most widely used kernel (Rasmussen and Williams, 2006).

$$\mathrm{cov}\left(\mathbf{x}, \mathbf{x}'; \boldsymbol{\theta} = \left\{\sigma^2, \boldsymbol{l}\right\}\right) = \sigma^2 \exp\left(-\left\|\boldsymbol{l}^{\mathsf{T}}(\mathbf{x} - \mathbf{x}')\right\|^2\right) \tag{5.7}$$

where $\boldsymbol{l}$ is the inverse length-scale parameters vector that characterizes the sensitivity of the variables $\mathbf{x}$ to the Gaussian process outputs. In parameter design, we cannot assume the sensitivities of all the parameters to the outputs are the same. When a variable x in $\mathbf{x}$ is sensitive to the output, the corresponding l in $\boldsymbol{l}$ will be estimated as a large number. Insensitive x will have a small l. σ^2 is the Gaussian process variances, and $\boldsymbol{\theta} = \{\sigma^2, \boldsymbol{l}\}$ are to be estimated. The hyperparameters of the

Gaussian processes $\boldsymbol{\theta}_v$ and $\boldsymbol{\theta}_m$ are estimated by the maximum likelihood estimation.

δ_v and $\delta_m(\mathbf{x})$ denote the sampling uncertainty of the two estimators $v(\mathbf{x})$ and $m(\mathbf{x})$; the variances of $v(\mathbf{x})$ and $m(\mathbf{x})$ are characterized through the variances of δ_v and $\delta_m(\mathbf{x})$, respectively. The log sample variance of normal random variable has often been approximated by the normal distribution (Bartlett and Kendall, 1946; Nair and Pregibon, 1988).

$$\mathbb{V}\mathrm{ar}\,\delta_v = \frac{2}{N-1} \tag{5.8}$$

With this approximated variance, the Gaussian process model in (5.5) can be independently modeled and fitted. The mathematical derivation of (5.8) is given (in terms of N) in Appendix 5.A.1 of the supplementary material.

The variance of the sample mean is heterogeneous across $\mathbf{x}$ because $\mathbb{V}\mathrm{ar}\,\delta_m(\mathbf{x}) = \sigma_y^2(\mathbf{x})/N$, and we do not know the exact expression of $\sigma_y^2(\mathbf{x})$. It is important to characterize this heterogeneity to precisely quantify the uncertainty, which is crucial in Bayesian optimization. We predict this variance $\mathbb{V}\mathrm{ar}\,\delta_m(\mathbf{x})$ by using a plug-in estimator with the predictive mean of $v(\mathbf{x})$. Notice that $v(\mathbf{x})$ is a biased estimator of $\log\sigma_y^2(\mathbf{x})$ with a constant bias; $\mathbb{E}\,v(\mathbf{x}) = \log\sigma_y^2(\mathbf{x}) + b$ where the bias $b = -1/(N-1)$ is approximated by the Taylor expansion (See Appendix 5.A.1 of the supplementary material). Given these two equations, we obtain $\mathbb{V}\mathrm{ar}\,\delta_m(\mathbf{x}) = \exp(\mathbb{E}\,v(\mathbf{x}) - b)/N$. By plugging the predictive mean $\hat{\mu}_v(\mathbf{x})$ into $\mathbb{E}\,v(\mathbf{x})$, we obtain

$$\mathbb{V}\mathrm{ar}\,\delta_m(\mathbf{x}) = \frac{1}{N}\exp\left(\hat{\mu}_v(\mathbf{x}) + \frac{1}{N-1}\right) \tag{5.9}$$

where $\hat{\mu}_v(\mathbf{x})$ is the predictive mean of $v(\mathbf{x})$. The derivation of (5.9) is presented in Appendix 5.A.2 of the supplementary material.

Notice that the predictive mean $\hat{\mu}_v$ is used in (5.9). The predictive mean and variance, $\hat{\mu}_v$ and $\hat{\sigma^2}_v$, are obtained by the conditional posterior mean and variance. Suppose that $\mathbf{X} = [\mathbf{x}_1 \cdots \mathbf{x}_t]$ is the t control variable values used so far, and $\mathbf{v} = [v_1 \cdots v_t]^\mathsf{T}$ is the corresponding observed log sample variances. The predictive mean and variance at $\mathbf{x}^*$, given $\mathbf{X}$ and $\mathbf{v}$ are written as:

$$\hat{\mu}_v = \mu_v^o(\mathbf{x}^*) + C_{\mathbf{x}^*,\mathbf{X}} \left\{ C_{\mathbf{X},\mathbf{X}} + \mathbb{V}\mathrm{ar}\, \delta_v I \right\}^{-1} (\mathbf{v} - \mu_v^o(\mathbf{X})) \tag{5.10}$$

$$\hat{\sigma}_v^2 = C_{\mathbf{x}^*,\mathbf{x}^*} - C_{\mathbf{x}^*,\mathbf{X}} \left\{ C_{\mathbf{X},\mathbf{X}} + \mathbb{V}\mathrm{ar}\, \delta_v I \right\}^{-1} C_{\mathbf{X},\mathbf{x}^*} \tag{5.11}$$

where $C_{\mathbf{X},\mathbf{X}'} = C\left(\mathbf{X}, \mathbf{X}'; \hat{\boldsymbol{\theta}}\right)$ is the prior (cross-)covariance matrix of $\mathbf{X}$ and $\mathbf{X}'$, whose (i,j) element is $\mathrm{cov}\left(\mathbf{x}_i, \mathbf{x}'_j\right)$ where $\mathbf{x}_i$ and $\mathbf{x}'_j$ are the i and j th columns of $\mathbf{X}$ and $\mathbf{X}'$, respectively. Both predictive mean and variance of the Gaussian process surrogate models of v and m play important roles in the acquisition function. The predictive means predict the means of v and m (i.e., μ_v and μ_m in (5.3) and (5.4)), respectively, and the predictive variances $\hat{\sigma}_v^2$ and $\hat{\sigma}_m^2$ quantify the uncertainties in the predictions. Here, we show the predictive mean and variance of the surrogate models for v, but those of the surrogate model for m can also be obtained in the same manner.

Therefore, we used $N = 2$ for the sample mean m and log sample variance v.

5.3.2 Stochastic Constraints Bayesian Optimization Acquisition Function

We propose an acquisition function that explicitly encourages exploration over high uncertainty regions for the next experiment, presented in Section 5.3.2.1. The description of obtaining the current best predicted feasible optimal solution is detailed in Section 5.3.2.2.

5.3.2.1 Proposed acquisition function

In the Bayesian optimization, the point for the next experiment is selected based on the acquisition function. We propose an acquisition function that explicitly encourages exploration over the space with high uncertainty looking for a better solution. It also exploits the knowledge about the objective and constraint functions (i.e., optimality and feasibility). The proposed acquisition function $\alpha(\mathbf{x})$ is

$$\alpha(\mathbf{x}) = \hat{\sigma}_m(\mathbf{x})\, \Phi\left(\frac{u_y - \hat{\mu}_m(\mathbf{x})}{\hat{\sigma}_m(\mathbf{x})}\right) \Phi\left(\frac{v_{best} - \hat{\mu}_v(\mathbf{x})}{\hat{\sigma}_v(\mathbf{x})}\right) \tag{5.12}$$

where v_{best} is the current best objective value, and $\Phi(\cdot)$ is the standard normal cumulative density function. $\hat{\mu}_m(\mathbf{x})$ and $\hat{\sigma}_m(\mathbf{x})$ are the predictive mean and standard deviation of m at $\mathbf{x}$, and $\hat{\mu}_v(\mathbf{x})$ and $\hat{\sigma}_v(\mathbf{x})$ are those of v at $\mathbf{x}$. In (5.12), the second term $\Phi\left(\left(u_y - \hat{\mu}_m(\mathbf{x})\right)/\hat{\sigma}_m(\mathbf{x})\right)$ is the probability of feasibility, which is the probability that the given $\mathbf{x}$ is feasible under the predicted probability distribution by the mean surrogate model. Apparently, this term evaluates the feasibility of the given $\mathbf{x}$. The last term $\Phi\left(\left(v_{best} - \hat{\mu}_v(\mathbf{x})\right)/\hat{\sigma}_v(\mathbf{x})\right)$ is the probability of improvement, which

is the probability that the given x is better than the current best value v_{best}; this term evaluates the optimality of the given x. These two terms evaluate the optimality and feasibility conditions through probability functions. The most important term is the first term $\hat{\sigma}_m(\mathbf{x})$, which evaluates the uncertainty of the prediction. By selecting x that has the largest standard deviation, we can reduce the largest uncertainty about the response (Srinivas et al., 2012). Naturally, $\alpha(\mathbf{x})$ explicitly encourages exploration and filters out bad candidates (infeasible nor not improving the objective value) to avoid over-exploration. On the contrary, the weighted expected improvement (Gardner et al., 2014; Letham et al., 2019) is known as a greedy approach. The weighted expected improvement multiplies two terms, the expected improvement and the probability of feasibility, and these both terms significantly favor exploitation not performing enough exploration.

The setting for next experiment is selected at each iteration by maximizing the proposed acquisition function:

$$\mathbf{x}_{next} = \underset{\mathbf{x}}{\operatorname{argmax}}\, \alpha(\mathbf{x}) \tag{5.13}$$

The explicit exploration of our acquisition function provides many benefits: It facilitates finding a better solution quickly and finding a global optimum. Even after finding a good solution (or a local optimum), this acquisition function keeps searching unexplored regions. In addition, exploration improves the precision of the function estimation (Srinivas et al., 2012), which then improves the accuracy of the solutions.

In the following Section 5.3.2.2, the current best feasible solution v_{best} used in

(5.12) is described.

5.3.2.2 Current best predicted feasible optimal solution

We find the current best feasible solution $\mathbf{x}_{best}$ by the predicted objective and constraint functions then predict v_{best} at $\mathbf{x}_{best}$.

$$\mathbf{x}_{best} = \arg\min_{\mathbf{x}} \hat{\mu}_v(\mathbf{x}) + \lambda \max(0, \hat{\mu}_m(\mathbf{x}) - u_y) \tag{5.14}$$

$$v_{best} = \hat{\mu}_v(\mathbf{x}_{best}) \tag{5.15}$$

where λ is an arbitrary large number. The current best feasible point $\mathbf{x}_{best}$ is selected from the predictive Gaussian process means of the objective and constraint functions. Notice that the predictive mean is the empirical best linear unbiased predictor (EBLUP) (Santner et al., 2018). The second term in (5.14) gives the penalty to the feasibility gap, $\max(0, \hat{\mu}_m(\mathbf{x}) - u_y)$. The feasibility gap quantifies how much violated is the constraint; a feasible point has zero feasibility gap, and the feasibility gap of an infeasible point is the distance of the point from the upper bound. As a result, (5.14) finds the minimum point of the predicted objective function if there exist feasible points. If feasible points are not available, it finds the point with the smallest feasibility gap. v_{best} is used as the best feasible point in (5.12) in the next iteration, and $\mathbf{x}_{best}$ will serve as the best solution when the algorithm is terminated. We would like to point out that λ in (5.14) needs to be large enough. When there is no feasible region, the penalty weighs significantly on the feasibility gap; therefore,

we get a point close to the one with the minimum feasibility gap as the optimal solution of (5.14). Nevertheless, our preliminary experiments indicate that the values of λ barely affect the performance results as long as it is large enough, which is shown in Appendix 5.A.4 of the supplementary material.

For the optimization of the acquisition functions, there are many options. We used the DiRect (Dividing Rectangles) algorithm (Jones et al., 1993) to optimize the acquisition functions. DiRect algorithm is a deterministic global optimization (Jones and Martins, 2020) and is greatly used because it is fast and effectively finds the global optimum. DiRect has also been used for acquisition function optimization (Kandasamy et al., 2016, 2019).

The proof of the convergence of the proposed acquisition function is a challenging problem. In this work, instead of the proof, we validate the performance of our proposed method by the results from the extensive numerical and case studies. The results confirm the benefits of the proposed method.

The complete algorithm of the proposed method is summarized in Algorithm 2.

Algorithm 2 Stochastic constrained Bayesian optimization for robust parameter design

Input: : Budget for the number of experiments b, the number of initial data points M, initial observations $\mathcal{D}_0 = \{(\mathbf{x}_{0j}, y_{01j}, y_{02j}), j = 1 \ldots, M\}$, penalty λ, initial optimum value with an arbitrary large number $v_{best} = \lambda$.
Output: : $\mathbf{x}_{best}$
Initialize: : Obtain $\mathcal{H}_1 = \{(\mathbf{x}_{0j}, v_{0j}, m_{0j}), j = 1 \ldots, M\}$ from $\mathcal{D}_0$, $t = 1$.
while $t \leq \lfloor b/2 \rfloor$ **do**
  Estimate the Gaussian process parameters $\boldsymbol{\theta}_v$in (5.5) given $\mathcal{H}_t$ by the MLE.
  Estimate the Gaussian process parameters $\boldsymbol{\theta}_m$ in (5.6) given $\mathcal{H}_t$ by the MLE.
  $\mathbf{x}_t = \operatorname{argmax} \alpha(\mathbf{x})$ in (5.12).
  $\mathbf{x}_{best} = \arg\min_{\mathbf{x}} \hat{\mu}_v(\mathbf{x}) + \lambda \max(0, \hat{\mu}_m(\mathbf{x}) - u_y)$ in (5.14).
  $v_{best} = \hat{\mu}_v(\mathbf{x}_{best})$ in (5.15).
  Observe y_{t1} and y_{t2} at $\mathbf{x}_t$.
  Calculate v_t and m_t with y_{t1} and y_{t2}.
  $\mathcal{H}_{t+1} = \mathcal{H}_t \bigcup \{(\mathbf{x}_t, v_t, m_t)\}$.
  $t = t + 1$
end while

5.4 Numerical Study

We demonstrate the performance of our proposed method by using simulated data from synthetic problems. We used four synthetic problems that have been widely used for the performance comparison of the sequential optimization methods designed for expensive-to-evaluate data (Letham et al., 2019). Each problem consists of an objective function and a constraint function that are in complex shapes, and both objective and constraint functions have many peaks and valleys; therefore, there exist multiple local optimal solutions. Every synthetic problem consists of an objective function $f(\mathbf{x})$ and a constraint $g(\mathbf{x}) \leq u_g$. We tailored the problems for

the formulation of the robust parameter design:

$$\begin{aligned} \min \quad & \log f(\mathbf{x}) \\ \text{s.t.} \quad & g(\mathbf{x}) \leq u_g \end{aligned} \tag{5.16}$$

and we assume that the response $y(\mathbf{x})$ is generated from the normal distribution with mean $g(\mathbf{x})$ and standard deviation $f(\mathbf{x})$:

$$y(\mathbf{x}) \sim \mathcal{N}\Big(g(\mathbf{x})\,, \{f(\mathbf{x})\}^2\Big) \tag{5.17}$$

The robust parameter design has different characteristics from the general constrained sequential optimization settings. For example, function $f(\mathbf{x})$ must be positive as a standard deviation. We also want to ensure that the ratio of the mean to the standard deviation is not too large to make the problem practical. Accordingly, we rescaled the original functions but remained the shape of the original functions and constraints. The details of the used functions are included in Appendix 5.B.1 of the supplementary material. We assume that we cannot observe the true underlying functions $f(\mathbf{x})$ and $g(\mathbf{x})$ but only observe the responses $y(\mathbf{x})$ after function evaluations at $\mathbf{x}$. We used four synthetic problems as below.

1. Gardner1: Two dimensional functions, originally from Gardner et al. (2014). The objective function is wavy and thus has multiple local minimums. The feasible regions of the constraints have two separated regions.

2. Gardner2: Two dimensional functions, originally from Gardner et al. (2014). This problem has two separated small feasible regions.

3. Gelbart: Two dimensional functions, originally from Gelbart et al. (2014). Branin function is used as the objective function and the feasible region is a circle.

4. Hartmann: Six dimensional functions, originally from Jalali et al. (2017). This problem is included to show the performance in a high dimensional problem.

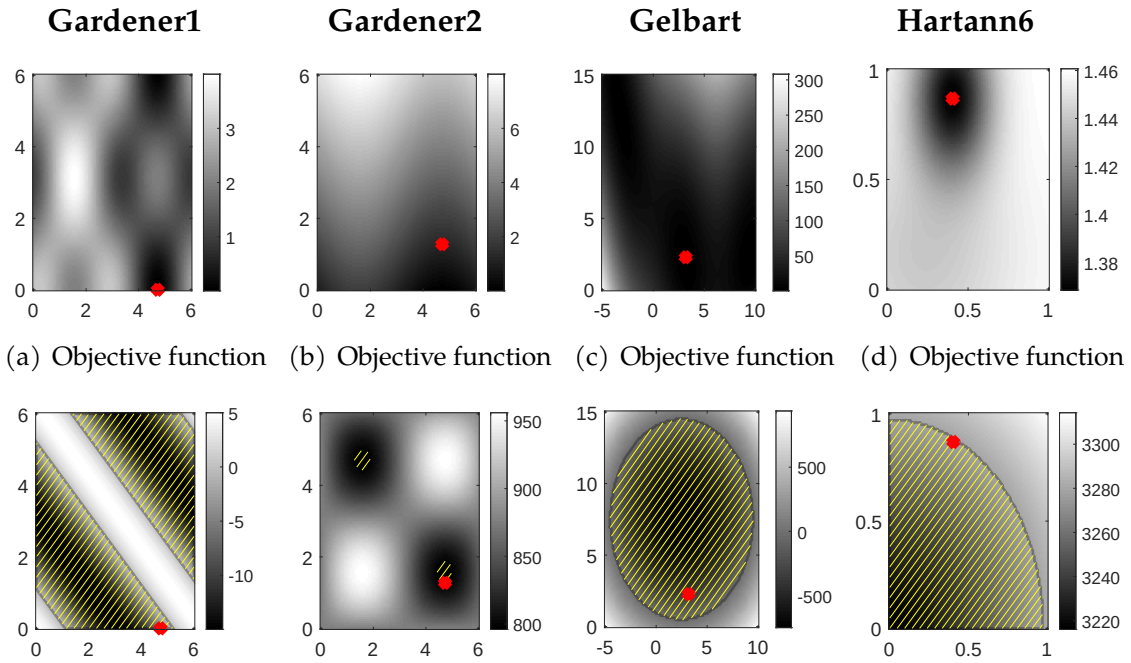
Fig. 5.4.1 shows the four synthetic problems used in the numerical study. The feasible regions are highlighted by the shades with diagonal lines.

We compared the performance of our method with four reference methods presented below.

- E2CBO: Our proposed method, "exploration-exploitation constrained Bayesian optimization". Dual stochastic response models presented in (5.3)-(5.6) are used for the surrogate models. The proposed acquisition function in (5.12) is used to find the next evaluation point, and (5.14) is used for the best feasible point. The DiRect search algorithm is used for the optimization of the acquisition functions.

- PS: Pattern search method, which is a derivative-free (i.e., no need to calculate gradient) global optimization technique for stochastic functions (Hooke and Jeeves, 1961). We used the tool provided by MATLAB. This method is included to show the performance of the general stochastic optimization technique.

- SoGP: Space-filling design with Gaussian process. The Sobol sequence, a sequential space-filling design method, is used to select the next evaluation



(a) Objective function (b) Objective function (c) Objective function (d) Objective function

(e) Constraint function (f) Constraint function (g) Constraint function (h) Constraint function

Figure 5.4.1: Four synthetic problems used in the numerical study where each column represents each problem. The first row presents the best feasible objective function, and the second row shows the constraint of each problem. The feasible regions are highlighted by the shades with parallel lines. The global feasible optimal point of each constrained optimization problem is marked with a dot. Hartmann6 is plotted over the first and second control variables while fixing the rest variables to 0.1.

point. Then, the Gaussian process models are fitted to the mean and log sample variance. The best feasible point is found by the predictive mean functions using (5.14) through the DiRect search algorithm. Space-filling design has been used in batch-based designs (Tan, 2015; Bektas et al., 2017) with Latin hypercube. For the sequential algorithm, we used the Sobol sequence instead. Space-filling designs only do exploration.

- HWEI: Heuristic version of the weighted expected improvement. To validate the performance of our proposed acquisition functions, existing acquisition functions are included for comparison. Please note that we used our proposed surrogate models presented in (5.3)-(5.6). HWEI is used as the acquisition function. Letham et al. (2019) modified WEI heuristically to use it as a reference method in their experiments using stochastic functions. In particular, HWEI obtains the best feasible point by the predictive mean of objective and constraint functions (while WEI picks the best feasible point among the observed data). WEI (accordingly, HWEI) is a greedy acquisition function, favoring exploitation.

- NWEI: Noisy weighted expected improvement. This is a stochastic version of WEI developed by Letham et al. (2019). However, NWEI is not efficient because of its sampling schemes. More importantly, like HWEI, NWEI is a greedy acquisition function, favoring exploitation. Please note that we used our proposed surrogate models, but NWEI is only used for the acquisition function to compare the performance of the acquisition functions.

We report the two performance indicators over the number of function evaluations: The *best feasible objective value* and the *feasibility gap*. In particular, the medians of these indicators over the replications of runs are presented. The best feasible objective value is the true value of the objective function $f(\mathbf{x}_{best})$ at the current best point $\mathbf{x}_{best}$ only when the mean constraint $g(\mathbf{x}_{best}) \leq u_g$ is satisfied. The median of the best feasible objective is calculated only with the feasible solutions. The feasibility gap indicates how much the feasibility is violated based on the true

constraint function value: $\max(0, g(\mathbf{x}_t) - u_g)$. If the current solution is feasible, the gap is 0, and if infeasible, the gap is the distance of the constraint function value from the upper bound: $g(\mathbf{x}_t) - u_g$.

These two indicators show the *optimality* and *feasibility* of the provided solution. There is a trade-off between optimality and feasibility; if a constraint is violated with a larger amount, one can easily achieve better objective values. Therefore, we provide the feasibility gap that measures how close the solution is to the feasible region. Because an infeasible point can have a better objective value, we use the best *feasible* objective value as an indicator for the optimality. Our goal is to achieve the minimum values in both the best feasible objective value and the feasibility gap. An algorithm that has a significantly large feasibility gap may have a better best feasible objective value, but this is not ideal because such an algorithm tends to violate the constraints to provide a better objective value that is rarely feasible.

We performed experiments with 100 replications. We used the sample size of two for the sample mean and the log sample variance. For each replication, two initial points of $\mathbf{x}$ are randomly selected (i.e., initially four evaluations are conducted). Then, optimization is carried out by each of the five methods with the same initial points. Only SoGP takes its initial points by the Sobol sequence. We set the budget of the function evaluations to the number of parameters multiplied by 100. For example, we limited our function evaluations to 200 for the two-dimensional functions, such as Gelbart, and 600 for the Hartmann6, a six-dimensional function.

Fig. 5.4.2 confirms that our proposed method significantly outperforms the reference methods. In general, the Bayesian optimization methods (E2CBO, HWEI,

and NWEI) outperformed the others (PS and SoGP); the Bayesian optimization methods decrease both the feasible objective value and the feasibility gap quicker than the other two methods. These results indicate that the Bayesian optimization approach using our proposed surrogate models is effective in the stochastic constrained optimization of RPD. Because our surrogate models can characterize not only the mean but also the uncertainty of the evaluated function, we can leverage both information to minimize the number of evaluations via Bayesian optimization. The performance of both SoGP and PS are not satisfactory. SoGP does not utilize the historical information for the next evaluation and only explores a new region. PS, on the other hand, uses the observed data for optimization but does not consider the estimation uncertainty. In our additional experiments, PS tended to converge eventually to the optimal solutions but after a significant number of evaluations.

We would like to highlight two observed advantageous features of our proposed acquisition function compared to the other two acquisition methods (HWEI and NWEI). First, our method quickly finds the better objective value. It reaches the minimum points in the both best feasible objective value and the feasibility gap the fastest. HWEI sometimes reaches the minimum feasibility gap quickly, but the feasible objective value was slower to get to the optimal solution. Second, our method tends to find the global optima while HWEI and NWEI stick to the local optima. Even when our method finds a local minimum, it keeps exploring to find a better solution. Our method not only keeps reducing the objective function but also the feasibility gap as more evaluations are conducted. On the other hand, HWEI and NEI were no longer able to decrease the feasibility gap in Gardner1

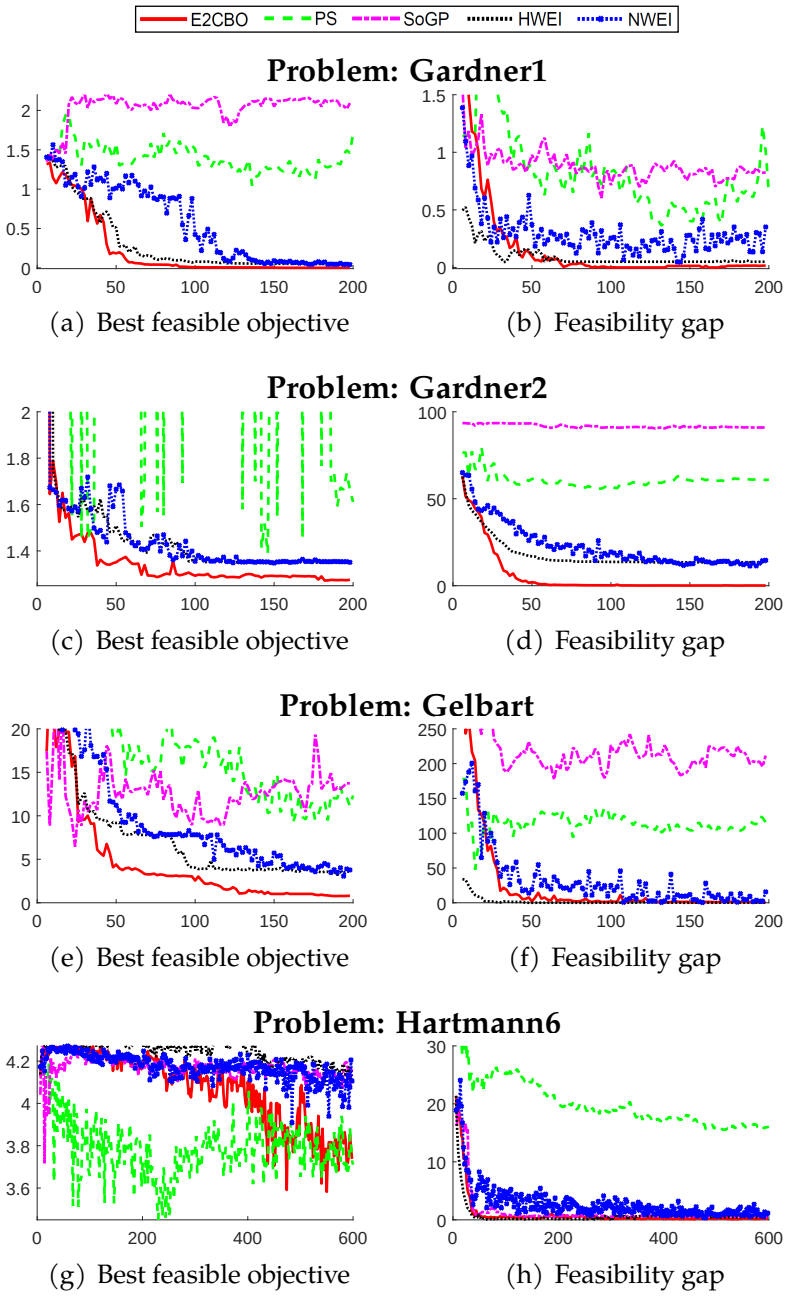


Figure 5.4.2: The performance of five methods with four synthetic problems. E2CBO is our proposed method. Each row shows the results from each problem. First column shows the best feasible objective values, and the second column shows the feasibility gap. They are shown over the number of function evaluations.

and 2 problems. In a six-dimensional problem, both HWEI and NWEI successfully reduced the feasibility gap but showed limitations in improving the objective values.

We also conducted the experiments by using different sample sizes for the sample mean and the log sample variance with all the methods presented here. The performance results are included in Appendix 5.B.2 of the supplementary material. Given the same budget for the number of function evaluations, the performance is the best with a sample size of 2 and better in a low sample size. This is because by using only a small sample size, we could effectively utilize the spared samples to explore a better solution.

5.5 Case Study

To demonstrate the performance of the proposed method with real-world data, we used the data generated from the reduced graphene oxide field-effect transistor (GFET) nanosensor model (Wang et al., 2021b); this nanosensor is designed to detect lead ions in water. Such graphene-based FET sensors have many promising beneficial characteristics and thus have been greatly developed to detect a wide range of target substances, such as heavy metals (Zhou et al., 2014), Escherichia coli (Thakur et al., 2018), and specific proteins (Liu et al., 2012). The GFET nanosensors are only for one-time use and can only be used once, and the sensor-to-sensor variations in structures are quite high (Lee et al., 2020). Naturally, the quality of the GFET sensors involves high variations. Therefore, finding the sensor design setting that has the minimum variance of the responses is important. Fig. 5.5.1 depicts

the diagram of the GFET sensor structure. The output of the GFET sensor is the current measured between the drain and source electrodes after a certain level of voltage is applied between these two electrodes. Once the sensor is exposed to the water containing the lead ions, the current drops. Then, the quality of the sensor is characterized by the "response ratio" (Maity et al., 2020).

$$y(\mathbf{x}) = \frac{I_b(\mathbf{x}) - I_t(\mathbf{x})}{I_b(\mathbf{x})} \tag{5.18}$$

where $\mathbf{x}$ is the control variable, and $I_b(\mathbf{x})$ and $I_t(\mathbf{x})$ are the drain-source current measured from the sensor in pure water and in the contaminated water with 20 ppb lead ions, respectively, given $\mathbf{x}$. Five control variables, length and width of the graphene channel, temperature, gate electrode thickness, and drain-source voltage are considered as the control variables. Eight noise variables are randomly generated and accordingly the stochastic output $y(\mathbf{x})$ is generated. The detailed simulation settings are included in Appendix 5.C.1 in the supplementary material. Even though the response ratio is deterministic given the values of the control and noise variables, we regarded $y(\mathbf{x})$ as stochastic and assumed the noise variables unobservable. The goal of this case study is to find the optimal setting of the control variables $\mathbf{x}$ that have the minimum variance of the response ratio $y(\mathbf{x})$ while satisfying that the mean response ratio is at least larger than a given tolerance; this mean tolerance is set by 0.07.

We followed similar settings of the numerical study in Section 5.4. We conducted 100 replicated experiments. For each replicated experiment, we randomly selected two initial samples, and all the methods shared these same initial samples (except

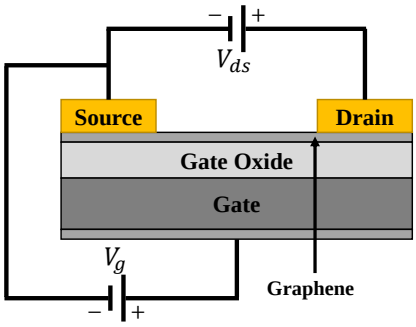
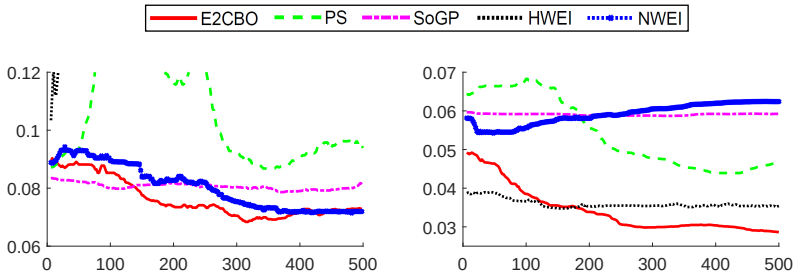


Figure 5.5.1: Graphene field-effect transistor sensor

SoGP that used Sobol sequence). The four reference methods used in the numerical studies were carried out to compare the performance of our proposed method. We present the best feasible objective value and feasibility gap. In particular, the median of these performance indicators over the experiment replications is presented. The major difference of the case study from the numerical study is that we do not know the true value of the mean nor the variance of the given function from which the data are generated. Thus, we obtained 100 samples and used their sample mean and sample variance as the indicators of the true mean and variance for validation. As a result, the resulting performance involved further sample variations in validation. To help readers interpret the figures, we smoothed the resulting graphs by the moving median with a sliding window of length 50. The original plots are presented in Appendix 5.C.2 of the supplementary material.

Fig. 5.5.2 depicts the best feasible objective values and the feasibility gaps of the five methods. The figure confirms that only our proposed method quickly finds the solution that has the minimum objective values and feasibility gap. All the reference methods did not perform well in this dataset. Among the reference



(a) best feasible objective (b) Feasibility gap

Figure 5.5.2: The performance of five methods in the case study. E2CBO is our proposed method. The best feasible objective values and feasibility gap over the number of evaluations are presented.

methods, some methods (SoGP and NWEI) reached low objective values quickly but the feasibility gaps of these methods were much higher than ours. HWEI found the solution with a low feasibility gap, but its objective value did not decrease but rather increased above out of the visualized range. PS decreased both the best feasible objective value and the feasibility gap, but the objective value was higher than PS and NWEI and the feasibility gap was higher than HWEI. Our method reached the lowest objective value along with NWEI and the feasibility gap was lower than any other methods.

We also included the case study results conducted using different sample sizes N for the sample mean and sample variance in Appendix 5.C.3 of the supplementary material. Similar to the numerical study, given the same number of function evaluations, the performance from a small sample size was better than those from a larger sample size.

5.6 Conclusion

In this work, we proposed a stochastic constrained Bayesian optimization method that minimizes the variance of the response while satisfying the constraint on its mean, namely the cosntrained robust parameter design. In particular, our method deals with the data generated from a stochastic black-box function. The dual stochastic response surface models are constructed as surrogate models to model the mean and log variance of the performance measures by the Gaussian processes. The sample mean and log sample variance of sample size 2 are used as the responses of the surrogate models. The approximated variance of the log sample variance is used for the log variance model. The heterogeneous variance of the sample mean is modeled by the plug-in estimator by the predictive mean of the log variance model. We also proposed an acquisition function that favors exploration across the control variable space while the optimality and feasibility of the point are taken into account. It encourages evaluating a high uncertainty region that is likely to be feasible and improve the objective value. The results from the numerical and case studies confirm that the performance of our proposed method is significantly better than the reference methods.

Our method widens the application range of RPD. Our proposed method can be used for real-world data or simulation data whose observations are stochastic and data generation function is complex and unknown. Our method automatically finds the best robust setting of engineering systems when it is incorporated into the target engineering system, such as simulators.

We did not prove the convergence of the proposed method. The convergence

proof of the Bayesian optimization is indeed challenging. The convergence of the conventional unconstrained acquisition functions has been proved recently, for such as the upper confidence bound (Srinivas et al., 2012) and posterior sampling (Russo and Roy, 2014). However, the convergence of the constrained Bayesian optimizations, such as the weighted expected improvement, has not been studied yet. We will further investigate this topic and plan to report the results in the near future.

APPENDIX

5.A Derivation of results with N sample size

5.A.1 Approximated variance of the log sample variance

N denotes sample size for the sample mean and log sample variance. $\mathbb{V}\mathrm{ar}\log(s^2)$ is approximated by the delta method. In the following proof, $f(\cdot) = \log(\cdot)$:

$$\mathbb{V}\mathrm{ar}\log(s^2) \approx \left(f'\left(\mathbb{E}\, s^2\right)\right)^2 \mathbb{V}\mathrm{ar}\, s^2 = \frac{1}{\left(\mathbb{E}\, s^2\right)^2}\mathbb{V}\mathrm{ar}\, s^2 \tag{5.19}$$

$$= \frac{1}{\left(\sigma^2\right)^2}\frac{2\sigma^4}{N-1} \tag{5.20}$$

$$= \frac{2}{N-1} \tag{5.21}$$

5.A.2 Approximated variance of the sample mean

First, we approximate the bias of log sample variance by the second order Taylor approximation.

$$\mathbb{E}\log\left(s^2\right) \approx \log\left(\mathbb{E}\, s^2\right) + \frac{1}{2}f''\left(\mathbb{E}\, s^2\right)\mathbb{V}\mathrm{ar}\, s^2 \tag{5.22}$$

$$= \log\left(\mathbb{E}\, s^2\right) - \frac{\mathbb{V}\mathrm{ar}\, s^2}{2\left\{\mathbb{E}\, s^2\right\}^2} \tag{5.23}$$

$$= \log\left(\sigma^2\right) - \left(\frac{2\sigma^4}{N-1}\right)\left\{\frac{1}{2\left(\sigma^2\right)^2}\right\} \tag{5.24}$$

$$= \log\left(\sigma^2\right) - \frac{1}{N-1} \tag{5.25}$$

where

$$\mathbb{V}\mathrm{ar}\, s^2 = \frac{2\sigma^4}{N-1} \tag{5.26}$$

because y follows the normal distribution.

Now, we know $\log\left(\sigma_y^2\right) \approx \mathbb{E}\log\left(s^2\right) + 1/(N-1)$ and $\mathbb{V}\mathrm{ar}\, m = \sigma_y^2/N$. Then, $\mathbb{V}\mathrm{ar}\, m \approx \exp\left(\mathbb{E}\log\left(s^2\right) + 1/(N-1)\right)/N$. Therefore, the plugged-in variance is

$$\mathbb{V}\mathrm{ar}\, m \approx \frac{1}{N}\exp\left(\hat{\mu}_{lv}(\mathbf{x}) + \frac{1}{N-1}\right) \tag{5.27}$$

where $\hat{\mu}_{lv}(\mathbf{x})$ is the posterior mean of the log sample variance model at $\mathbf{x}$.

5.A.3 Effectiveness of the approximated variance in Bayesian optimization

Approximated variance presented in Eq. (8) and (9) are lower than the true values by a constant. Nevertheless, they effectively characterize the heterogeneity of the variance of the sample mean. These variances play an important role in uncertainty estimation so that the algorithm explores the uncertain regions in sequential optimization. Therefore, the relative size of the variance over the control variable is the most important. It is often observed that the variances smaller than the true values are more effective in Bayesian optimization. When the prior variance is large, the prior mean dominates in the predictive mean when the sample size is small, which deteriorates prediction performance. To exploit the given observations

for effective prediction, many samples are required. This contradicts the goal of the Bayesian optimization that aims at using the least evaluated samples to find the optimal solution.

5.A.4 Sensitivity analysis over the penalty λ

We conducted the sensitivity analysis with two-dimensional synthesized problems used in our numerical studies by varying the penalty λ. Figure 5.A.1 indicates that the penalty λ in our proposed method does not affect the performance much.

5.B Numerical study

5.B.1 Synthesized problems

Below are the four synthesized problems used in the numerical study.

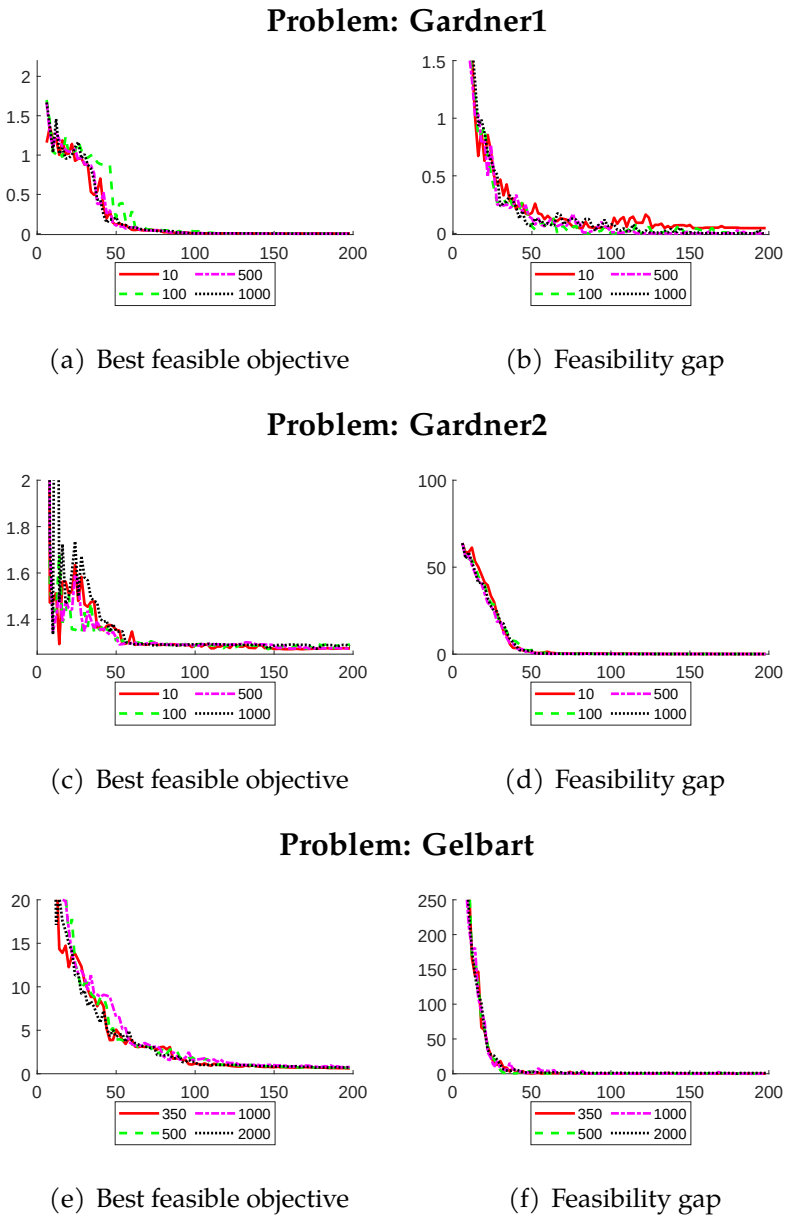
- Gardner1

$$f(x_1, x_2) = \cos(2x_1)\cos(x_2) + \sin(x_1) \tag{5.28}$$

$$g(x_1, x_2) = \cos(x_1)\cos(x_2) - \sin(x_1)\sin(x_2) - 0.5 \tag{5.29}$$

$$u_g = 0 \tag{5.30}$$

$$\mathbf{x} \in [0, 6]^2 \tag{5.31}$$



(a) Best feasible objective (b) Feasibility gap

(c) Best feasible objective (d) Feasibility gap

(e) Best feasible objective (f) Feasibility gap

Figure 5.A.1: Sensitivity analysis over the penalty λ. The minimum penalty is set larger than the maximum value of each problem.

- Gardner2

$$f(x_1, x_2) = \sin(x_1) + x_2 \tag{5.32}$$

$$g(x_1, x_2) = \sin(x_1)\sin(x_2) + 0.95 \tag{5.33}$$

$$u_g = 0 \tag{5.34}$$

$$\mathbf{x} \in [0, 6]^2 \tag{5.35}$$

- Gelbart

$$f(x_1, x_2) = \left(x_2 - \frac{5.1}{4\pi^2}x_1^2 + \frac{5}{\pi}x_1 - 6\right)^2 + 10\left(1 - \frac{1}{8\pi}\right)\cos(x_1) + 10 \tag{5.36}$$

$$g(x_1, x_2) = (x_1 - 2.5)^2 + (x_2 - 7.5)^2 - 50 \tag{5.37}$$

$$u_g = 0 \tag{5.38}$$

$$x_1 \in [-5, 10], x_2 \in [0, 15], \tag{5.39}$$

- Hartman6

$$f(\mathbf{x}) = -\sum_{i=1}^{4} \alpha_i \exp\left(-\sum_{j=1}^{6} A_{ij}(x_j - P_{ij})^2\right) \tag{5.40}$$

$$g(\mathbf{x}) = \|\mathbf{x}\| - 1, \tag{5.41}$$

$$u_g = 0 \tag{5.42}$$

$$\mathbf{x} \in [0, 1]^6 \tag{5.43}$$

with $\alpha = [1.01.23.03.2]$,

$$A = \begin{bmatrix} 10 & 3 & 17 & 3.5 & 1.7 & 8 \\ 0.05 & 10 & 17 & 0.1 & 8 & 14 \\ 3 & 3.5 & 1.7 & 10 & 17 & 8 \\ 17 & 8 & 0.05 & 10 & 0.1 & 14 \end{bmatrix}, \text{ and}$$

$$P = 10^{-4} \begin{bmatrix} 1312 & 1696 & 5569 & 124 & 8283 & 5886 \\ 2329 & 4135 & 8307 & 3736 & 1004 & 9991 \\ 2348 & 1451 & 3522 & 2883 & 3047 & 6650 \\ 4047 & 8828 & 8732 & 5743 & 1091 & 381 \end{bmatrix}$$

5.B.2 Results with different sample sizes

We conducted the numerical study with different sample size for the sample mean and log sample variance. The results are presented below. In summary, the performances with a lower sample size outperformed those with a higher sample size

5.B.2.1 Sample size 3

The numerical study results with sample size 3 is presented in Fig 5.B.1.

5.B.2.2 Sample size 5

The numerical study results with sample size 5 is presented in Fig 5.B.2.

5.B.2.3 Sample size 10

The numerical study results with sample size 10 is presented in Fig 5.B.3.

5.C Case study

5.C.1 Setting of the case study

We used the data generated from the reduced graphene oxide field-effect transistor (GFET) nanosensor model (Wang et al., 2021b). The output currents are simulated both from the pure water and from the water contaminated by 20 ppb lead ions. Then, the response ratio is calculated based on these two currents. The model takes five control variables, the length and width of the graphene channel, temperature, gate electrode thickness, and drain-source voltage. Eight noise variables are randomly generated; the response ratio requires two sensor currents, and four noise variables are used for each sensor current. The four noise variables are the hole and electron carrier mobility, the gate voltage at the point of minimum drain current, and the electron-hole puddle. The values of the noise variables for the pure water sensor current and contaminated water sensor current are obtained from Wang et al. (2021b). They are generated by the normal distribution; as a result, the created output currents contained random variations. To see the detailed equations for the sensor output, please refer to Wang et al. (2021b)

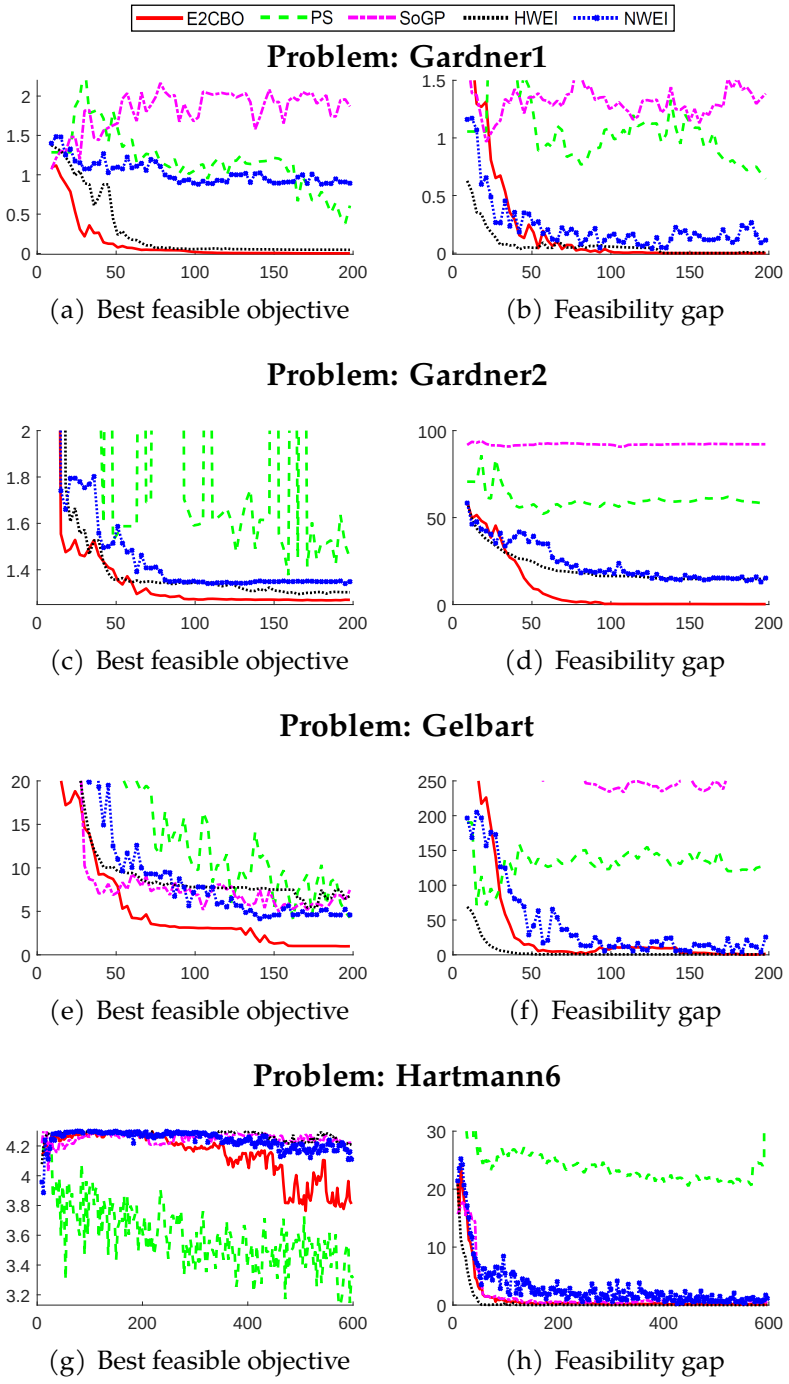
5.C.2 Results of the case study before smoothing

Fig 5.C.1 shows the case study results before smoothing.

5.C.3 Results with different sample sizes

Here, we provide the case study results with different sample sizes: 3, 5, 10. We made the resulting graphs smooth by the moving median with a sliding window of length 50.

The case study results with respect to sample size 3, 5, and 10 are presented in Fig 5.C.2, Fig 5.C.3, and Fig 5.C.4.



(a) Best feasible objective (b) Feasibility gap

(c) Best feasible objective (d) Feasibility gap

(e) Best feasible objective (f) Feasibility gap

(g) Best feasible objective (h) Feasibility gap

Figure 5.B.1: Performance of the five methods in the numerical study ($n = 3$)

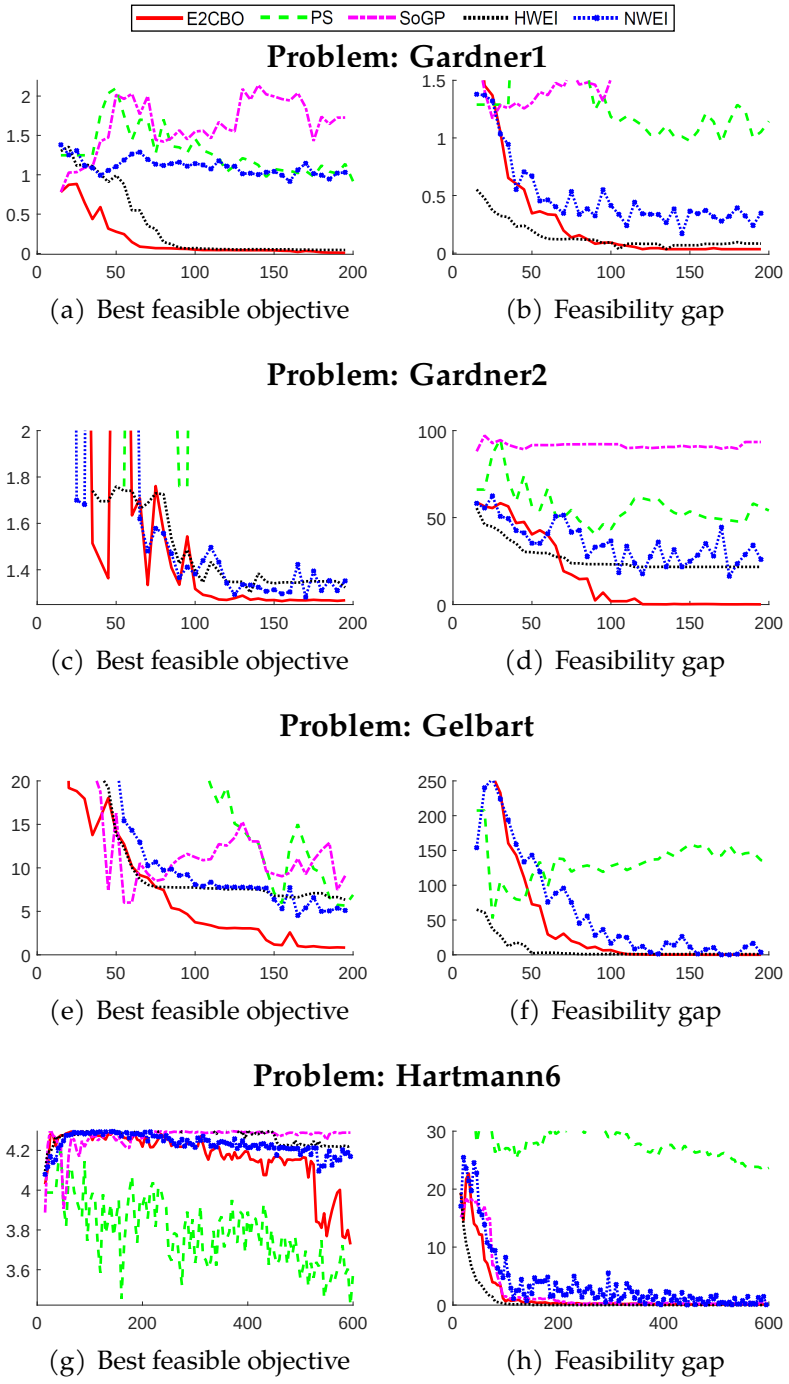
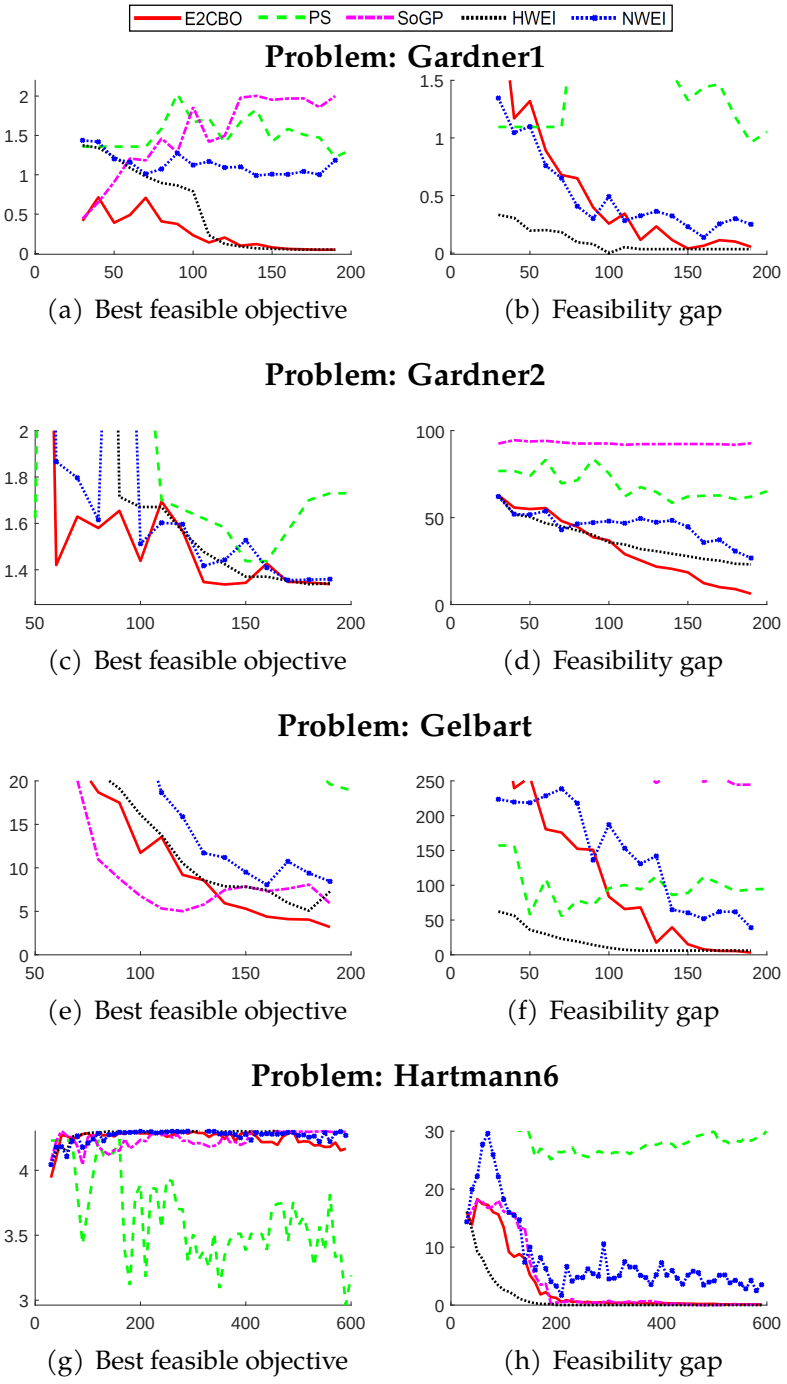


Figure 5.B.2: Performance of the five methods in the numerical study ($n = 5$)



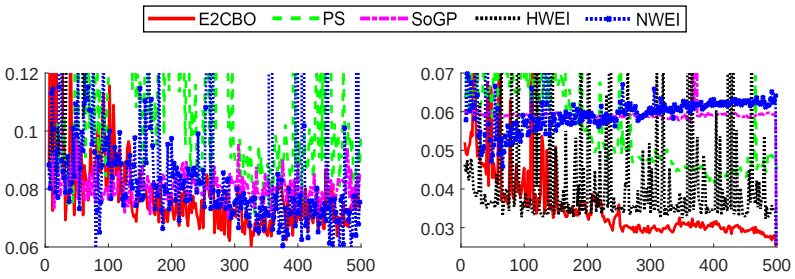
(a) Best feasible objective (b) Feasibility gap

(c) Best feasible objective (d) Feasibility gap

(e) Best feasible objective (f) Feasibility gap

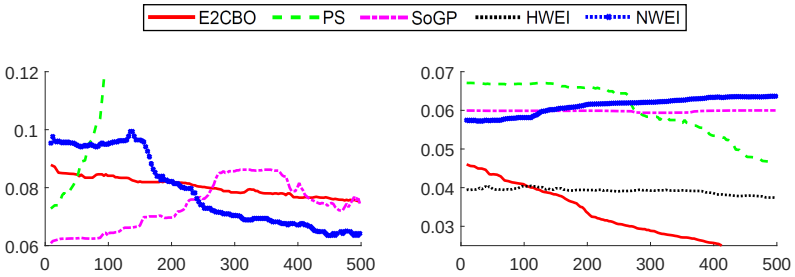
(g) Best feasible objective (h) Feasibility gap

Figure 5.B.3: Performance of the five methods in the numerical study ($n = 10$)



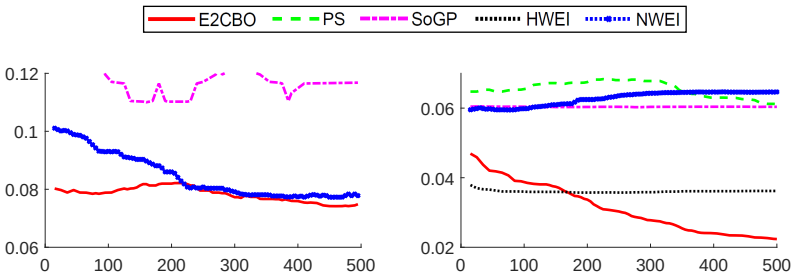
(a) Best feasible objective (b) Feasibility gap

Figure 5.C.1: Case Study results before smoothing $N = 2$



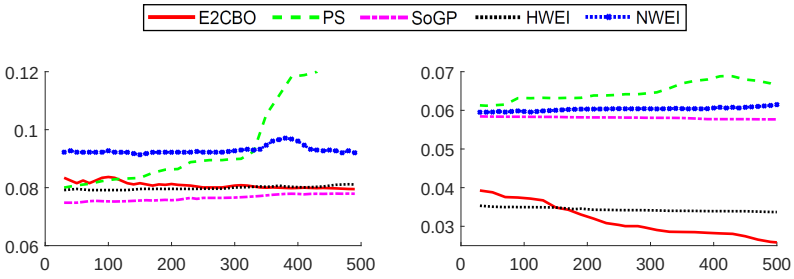
(a) Best feasible objective (b) Feasibility gap

Figure 5.C.2: Case Study results $N = 3$



(a) Best feasible objective (b) Feasibility gap

Figure 5.C.3: Case Study results $N = 5$



(a) Best feasible objective (b) Feasibility gap

Figure 5.C.4: Case Study results $N = 10$

6 CONCLUSION AND FUTURE WORK

In this dissertation, variations and uncertainties of the advanced manufacturing processes and systems are modeled. The contributions of this work are summarized as follows.

1. The proposed sparsity enhanced prior enables us to identify the faulty operation with excessive variations even when we have a large number of operations and do not have enough sensors. Our model is carefully designed to incorporate Engineering domain knowledge. The proposed sparsity-enhanced prior distribution can be flexibly adjusted according to the practitioners' empirical knowledge of the fault occurrence frequency and different tolerance levels across operations.

2. The random coverage and thickness of two-dimensional materials printed by the inkjet printing technique are modeled by stochastic geometry model, and the relationship between the process variables and the statistical model is also investigated. Furthermore, a statistic is established to identify abnormal patterns based on image data.

3. A nonparametric model that can precisely characterize the functional data with random shape and random shifting is proposed. Building upon the proposed model, a Bayesian framework to infer the underlying input variable is established. With the proposed model, the underlying variable of a new sampled signal is inferred accurately with precisely quantified uncertainties.

4. New stochastic constrained Bayesian optimization method is proposed for constrained robust parameter design, which aims to minimize the variance of the quality while satisfying a constraint on the mean quality. Specifically, for the Bayesian optimization, both the surrogate models and acquisition functions are established. The proposed surrogate models successfully characterize the heterogeneous variance of errors, and the proposed acquisition explicitly explores the design space to quickly find the global optimal solution not being satisfied by local optimums.

This dissertation mainly focused on establishing statistical models and methods that characterize and manage various types of uncertainties in the data from advanced manufacturing processes and systems. For future research, I plan to continue my research on the process and quality control in advanced manufacturing processes and systems as well as smart and connected service systems. The potential future research directions are as follows:

1. I plan to continue my research in establishing statistical and data mining models and methods on emerging manufacturing processes. There are many emerging manufacturing processes that need data-driven process and quality control methods. To address complex processes, I will consider using non-parametric statistical methods and machine learning methods such as neural networks. To characterize uncertainties in a large-scale dataset, Bayesian deep neural networks will be also considered.

2. By leveraging the historical data, I will investigate condition-based operation

and maintenance planning methods for smart and connected production systems. Especially when there are some dependencies between components (e.g., working machines) in the production systems, we need to consider the status of the entire components. I will integrate the Markov decision process, stochastic process, and reinforcement learning methods to find the optimal operation and maintenance plans.

3. I plan to extend my research applications to smart and connected service systems, such as healthcare management systems, where the data are obtained from individual clients/patients. In smart service systems, it is very important to predict the future status of clients/patients and make smart decisions. Specifically, I will investigate advanced nonparametric predictive modeling and decision-making models, such as partially observable Markov decision process models.

BIBLIOGRAPHY

Al-Aomar, R. (2006, April). Incorporating robustness into genetic algorithm search of stochastic simulation outputs. *Simulation Modelling Practice and Theory 14*(3), 201–223.

Apley, D. W., J. Liu, and W. Chen (2005, December). Understanding the effects of model uncertainty in robust design with computer experiments. *Journal of Mechanical Design 128*(4), 945–958.

Apley, D. W. and J. Shi (1998). Diagnosis of multiple fixture faults in panel assembly. *Journal of Manufacturing Science and Engineering 120*(4), 793.

Bartlett, M. S. and D. G. Kendall (1946). The statistical analysis of variance-heterogeneity and the logarithmic transformation. *Supplement to the Journal of the Royal Statistical Society 8*(1), 128–138.

Bastani, K., B. Barazandeh, and Z. J. Kong (2017). Fault diagnosis in multistation assembly systems using spatially correlated bayesian learning algorithm. *Journal of Manufacturing Science and Engineering 140*(3), 031003.

Bastani, K., Z. Kong, W. Huang, X. Huo, and Y. Zhou (2013). Fault diagnosis using an enhanced relevance vector machine (rvm) for partially diagnosable multistation assembly processes. *IEEE Transactions on Automation Science and Engineering 10*, 124.

Bastani, K., Z. J. Kong, W. Huang, and Y. Zhou (2016). Compressive sensing–based optimal sensor placement and fault diagnosis for multi-station assembly processes. *IIE Transactions 48*(5), 462–474.

Bektas, E., K. Broermann, G. Pecanac, S. Rzepka, C. Silber, and B. Wunderle (2017, April). Robust design optimization: On methodology and short review. In *2017 18th International Conference on Thermal, Mechanical and Multi-Physics Simulation and Experiments in Microelectronics and Microsystems (EuroSimE)*. IEEE.

Berk, J., V. Nguyen, S. Gupta, S. Rana, and S. Venkatesh (2019). Exploration enhanced expected improvement for bayesian optimization. In *Machine Learning and Knowledge Discovery in Databases*, pp. 621–637. Springer International Publishing.

Bhadra, A., J. Datta, N. G. Polson, and B. Willard (2017). The horseshoe+ estimator of ultra-sparse signals. *Bayesian Analysis 12*(4), 1105–1131.

Billingsley, P. (1995). *Probability and Measure*. Wiley.

Brochu, E., V. M. Cora, and N. de Freitas (2010, December). A tutorial on bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. eprint arXiv:1012.2599, arXiv.org.

Camelio, J., S. J. Hu, and D. Ceglarek (2003). Modeling variation propagation of multi-station assembly systems with compliant parts. *Journal of Mechanical Design 125*(4), 673.

Carvalho, C. M., N. G. Polson, and J. G. Scott (2009, 16–18 Apr). Handling sparsity via the horseshoe. In *Proceedings of the Twelth International Conference on Artificial*

Intelligence and Statistics, Volume 5 of *Proceedings of Machine Learning Research*, pp. 73–80. PMLR.

Carvalho, C. M., N. G. Polson, and J. G. Scott (2010, April). The horseshoe estimator for sparse signals. *Biometrika 97*(2), 465–480.

Castillo, E. D., S.-K. Fan, and J. Semple (1997, July). The computation of global optima in dual response systems. *Journal of Quality Technology 29*(3), 347–353.

Chang, J., H. Pu, S. A. Wells, K. Shi, X. Guo, G. Zhou, X. Sui, R. Ren, S. Mao, Y. Chen, M. C. Hersam, and J. Chen (2019). Semi-quantitative design of black phosphorous field-effect transistor sensors for heavy metal ion detection in aqueous media. *Molecular Systems Design & Engineering 4*(3), 491–502.

Chang, M. and D. Gossard (1998). Computational method for diagnosis of variation-related assembly problems. *International Journal of Production Research 36*(11), 2985–2995.

Chen, H., L. Zheng, R. AL Kontar, and G. Raskutti (2020). Stochastic gradient descent in correlated settings: A study on gaussian processes. In H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin (Eds.), *Advances in Neural Information Processing Systems*, Volume 33, pp. 2722–2733. Curran Associates, Inc.

Cheriet, M., J. Said, and C. Suen (1998, June). A recursive thresholding technique for image segmentation. *IEEE Transactions on Image Processing 7*(6), 918–921.

Chiu, S. N., D. Stoyan, W. S. Kendall, and J. Mecke (2013, August). *Stochastic Geometry and its Applications*. Wiley.

Copeland, K. A. F. and P. R. Nelson (1996, July). Dual response optimization via direct function minimization. *Journal of Quality Technology 28*(3), 331–336.

Deegan, R. D., O. Bakajin, T. F. Dupont, G. Huber, S. R. Nagel, and T. A. Witten (1997, October). Capillary flow as the cause of ring stains from dried liquid drops. *Nature 389*(6653), 827–829.

Ding, Y., D. Ceglarek, and J. Shi (2002). Fault diagnosis of multistage manufacturing processes by using state space approach. *Journal of Manufacturing Science and Engineering 124*, 313.

Ding, Y., S. Zhou, and Y. Chen (2005). A comparison of process variation estimators for in-process dimensional measurements and control. *Journal of dynamic systems, measurement, and control 127*, 69.

Dua, V., S. Surwade, S. Ammu, S. Agnihotra, S. Jain, K. Roberts, S. Park, R. Ruoff, and S. Manohar (2010, March). All-organic vapor sensor using inkjet-printed reduced graphene oxide. *Angewandte Chemie International Edition 49*(12), 2154–2157.

Fan, S.-K. S. (2000, October). A generalized global optimization algorithm for dual response systems. *Journal of Quality Technology 32*(4), 444–456.

Fang, K.-T., D. K. Lin, P. Winker, and Y. Zhang (2000, August). Uniform design: Theory and application. *Technometrics 42*(3), 237–248.

Farmanesh, B., A. Pourhabib, B. Balasundaram, and A. Buchanan (2020). A bayesian framework for functional calibration of expensive computational models through non-isometric matching. *IISE Transactions*, 1–13.

Fischer, B. J. (2002). Particle convection in an evaporating colloidal droplet. *Langmuir 18*(1), 60–67.

Gardner, J., G. Pleiss, K. Q. Weinberger, D. Bindel, and A. G. Wilson (2018). Gpytorch: Blackbox matrix-matrix gaussian process inference with gpu acceleration. In *Advances in Neural Information Processing Systems*, Volume 31.

Gardner, J. R., M. J. Kusner, Z. Xu, K. Q. Weinberger, and J. P. Cunningham (2014). Bayesian optimization with inequality constraints. In *Proceedings of the 31st International Conference on International Conference on Machine Learning - Volume 32*, ICML'14, pp. II–937–II–945. JMLR.org.

Gasser, T. and A. Kneip (1995). Searching for structure in curve samples. *Journal of the American Statistical Association 90*(432), 1179–1188.

Gelbart, M. A., J. Snoek, and R. P. Adams (2014). Bayesian optimization with unknown constraints. In *Proceedings of the Thirtieth Conference on Uncertainty in Artificial Intelligence*, UAI'14, Arlington, Virginia, USA, pp. 250–259. AUAI Press.

Gelman, A. (2006). Prior distributions for variance parameters in hierarchical models (comment on article by browne and draper). *Bayesian Analysis 1*(3), 515–534.

Gelman, A., J. Carlin, H. Stern, D. Dunson, A. Vehtari, and D. Rubin (2013a). *Bayesian Data Analysis, Third Edition*. Chapman & Hall/CRC Texts in Statistical Science. Taylor & Francis.

Gelman, A., J. B. Carlin, H. S. Stern, D. B. Dunson, A. Vehtari, and D. B. Rubin (2013b). Bayesian data analysis. *Chapman and Hall/CRC*.

Giovagnoli, A. and D. Romano (2008). Robust design via simulation experiments: a modified dual response surface approach. *Quality and Reliability Engineering International 24*(4), 401–416.

Goh, J., D. Bingham, J. P. Holloway, M. J. Grosskopf, C. C. Kuranz, and E. Rutter (2013). Prediction and computer model calibration using outputs from multifidelity simulators. *Technometrics 55*(4), 501–512.

Gramacy, R. B., D. Bingham, J. P. Holloway, M. J. Grosskopf, C. C. Kuranz, E. Rutter, M. Trantham, and R. P. Drake (2015). Calibrating a large computer experiment simulating radiative shock hydrodynamics. *The Annals of Applied Statistics 9*(3), 1141–1168.

Guo, X., Y. He, B. Zhu, Y. Yang, K. Deng, R. Liu, and C. Ji (2017). Bayesian uncertainty quantification for functional response. In *Uncertainty Quantification and Model Calibration*. InTech.

Han, D., B. S. Kim, and A. Chatterjee (2009). Dsp-driven self-tuning of rf circuits for process-induced performance variability. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems 18*(2), 305–314.

Hanley, J. and B. McNeil (1982). The meaning and use of the area under a receiver operating characteristic (roc) curve. *Radiology 143*(1), 29–36.

He, P. (2016). *Inkjet printing of two dimensional materials*. Ph. D. thesis, University of Manchester.

He, P. and B. Derby (2017, mar). Inkjet printing ultra-large graphene oxide flakes. *2D Materials 4*(2), 021021.

Herbrich, R., N. Lawrence, and M. Seeger (2003). Fast sparse gaussian process methods: The informative vector machine. In S. Becker, S. Thrun, and K. Obermayer (Eds.), *Advances in Neural Information Processing Systems*, Volume 15. MIT Press.

Higdon, D., J. Gattiker, E. Lawrence, C. Jackson, M. Tobis, M. Pratola, S. Habib, K. Heitmann, and S. Price (2013). Computer model calibration using the ensemble kalman filter. *Technometrics 55*(4), 488–500.

Hobert, J. P. and G. Casella (1996). The effect of improper priors on gibbs sampling in hierarchical linear mixed models. *Journal of the American Statistical Association 91*(436), 1461–1473.

Hooke, R. and T. A. Jeeves (1961, April). "direct search" solution of numerical and statistical problems. *Journal of the ACM 8*(2), 212–229.

Hsin, C. and M. Liu (2004). Network coverage using low duty-cycled sensors. In *Proceedings of the third international symposium on Information processing in sensor networks - IPSN'04*. ACM Press.

Hu, H. and R. G. Larson (2006, April). Marangoni effect reverses coffee-ring depositions. *The Journal of Physical Chemistry B 110*(14), 7090–7094.

Huang, D.-Y. and C.-H. Wang (2009, February). Optimal multi-level thresholding using a two-stage otsu optimization approach. *Pattern Recognition Letters 30*(3), 275–284.

Huang, L., Y. Huang, J. Liang, X. Wan, and Y. Chen (2011, March). Graphene-based conducting inks for direct inkjet printing of flexible conductive patterns and their applications in electric circuits and chemical sensors. *Nano Research 4*(7), 675–684.

Huang, W., J. Lin, M. Bezdecny, Z. Kong, and D. Ceglarek (2007). Stream-of-variation modeling—part i: A generic three-dimensional variation model for rigid-body assembly in single station assembly processes. *Journal of Manufacturing Science and Engineering 129*(4), 821.

Huber, B., P. B. Popp, M. Kaiser, A. Ruediger, and C. Schindler (2017, April). Fully inkjet printed flexible resistive memory. *Applied Physics Letters 110*(14), 143503.

Ingrassia, S. and I. Morlini (2005). Neural network modeling for small datasets. *Technometrics 47*(3), 297–311.

Jalali, H., I. V. Nieuwenhuyse, and V. Picheny (2017, August). Comparison of kriging-based algorithms for simulation optimization with heterogeneous noise. *European Journal of Operational Research 261*(1), 279–301.

Jasrasaria, D. and E. O. Pyzer-Knapp (2018, November). Dynamic control of explore/exploit trade-off in bayesian optimization. In *Advances in Intelligent Systems and Computing*, pp. 1–15. Springer International Publishing.

Jin, J. and J. Shi (1999a). Feature-preserving data compression of stamping tonnage information using wavelets. *Technometrics 41*(4), 327–339.

Jin, J. and J. Shi (1999b). State space modeling of sheet metal assembly for dimensional control. *Journal of Manufacturing Science and Engineering 121*(4), 756.

Jones, D. R. and J. R. R. A. Martins (2020, October). The DIRECT algorithm: 25 years later. *Journal of Global Optimization 79*(3), 521–566.

Jones, D. R., C. D. Perttunen, and B. E. Stuckman (1993, October). Lipschitzian optimization without the lipschitz constant. *Journal of Optimization Theory and Applications 79*(1), 157–181.

Jones, D. R., M. Schonlau, and W. J. Welch (1998). Efficient global optimization of expensive black-box functions. *Journal of Global Optimization 13*(4), 455–492.

Kandasamy, K., G. Dasarathy, J. Oliva, J. Schneider, and B. Póczos (2019, September). Multi-fidelity gaussian process bandit optimisation. *Journal of Artificial Intelligence Research 66*, 151–196.

Kandasamy, K., G. Dasarathy, B. Poczos, and J. Schneider (2016). The multi-fidelity multi-armed bandit. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett (Eds.), *Advances in Neural Information Processing Systems*, Volume 29. Curran Associates, Inc.

Kennedy, M. C. and A. O'Hagan (2001). Bayesian calibration of computer models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology) 63*(3), 425–464.

Khosravi, A., S. Nahavandi, D. Creighton, and A. F. Atiya (2011). Comprehensive review of neural network-based prediction intervals and new advances. *IEEE Transactions on Neural Networks 22*(9), 1341–1356.

Kim, Y. K., J. A. Park, W. H. Yoon, J. Kim, and S. Jung (2016, November). Drop-on-demand inkjet-based cell printing with 30-μm nozzle diameter for cell-level accuracy. *Biomicrofluidics 10*(6), 064110.

Köksoy, O. and N. Doganaksoy (2003, July). Joint optimization of mean and standard deviation using response surface methods. *Journal of Quality Technology 35*(3), 239–252.

Kontar, R., S. Zhou, C. Sankavaram, X. Du, and Y. Zhang (2018). Nonparametric modeling and prognosis of condition monitoring signals using multivariate gaussian convolution processes. *Technometrics 60*(4), 484–496.

Kotte, S., P. R. Kumar, and S. K. Injeti (2018, December). An efficient approach for optimal multilevel thresholding selection for gray scale images based on improved differential search algorithm. *Ain Shams Engineering Journal 9*(4), 1043–1067.

Lee, B. S., M. Haran, R. W. Fuller, D. Pollard, and K. Keller (2020). A fast particle-based approach for calibrating a 3-d model of the antarctic ice sheet. *Annals of Applied Statistics 14*(2), 605–634.

Lee, J., J. Son, S. Zhou, and Y. Chen (2020). Variation source identification in manu-

facturing processes using bayesian approach with sparse variance components prior. *IEEE Transactions on Automation Science and Engineering 17*(3), 1–17.

Lee, J., S. Zhou, and J. Chen (2020, September). Statistical modeling and analysis of k-layer coverage of two-dimensional materials in inkjet printing processes. *Technometrics 63*(3), 1–11.

Lehman, J. S., T. J. Santner, and W. I. Notz (2004). Designing computer experiments to determine robust control variables. *Statistica Sinica 14*(2), 571–590.

Lei, Y., Z. Zhang, and J. Jin (2010a). Automatic tonnage monitoring for missing part detection in multi-operation forging processes. *Journal of Manufacturing Science and Engineering 132*(5).

Lei, Y., Z. Zhang, and J. Jin (2010b). Automatic tonnage monitoring for missing part detection in multi-operation forging processes. *Journal of Manufacturing Science and Engineering 132*(5).

Letham, B., B. Karrer, G. Ottoni, and E. Bakshy (2019, June). Constrained bayesian optimization with noisy experiments. *Bayesian Analysis 14*(2), 495–519.

Li, J., M. C. Lemme, and M. Östling (2014, August). Inkjet printing of 2d layered materials. *ChemPhysChem 15*(16), 3427–3434.

Li, J., M. M. Naiini, S. Vaziri, M. C. Lemme, and M. Östling (2014, August). Inkjet printing of MoS2. *Advanced Functional Materials 24*(41), 6524–6531.

Li, S. and Y. Chen (2016). A bayesian variable selection method for joint diagnosis of manufacturing process and sensor faults. *IIE Transactions 48*(4), 313–323.

Li, Y., R. Torah, S. Beeby, and J. Tudor (2012, October). An all-inkjet printed flexible capacitor on a textile using a new poly(4-vinylphenol) dielectric ink for wearable applications. In *2012 IEEE Sensors*. IEEE.

Li, Z. and S. Zhou (2005). Robust method of multiple variation sources identification in manufacturing processes for quality improvement. *Journal of Manufacturing Science and Engineering 128*(1), 326–336.

Li, Z., S. Zhou, and Y. Ding (2007). Pattern matching for variation-source identification in manufacturing processes in the presence of unstructured noise. *IIE Transactions 39*(3), 251–263.

Liu, B., P. Brass, O. Dousse, P. Nain, and D. Towsley (2005). Mobility improves coverage of sensor networks. In *Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing - MobiHoc '05*. ACM Press.

Liu, B. and D. Towsley (2003, March). On the Coverage and Detectability of Large-scale Wireless Sensor Networks. In *WiOpt'03: Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks*, Sophia Antipolis, France.

Liu, B. and D. Towsley (2004). A study of the coverage of large-scale sensor networks. In *2004 IEEE International Conference on Mobile Ad-hoc and Sensor Systems* (*IEEE Cat. No.04EX975*). IEEE.

Liu, C., K. Wu, and V. King (2005). Randomized coverage-preserving scheduling schemes for wireless sensor networks. In *NETWORKING 2005. Networking Technologies, Services, and Protocols; Performance of Computer and Communication*

Networks; Mobile and Wireless Communications Systems, pp. 956–967. Springer Berlin Heidelberg.

Liu, Y., X. Dong, and P. Chen (2012). Biological and chemical sensors based on graphene materials. *Chem. Soc. Rev. 41*(6), 2283–2307.

Lu, Y., K. S. Subramani, H. Huang, N. Kupp, K. Huang, and Y. Makris (2015). A comparative study of one-shot statistical calibration methods for analog/rf ics. In *2015 IEEE International Test Conference (ITC)*, pp. 1–10. IEEE.

Maity, A., X. Sui, H. Pu, K. J. Bottum, B. Jin, J. Chang, G. Zhou, G. Lu, and J. Chen (2020). Sensitive field-effect transistor sensors with atomically thin black phosphorus nanosheets. *Nanoscale 12*(3), 1500–1512.

Maity, A., X. Sui, C. R. Tarman, H. Pu, J. Chang, G. Zhou, R. Ren, S. Mao, and J. Chen (2017). Pulse-driven capacitive lead ion detection with reduced graphene oxide field-effect transistor integrated with an analyzing device for rapid water quality monitoring. *ACS Sensors 2*(11), 1653–1661.

Makalic, E. and D. F. Schmidt (2016). A simple sampler for the horseshoe estimator. *IEEE Signal Processing Letters 23*(1), 179–182.

McKay, M. D., R. J. Beckman, and W. J. Conover (1979, May). A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics 21*(2), 239.

Merrin, J., S. Leibler, and J. S. Chuang (2007, July). Printing multistrain bacterial patterns with a piezoelectric inkjet printer. *PLoS ONE 2*(7), e663.

Molchanov, I. (2005). Random closed sets. In *Space, Structure and Randomness*, pp. 135–149. Springer.

Molchanov, I. (2006). *Theory of Random Sets*. Probability and Its Applications. Springer London.

Montgomery, D. (2012). *Statistical Quality Control, 7th Edition*. John Wiley & Sons.

Mosesova, S. (2007). Flexible mixed-effect modeling of functional data, with applications to process monitoring.

Nair, V. N. and D. Pregibon (1988, August). Analyzing dispersion effects from replicated factorial experiments. *Technometrics 30*(3), 247–257.

Novoselov, K. S. (2004, October). Electric field effect in atomically thin carbon films. *Science 306*(5696), 666–669.

Osborne, C. (1991). Statistical calibration: A review. *International Statistical Review / Revue Internationale de Statistique 59*(3), 309.

Otsu, N. (1979, January). A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics 9*(1), 62–66.

Ouyang, L., J. Chen, Y. Ma, C. Park, and J. J. Jin (2019, August). Bayesian closed-loop robust process design considering model uncertainty and data quality. *IISE Transactions 52*(3), 288–300.

Park, T. and G. Casella (2008). The bayesian lasso. *Journal of the American Statistical Association 103*(482), 681–686.

Perkins, N. J. and E. F. Schisterman (2006). The inconsistency of "optimal" cut-points using two roc based criteria. *American journal of epidemiology 163*(7), 670–675.

Picknett, R. and R. Bexon (1977, September). The evaporation of sessile or pendant drops in still air. *Journal of Colloid and Interface Science 61*(2), 336–350.

Plumlee, M., V. R. Joseph, and H. Yang (2016). Calibrating functional parameters in the ion channel models of cardiac cells. *Journal of the American Statistical Association 111*(514), 500–509.

Polson, N. G. and J. G. Scott (2011). Shrink globally, act locally: Sparse bayesian regularization and prediction*. *Bayesian Statistics 9*, 501–538.

Ramsay, J. O., P. Gribble, and S. Kurtek (2014). Analysis of juggling data: Landmark and continuous registration of juggling trajectories. *Electronic Journal of Statistics 8*(2), 1835–1841.

Ramsay, J. O. and B. W. Silverman (2005). *Functional Data Analysis*. Springer New York.

Rasmussen, C. E. and C. K. Williams (2006). *Gaussian processes for machine learning*, Volume 1. MIT press Cambridge.

Reza, S., N. Martin, T. Buchheit, and J. D. Tucker (2020). Tolerance bound calculation for compact model calibration using functional data analysis. In *2020 4th IEEE Electron Devices Technology & Manufacturing Conference (EDTM)*. IEEE.

Rong, Q., D. Ceglarek, and J. Shi (1999). Dimensional fault diagnosis for compliant beam structure assemblies. *Journal of Manufacturing Science and Engineering 122*(4), 773–780.

Russo, D. and B. V. Roy (2014, November). Learning to optimize via posterior sampling. *Mathematics of Operations Research 39*(4), 1221–1243.

Santner, T. J., B. J. Williams, W. Notz, and B. J. Williams (2003). *The design and analysis of computer experiments*, Volume 1. Springer.

Santner, T. J., B. J. Williams, and W. I. Notz (2018). *The Design and Analysis of Computer Experiments*. New York, Ny Springer New York.

Science and Technology Policy Institute (2010). White papers on advanced manufacturing questions.

Shen, W. (2017, July). Robust parameter designs in computer experiments using stochastic approximation. *Technometrics 59*(4), 471–483.

Shi, J. and S. Zhou (2009). Quality control and improvement for multistage systems: A survey. *IIE Transactions 41*(9), 744–753.

Shin, K.-Y., J.-Y. Hong, and J. Jang (2011, April). Micropatterning of graphene sheets by inkjet printing and its wideband dipole-antenna application. *Advanced Materials 23*(18), 2113–2118.

Singh, M., H. M. Haverinen, P. Dhagat, and G. E. Jabbour (2010, February). Inkjet printing-process and its applications. *Advanced Materials 22*(6), 673–685.

Snoek, J., H. Larochelle, and R. P. Adams (2012). Practical bayesian optimization of machine learning algorithms. In *Advances in neural information processing systems*, pp. 2951–2959.

Sowade, E., E. Ramon, K. Y. Mitra, C. Martínez-Domingo, M. Pedró, J. Pallarès, F. Loffredo, F. Villani, H. L. Gomes, L. Terés, and R. R. Baumann (2016, September). All-inkjet-printed thin-film transistors: manufacturing process reliability by root cause analysis. *Scientific Reports 6*(1).

Srinivas, N., A. Krause, S. M. Kakade, and M. W. Seeger (2012, May). Information-theoretic regret bounds for gaussian process optimization in the bandit setting. *IEEE Transactions on Information Theory 58*(5), 3250–3265.

Sui, Y. and J. Appenzeller (2009, August). Screening and interlayer coupling in multilayer graphene field-effect transistors. *Nano Letters 9*(8), 2973–2977.

Taguchi, G. (1986). *Introduction to quality engineering: designing quality into products and processes*. Tokyo: The Organization.

Tan, M. H. Y. (2015, October). Robust parameter design with computer experiments using orthonormal polynomials. *Technometrics 57*(4), 468–478.

Tan, M. H. Y. and C. J. Wu (2012, February). Robust design optimization with quadratic loss derived from gaussian process models. *Technometrics 54*(1), 51–63.

Thakur, B., G. Zhou, J. Chang, H. Pu, B. Jin, X. Sui, X. Yuan, C.-H. Yang, M. Magruder, and J. Chen (2018). Rapid detection of single e. coli bacteria using a graphene-based field-effect transistor device. *Biosensors and Bioelectronics 110*, 16–22.

Tsin, Y., V. Ramesh, and T. Kanade (2001). Statistical calibration of ccd imaging process. In *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001*, Volume 1, pp. 480–487. IEEE.

Vanli, O. A. and E. D. Castillo (2009, February). Bayesian approaches for on-line robust parameter design. *IIE Transactions 41*(4), 359–371.

Vining, G. G. and R. H. Myers (1990, January). Combining taguchi and response surface philosophies: A dual response approach. *Journal of Quality Technology 22*(1), 38–45.

Wan, P.-J. and Chih-Wei (2006, June). Coverage by randomly deployed wireless sensor networks. *IEEE Transactions on Information Theory 52*(6), 2658–2669.

Wand, M. P., J. T. Ormerod, S. A. Padoan, and R. Frühwirth (2011). Mean field variational bayes for elaborate distributions. *Bayesian Analysis 6*(4), 847–900.

Wang, C., H. Pu, X. Sui, S. Zhou, and J. Chen (2021a). Hybrid modeling and sensitivity analysis on reduced graphene oxide field-effect transistor. *IEEE Transactions on Nanotechnology 20*, 404–416.

Wang, C., H. Pu, X. Sui, S. Zhou, and J. Chen (2021b). Hybrid modeling and sensitivity analysis on reduced graphene oxide field-effect transistor. *IEEE Transactions on Nanotechnology 20*, 404–416.

Wang, S., Z. Jin, X. Huang, S. Peng, D. Zhang, and J. Shi (2016). Abnormal dirac point shift in graphene field-effect transistors. *Materials Research Express 3*(9), 095602.

Williams, C. K. and C. E. Rasmussen (2006). *Gaussian processes for machine learning*, Volume 2. MIT press Cambridge, MA.

Yang, H.-C., C.-H. Huang, M. Adnan, C.-H. Hsu, C.-H. Lin, and F.-T. Cheng (2021, January). An online AM quality estimation architecture from pool to layer. *IEEE Transactions on Automation Science and Engineering 18*(1), 269–281.

Yanıkoğlu, İ., D. den Hertog, and J. P. C. Kleijnen (2015, August). Robust dual-response optimization. *IIE Transactions 48*(3), 298–312.

Yi, G., J. Q. Shi, and T. Choi (2011). Penalized gaussian process regression and classification for high-dimensional nonlinear data. *Biometrics 67*(4), 1285–1294.

Zeng, L. and N. Chen (2015). Bayesian hierarchical modeling for monitoring optical profiles in low-e glass manufacturing processes. *IIE Transactions 47*(2), 109–124.

Zhan, D. and H. Xing (2020, July). Expected improvement for expensive optimization: a review. *Journal of Global Optimization 78*(3), 507–544.

Zhang, C., H. Yan, S. Lee, and J. Shi (2018). Weakly correlated profile monitoring based on sparse multi-channel functional principal component analysis. *IISE Transactions 50*(10), 878–891.

Zhong, J., J. Shi, and J. Wu (2010, April). Design of DOE-based automatic process controller with consideration of model and observation uncertainties. *IEEE Transactions on Automation Science and Engineering 7*(2), 266–273.

Zhou, G., J. Chang, S. Cui, H. Pu, Z. Wen, and J. Chen (2014). Real-time, selective detection of pb^{2+} in water using a reduced graphene oxide/gold nanoparticle

field-effect transistor device. *ACS Applied Materials & Interfaces 6*(21), 19235–19241.

Zhou, S., Y. Chen, and J. Shi (2004). Statistical estimation and testing for variation root-cause identification of multistage manufacturing processes. *IEEE Transactions on Automation Science and Engineering 1*, 73.

Zhou, S., Y. Ding, Y. Chen, and J. Shi (2003). Diagnosability study of multistage manufacturing processes based on linear mixed-effects models. *Technometrics 45*(4), 312–325.