FAST METHODS FOR IDENTIFYING HIGH DIMENSIONAL SYSTEMS USING OBSERVATIONS

A Thesis
Presented to
The Academic Faculty

by

Matthew Plumlee

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Industrial and Systems Engineering

Georgia Institute of Technology
May 2015

FAST METHODS FOR IDENTIFYING HIGH DIMENSIONAL SYSTEMS USING OBSERVATIONS

Approved by:

Jianjun Shi, Advisor
School of Industrial and Systems Engineering
Georgia Institute of Technology

Roshan Joseph Vengazhiyil, Advisor
School of Industrial and Systems Engineering
Georgia Institute of Technology

Chien-Fu Jeff Wu
School of Industrial and Systems Engineering
Georgia Institute of Technology

Kamran Paynabar
School of Industrial and Systems Engineering
Georgia Institute of Technology

Richard K. Archibald
Scientist, Oak Ridge National Laboratory
Georgia Institute of Technology

Date Approved: March 26, 2015

To my parents,

M. Katherine Banks and A. Paul Schwab,

Marlene A. Plumlee and R. David Plumlee,

who blazed the trail for my journey.

And to Kori,

who saw me through it.

ACKNOWLEDGEMENTS

I would like to express gratitude to my advisors, Professors Jianjun Shi and Roshan Joseph Vengazhiyil, for their guidance and encouragement throughout my studies. While their academic knowledge enabled my success, their care guaranteed it. Their tireless commit to students will always be a model to me. I am also thankful to Professor C. F. Jeff Wu, a brilliant philosopher, who greatly inspired me many times. Thank you to Prof Kamran Paynabar, for participating and guiding the act of research every week through our meetings. Thank you to Richard Archibald, for his fresh perspective on a subject that is always refreshing. I am also grateful to Doctor Rui Tuo for his tremendous help, without which this work would be lacking.

I would like to thank Professors Gary Parker, Paul Kvam, and Alan Erera, for giving me the continued opportunity and support to succeed in their graduate program. Thank you also to professors Santanu Dey, Antonius Dieker, David Goldberg, and Ben Haaland; who have always been willing to discuss the next step in my career.

Thank you to the the ARCS foundation (specifically the Love family), Mary G. and Joseph Natrella, Ellis R. Ott, and ISyE donors Tennenbaum and Morris. Without your generous support, work like this would not be possible. Also, thank you to the National Science Foundation, the Department of Energy, and Oak Ridge National Labs for supporting my work.

I would like to thank all the staff members of ISyE, Georgia Tech, especially Warren Bell, Jennifer Harris, Yijun (May) Li, Pamela Morrison, Dima Nazzal, Judith Norback, Anita Race, Mark Reese, and Yvonne Smith for their kind support and help at every stage of my graduate life at Georgia Tech.

Lastly, I am very thankful to my colleagues: Isil Alev, Rodolfo Carvajal, Bahar

Cavdar, Chia-Jung Chang, Andres Iroume, Tugce Isik, Ran Jin, Brain Kues, Jin Lee, Nolan Leung, Kaibo Liu, Deigo Moran, Carl Morris, Mallory Nobles, Dimitri Papageorgiou, Norbert Remenyi, Chang-han Rhee, Soheil Shayegh, Daniel Silva, Mallory Soldner, Timothy Sprock, Stefania Helga Stefansdottir, Rodrigue Ngueyep Tzoumpe, Jan Valchy, Monica Villarreal, Yijie (Dylan) Wang, Tonya Woods, and Fiona Xiao, and many others. Thank you for sharing ideas, laughter and drinks with me over the last five years.

Contents

List of Tables

List of Figures

SUMMARY

Computational modeling is a popular tool to understand a diverse set of complex systems. The output from a computational model depends on a set of parameters which are unknown to the designer, but a modeler can estimate them by collecting physical data. In the second chapter of this thesis, we study the action potential of ventricular myocytes and our parameter of interest is a function as opposed to a scalar or a set of scalars. We develop a new modeling strategy to nonparametrically study the functional parameter using Bayesian inference with Gaussian process priors. We also devise a new Markov chain Monte Carlo sampling scheme to address this unique problem.

In the more general case, computational simulation is expensive. Emulators avoid the repeated use of a stochastic simulation by performing a designed experiment on the computer simulation and developing a predictive distribution. Random field models are considered the standard in analysis of computer experiments, but the current framework fails in high dimensional scenarios because of the cost of inference. The third chapter of this thesis shows by using a class of experimental designs, the computational cost of inference from random fields scales significantly better in high dimensions. Exact prediction and likelihood evaluation with close to half a million design points is possible in seconds using only a laptop computer. Compared to the more common space-filling designs, the proposed designs are shown to be competitive in terms of prediction accuracy through simulation and analytic results.

The fourth chapter of this thesis proposes a method to construct an emulator for a stochastic simulation. Existing emulators have focused on estimation of the

mean of the simulation output, but this work presents an emulator for the distribution of the output in a nonparametric setting. This construction provides both an explicit distribution and a fast sampling scheme. Beyond describing the emulator, this work demonstrates that the emulator's convergence rate is asymptotically rate optimal among all possible emulators using the same sample size. Lastly, the fifth chapter of this work investigates the use of a modified version of the above method to study patterns of defects on products. We achieve efficient inference on the defect patterns by developing a novel estimate of an inhomogeneous point process that is both computationally tractable and asymptotically appealing.

Chapter I

INTRODUCTION AND BACKGROUND

Computer, or mathematical, models of physical systems have become indispensable tools in the analysis of complex systems. These models use a collection of simple principles, such as how an agent acts, how a fluid moves, or how a material's structure changes, to infer on system level behavior. Models based on physics/mechanics of the process are often realized via finite element simulation. Discrete event and agent based simulation are other examples that have had tremendous impact on many academic disciplines, and moreover society.

This thesis covers methods to identify and understand these computer models using physical data. Two major obstacles exist: (i) the computational cost of the simulation is high and (ii) the interface between the data and the underlying model is complicated by noise and model-inexactness. In this thesis, the first problem will be mitigated using an emulator, a tool used to reduce amount of times simulation of a model is needed. The major gains in this work are the design of emulators for use in cases of high dimensional inputs and noisy outputs. The second challenge is domain dependent. This thesis includes two case studies that involve the explicit methods to interface noisy observations with the underlying model of interest.

1.1 A motivating example, calibration of parameters of a ion channel model

The practice of positing and verifying a computational model is commonly used to investigate hypotheses about complex objects. Often, these models are defined up to a set of unknown parameters. Then these parameters are estimated so that the model's response aligns with observations [16, 67, 99, 56, 4, 96, 63]. The knowledge of

the parameters' exact value provides not only a better predictive model but furthers the modelers' understanding of the system. Parameters discussed in the works of [9] and [56] include important physical constants such as melting temperatures and reaction rates.

For our case study on the ion channels in cardiac cells, the methods cited above proved insufficient. Typically, one first isolates some set of parameters in the conjectured model. As described in the second chapter of this thesis, the parameters for recently proposed models for our system originated from empirically defined functions. Instead of acquiescing in this formulation, this work considers the parameter of interest to be a function. We have data from a physical experiment that holds the input to the functional parameter constant and captures the behavior of the response. We infer about the functional parameter via a posterior distribution which melds the observations with a functional prior distribution.

In this case we have a huge dimensional, in this case functional, parameter. Fortunately, in this case study, our computer model can be evaluated quickly as it is a first order ordinary differential equation. In the general case, we are often not as lucky. The following section detail methods to resolve this issue

1.2 Gaussian processes and computer experiment design

Consider a case where a deterministic output can be observed corresponding to a controllable input and the cost of an observation is expensive or at least non-negligible. Analysis that requires a huge number of evaluations of the expensive function for different inputs can prove impractical. This work examines a method to avoid the impracticality problem with the creation of a function that behaves similarly to the function of interest with a relatively cheap evaluation cost. We term this cheap function a *predictor* as it can closely match the output for an untried input. The predictor can be used in place of the expensive function for subsequent analysis.

Beginning in the 1980s, research has emphasized the use of *Gaussian process* models to construct predictors of the expensive function [105]. This method, often referred to as *kriging*, sprouted in geostatistics [77] and is considered the standard approach to study expensive, deterministic functions. A great deal of attention has been paid to this method and important variations over the last two decades because of the increased emphasis on computer simulation [67, 106, 56, 47]. The major objectives for analysis outlined in [105] have remained basically constant: predict the output given inputs, optimize the function, and adjust inputs of the function to match observed data. Recently, researchers have studied a fourth objective of computing the uncertainty of the output when inputs are uncertain, a topic in the broad field of uncertainty quantification. All of these objectives can be achieved through the use of a predictor, though sometimes under different names, e.g. *emulator* or *interpolator*.

As summarized in [105], a predictor is constructed by assuming that the output, termed $y(\cdot)$, is a realization of an unknown, random function of a d dimensional input $\boldsymbol{x}$ in a space $X \subset \mathbb{R}^d$. One notational comment: each element in an input $\boldsymbol{x}$ is denoted $x^{(j)}$, i.e. $\boldsymbol{x} = [x^{(1)}, x^{(2)}, \ldots, x^{(d)}]$, while sequences of inputs are denoted with subscripts, e.g. $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots$. To construct a predictor, an experiment is performed by evaluating the function for a given experimental *design* (a sequence of inputs), $\mathcal{X} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$, creating a vector of observations $\boldsymbol{y} = [y(\boldsymbol{x}_1), \ldots, y(\boldsymbol{x}_N)]^\mathsf{T}$. The value of N is known as the *sample size* of the experimental design. A smaller sample size represents a less expensive design.

After observing these input/output pairs, a predictor is then built by finding a representative function based on the observations. The often adopted approach treats the unknown function as the realization of a stochastic process. Specifically, $y(\cdot)$ is a realization of a random function $Y(\cdot)$ which has the density of a Gaussian process. The capitalization of the output $Y(\boldsymbol{x})$ indicates a random output while the lower case $y(\boldsymbol{x})$ indicates the observed realization. We denote the Gaussian process assumption

on a random function $Y(\cdot)$ as

$$Y(\cdot) \sim GP(\mu(\cdot), C(\cdot, \cdot)),$$

where $\mu(\cdot)$ is the mean function and $C(\cdot, \cdot)$ is a function such that $C(\boldsymbol{x}_1, \boldsymbol{x}_2) = \text{cov}(Y(\boldsymbol{x}_1), Y(\boldsymbol{x}_2))$ for all possible $\boldsymbol{x}_1, \boldsymbol{x}_2 \in X$.

Our goal is to predict an unobserved output at an untried input $\boldsymbol{x}_0$ given $\boldsymbol{Y} := [Y(\boldsymbol{x}_1), \ldots, Y(\boldsymbol{x}_N)]^\mathsf{T} = \boldsymbol{y}$. The commonly used predictor of $y(\boldsymbol{x}_0)$ is

$$\hat{y}\left(\boldsymbol{x}_0\right) = \mu(\boldsymbol{x}_0) + \boldsymbol{\sigma}^\mathsf{T}(\boldsymbol{x}_0)\boldsymbol{w}, \tag{1}$$

where $\boldsymbol{w} \in \mathbb{R}^N$ is a vector of weights and $\boldsymbol{\sigma}^\mathsf{T}(\boldsymbol{x}_0) = [C(\boldsymbol{x}_0, \boldsymbol{x}_1), \ldots, C(\boldsymbol{x}_0, \boldsymbol{x}_N)]$. In general, $\boldsymbol{w}$ is given by the following relation

$$\boldsymbol{w} = \boldsymbol{\Sigma}^{-1}\left(\boldsymbol{y} - \boldsymbol{\mu}\right),$$

where $\boldsymbol{\mu} = [\mu(\boldsymbol{x}_1), \ldots, \mu(\boldsymbol{x}_N)]^\mathsf{T}$ and $\boldsymbol{\Sigma}$ is the $N \times N$ covariance matrix where the element in the ith row and jth column is $C(\boldsymbol{x}_i, \boldsymbol{x}_j)$. This predictor, $\hat{y}\left(\boldsymbol{x}_0\right)$, is commonly used because it is both the mean and median of the predictive distribution of $Y(\boldsymbol{x}_0)$ given $\boldsymbol{Y} = \boldsymbol{y}$. This property implies $\hat{y}\left(\boldsymbol{x}_0\right)$ is optimal among the class of both linear and nonlinear predictors of $y(\boldsymbol{x}_0)$ with respect to the quadratic and absolute loss functions.

1.3 Space-filling and lattice experimental designs

A typical assumption on the covariance structure is a separable covariance, defined as $C(\boldsymbol{x}_1, \boldsymbol{x}_2) = \prod_{i=1}^{d} C_i(x_1^{(i)}, x_2^{(i)})$ for all $\boldsymbol{x}_1, \boldsymbol{x}_2 \in X$. The functions C_i are covariance functions defined when the input is one dimensional. The value of $C_i(x, x')$ is proportional to the covariance between two outputs corresponding to inputs where only the ith input differs from x to x'. The results in this section require this covariance structure to hold, but no other assumptions are needed for $\mu(\cdot)$ and $C(\cdot, \cdot)$.

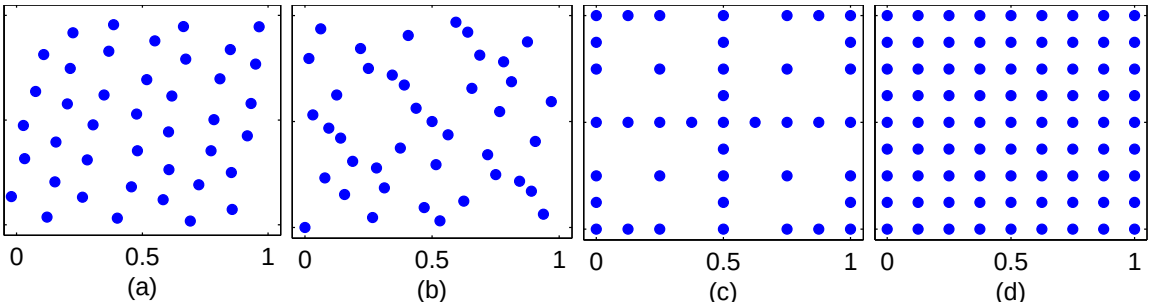

Figure 1: Examples of 2-dimensional designs: (a) A 41 point Latin-hypercube design. (b) The first 41 points in the Sobol sequence. (c) A 41 point sparse grid design. (d) An 81 point lattice design. Details of the construction of the sparse grid design in (c) are given in appendix Chapter 3.

The above approach, when applied in a direct manner, can become intractable because the inversion of the covariance matrix $\boldsymbol{\Sigma}$ is an expensive operation in terms of both memory and processing. Direct inversion can also induce numerical errors due to limitations of floating point mathematical computations [127, 52]. Previous research has focused on changing the matrix $\boldsymbol{\Sigma}$ to a matrix that is easier to invert, therefore making the computation of $\boldsymbol{w}$ faster [42, 24, 6]. We term this an *approximation* because this can degrade predictive performance, though sometimes only slightly.

This section will briefly discuss existing research on *space-filling* and *lattice* designs. The space-filling category includes the popular Latin hypercube designs. Lattice designs are a specific class of designs where each design is a *Cartesian product* of one dimensional designs. Visual examples are given in Figure 1 and they are contrasted with an example of a sparse grid design which will be explained in Chapter 3.

1.3.1 Space-filling designs: Efficient predictors but difficult computation

Current research has emphasized the design of points that are *space-filling* (see Figure 1 (a) and (b)). Designs of this type are often scattered, meaning they are not necessarily located on a lattice. The major focus has been on Latin hypercube

designs [79], seen in Figure 1 (a), and research has produced a swell of variations, e.g. [119, 92, 84, 135, 62]. These designs have been shown to perform well in many prediction scenarios and are often considered the standard method of designing computer experiments for deterministic functions.

However, space-filling designs experience significant difficulties when the input is high dimensional, i.e. $d > 3$. In these cases, one requires a large sample size N to develop an accurate predictor. This in turn makes the matrix $\boldsymbol{\Sigma}$ very large, meaning $\boldsymbol{w}$ is difficult to compute through inversion of $\boldsymbol{\Sigma}$. This has motivated the research into approximate predictors discussed in Chapter 2 that can be used with space-filling designs.

1.3.2 Lattice designs: Easy computation but inefficient predictors

One of the simplest forms of an experimental design is a *lattice* design, also known as a grid. This is defined as $\mathcal{X} = \mathcal{X}_1 \times \mathcal{X}_2 \times \ldots \times \mathcal{X}_d$ where each $\mathcal{X}_i$ is a set of one dimensional points we term a *component* design. For a set A and B, the Cartesian product, denoted $A \times B$, is defined as the set of all ordered pairs (a, b) where $a \in A$ and $b \in B$. If the number of elements in $\mathcal{X}_i$ is n_i, then the sample size of a lattice design is $\prod_{i=1}^{d} n_i$.

Let the covariance be as stated above, $C(\boldsymbol{x}_1, \boldsymbol{x}_2) = \prod_{i=1}^{d} C_i(x_1^{(i)}, x_2^{(i)})$. When a lattice design is used, the covariance matrix takes the form of a *Kronecker product* of matrices: $\otimes_{i=1}^{d} \boldsymbol{S}_i$, where $\boldsymbol{S}_i$ is a matrix composed of elements $C_i(x, x')$ for all $x, x' \in \mathcal{X}_i$. A useful property of Kronecker products can be derived using only the definition of matrix multiplication and the commutativity of scalar multiplication: if $\boldsymbol{A} = \boldsymbol{C} \otimes \boldsymbol{E}$ and $\boldsymbol{B} = \boldsymbol{D} \otimes \boldsymbol{F}$ then $\boldsymbol{AB} = \boldsymbol{CD} \otimes \boldsymbol{EF}$ (when matrices are appropriately sized). This immediately implies that if $\boldsymbol{C}$ and $\boldsymbol{E}$ are both invertible matrices, $\boldsymbol{A}^{-1} = \boldsymbol{C}^{-1} \otimes \boldsymbol{E}^{-1}$. Thus, if a lattice design is used,

$$\boldsymbol{w} = \left(\otimes_{i=1}^{d} \boldsymbol{S}_i^{-1}\right)(\boldsymbol{y} - \boldsymbol{\mu}),$$

which is an extremely fast algorithm because $\boldsymbol{S}_i$ are n_i sized matrices. Many authors have noted the power of using lattice designs for fast inference for these types of models [90, 12]. Say that we have a symmetric design where $\mathcal{X}_i = \mathcal{X}_j$ for all i and j. Computing $\boldsymbol{w}$ requires inversion of $N^{1/d} \times N^{1/d}$ sized matrices which are much smaller than the $N \times N$ sized matrix $\boldsymbol{\Sigma}$. Because inversion of an $N \times N$ size matrix requires $\mathcal{O}(N^3)$ arithmetic operations, inverting multiple small matrices versus one large one yields significant computational savings.

While lattice designs are extremely simple and result in fast-to-compute predictors, these are wholly impractical for use in high dimensions. First, lattices are grossly inefficient as experimental designs when the dimension is somewhat large ($d > 3$), which will be demonstrated in Chapter 3. Also, the sample size of a lattices designs, $\prod_{i=1}^{d} n_i$, is extremely inflexible regardless of the choice of n_i. At minimum $n_i = 2$, and then even for a reasonable number of dimensions the size of the design can become quite large. When $d = 15$ the smallest possible design size is over $30,000$.

1.4 Noise in computer experiments

Computer simulation is widely used to measure the performance of systems in the presence of stochastic behavior. Typically, the simulation has a collection of inputs which represent a variety of unknown or controllable aspects of the system. However, these simulations can be computationally expensive to run in fine-mesh or large-scale simulation environments. The investment in development of computer models can be lost if, for example, a large number of alternative inputs need to be investigated or the desired analysis requires repeated evaluations over long periods of time where computer clusters may be unavailable. For example, take the propagation of cracks in metals, where the stochastic nature of grain formation creates uncertainty in fracture growth rates [115]. The proliferation of increasingly complex numerical algorithms for fatigue analysis necessitates a limited sample size (see Chapter 4 for examples).

However, implementation scenarios involving online condition monitoring, e.g., [100], require a large number of evaluations with a limitless sample size.

This work proposes a method to *emulate* the stochastic simulation with a simple stochastic model. The emulator of a stochastic simulation provides two important constructions: (1) an explicit functional form of the distribution and (2) a fast sampling scheme. The emulator can then be integrated into analysis software (e.g. spreadsheet environments), which allows for timely results from investigations such as what-if scenarios and uncertainty quantification. An emulator is created by establishing a predictive distribution of the simulation output based on observations from an experiment. The predictive distribution is based on a stochastic model representing the simulation output, termed the *metamodel*. As has been shown in multiple disciplines, including geostatistics [77, 28] and analysis of deterministic computer code [105, 106], random field metamodels often offer superior representation of underlying continuous functions compared to polynomial metamodels [8]. The use of these random field metamodels is commonly referred to as *kriging*.

While previous attempts using random field metamodels have focused on the *mean* of the simulation output, an emulator for the simulation's *stochastic behavior* is often needed. [2] describes the case when the variance of the output significantly changes with changes in the inputs, an important concern in stochastic simulations. However, the use of the traditional random field metamodel as in [69, 2] and [93] is inadequate to provide predictive distributions due to a normality assumption on the stochastic behavior of the simulation output. The popular technique of model-based geostatistics [28] and similar methods [102, 55] addresses normality concerns when the output is in a parametric class (e.g. exponential). However, parametric assumptions often do not have the power to address the complex distributions that can result from simulations, e.g. [118] discusses bimodal cases. In our experience, the previously developed methods prove powerful when the respective assumptions

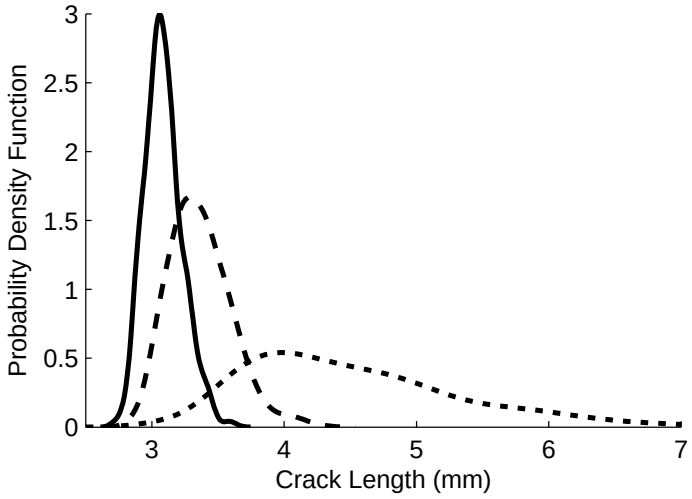

Figure 2: Empirical distributions of crack lengths after 2000 cycles with a stress ratio of 0 (solid line), .25 (long dashes) and .5 (short dashes).

hold, but there exist cases when a single parametric model for all inputs is not a reasonable assumption. For example, take the stochastic modification of the Forman equation, which is a general model for the growth rate of fractures based on a stress ratio (details are discussed in Chapter 4). The goal considered here is the prediction of the distribution of the crack size after 2000 cycles with an initial size of 2.54 mm. Figure 2 shows the distribution of the simulation output. When the stress ratio is nearly zero, an approximately Gaussian behavior results. As the stress ratio increases, this structure breaks down, indicating that previously developed techniques cannot be used.

1.4.1 Implications in steel manufacturing

There are theoretical connections between the the methods to construct inference on noisy computer models and estimation of general functions. One particular example of this connection is in the estimation of the intensity function of point processes. These are encountered, for example, when one observes defects on products. This will be discussed in Chapter 5.

Automated detection of defects on products has become a reality thanks to developments in video recording, computer processing and digital storage technology. When defects are automatically detected, the spatial locations of the defects are recorded for each product. One of the most important features of any production process is not necessarily the presence of faults (which are unavoidable due to the stochastic nature of a production line) but instead the cause of these defects. In this case, this is determined by a *defect pattern*, which is defined as the relative occurrence rate of defects in any section of the product. If a region has an abnormally large amount of defects, we consider the pattern important as it allows a user to isolate the cause to certain subsystems. If the chance of defects throughout the entire product is the same, we say there is *no* defect pattern. More generally, we can say that if a pattern meets our expectations it follows a *null* pattern. For example, a product that is designed with both finished and unfinished surfaces would have significantly more defects in the unfinished regions.

While there has been statistical study of cases when a production system relays information in the form of a scalar or a vector, e.g. [97], the study of spatial patterns of defects remains mostly unexplored. This work considers two goals:

- to provide a certificate of whether the pattern differs from a null pattern and
- to give a visual indication of where defects are more likely to occur.

We will do this by *estimating the underlying functional form of the pattern of defects*. Explicitly, we will model each individual product as an independent realization of an inhomogeneous Poisson process with an unknown intensity function. This intensity function represents the probability of defects in any subregion of the product. Given the application, it is critical that the estimated pattern coverages *quickly* to the generating the observed data. The major paradigm, see [98] and [29], results in estimates that do poorly in this scenario for reasons that will be discussed. In short,

since that framework places an emphasis on shrinking the lengthscale of the basis functions as more observations are collected, the ability of the method to detect a variety of pattern shapes is limited. In this thesis, an alternative estimate is proposed to solve this problem.

Chapter II

CALIBRATING FUNCTIONAL PARAMETERS IN THE ION CHANNEL MODELS OF CARDIAC CELLS

This work presents an alternative to the empirical methods that have been employed to study ion channels in cardiac cells. The data we used for our study were collected and first analyzed by [36]. In that original empirical analysis, [36] first used least squares to fit exponential models to recordings of membrane current over time. Then the fitted exponential constant was used to make inferences on the system. This method is similar to techniques to calibrate and justify computational models [21, 116, 83]. We can broadly call this a projection method, which is not necessarily an incorrect approach provided the projection preserves the features present in the data [56]. But an examination of our longitudinal responses shows they are extremely dissimilar from exponential shapes. This projection method may thus have a tempered capability to draw meaningful conclusions. In this work, we consign inferential tools that are designed to meet the exigent demand for more rigorous methods.

This chapter will discuss background in Section 2.1 followed by statistically motivated modifications and observations in Section 5.2. Section 2.3 and Section 2.4 will precisely define our model with functional parameters and the Markov chain Monte Carlo algorithm used to conduct inference. Lastly, Section 2.5 will detail discoveries made during our case study. Some of these results contradict portions of knowledge observed in the existing literature on cardiac cell ion channels.

2.1 Background

This work studies the electrical activity of myocytes (muscle fiber cells) located on a heart's ventricle walls. When activated, ion channels located on the boundary of the cell allow for the influx and efflux of ions through the channels generating electrical currents. The time course of this activity is referred to as *action potential* of the ventricular myocytes. A steady recurrence of the de- and repolarization of these myocytes aggregates to the rhythmic, steady behavior of the heart. Slight changes in channel kinetics can alter the action potential waveform, and potentially cardiac excitation and conduction. For additional background on the general conceptual basis for cardiac cell ion channels see [57] (or [48] for a more condensed summary).

[58] formulated the first cell action potential model using a set of nonlinear and ordinary differential equations. This original model was not designed for cardiac cells. Subsequent models of cardiac cells were developed to simulate the action potential through the ion channels, see the review of [89]. These models of trans-membrane ionic currents consider ion channel kinetics as well as ionic concentrations. Different species and regions of the heart yield vastly different electrophysiological behaviors. We chose to borrow a model of ion channels from research with similar conditions to our case study on mouse cardiac cells. The impactful chapter by [21] was used by [15] to mimic the action potential of ventricular myocytes in adult mice, thus we adopt the model of [21].

The membrane potential, denoted by the function of time $v(\cdot)$, is the potential difference between the intracellular and extracellular sides of the cell. The current $i(\cdot)$ is composed of several trans-membrane currents grouped by the ion being transported, either sodium, potassium, or calcium. The model of [21] is based on the principle of the opening and closing of ion channels, known as gating. Research has shown that closed states can be further broken down into inactive and simply closed states. While in both states no current occurs, the inactive states are much less likely to move into

an open state. It is possible for a channel to undergo movement from either an open or a closed state to an inactivated state.

Our data consists of recorded current through only the sodium channels in a cell membrane. The current flowing through sodium channels can be represented by

$$\mathrm{i}(t) = G\,\mathrm{o}(t)(\mathrm{v}(t) - \mathrm{e}(t)) + G_b(\mathrm{v}(t) - \mathrm{e}(t)),$$

where G is a conductance parameter which determines the dynamic ion permeation of the cell, G_b is another the conductance parameter which determines background ion permeation of the cell, $\mathrm{o}(t)$ is the proportion of open channels at time t, and $\mathrm{e}(t)$ is channel the reversal potential at time t. A key point here, which will be directly addressed soon, is that the *dynamics* of o depend on the membrane potential, v. The form of $\mathrm{e}(t)$ is given by $\mathrm{e}(t) = RT/F \cdot \log(Na_0/Na_i)$ where R, T, and F are physical constants and Na_0 and Na_i are the extracellular and intracellular sodium concentrations. The intracellular sodium concentrations will change with the flow of ions.

Our data are observations from a voltage clamp experiment. The voltage-clamp method is a laboratory technique used to study currents passing through the cell membrane. [10, 60]. In the voltage-clamp experiment, we first place electrodes in the intracellular and extracellular space. Starting at time zero the transmembrane voltage is held at a predefined level and ionic currents flow through the membrane. Both electrodes are connected to an amplifier which measures the membrane voltage. At the same time, a signal generator can input an external voltage (a holding potential) to the cell. In the patch clamp experiment [87], the operator uses a glass micropipette with an open tip diameter of about one micrometer as an electrode. The micropipette is filled with a solution that is paired with the ionic composition of the bath solution. A chloride silver wire is placed in the bath solution that conducts electric currents to amplifier. Whole-cell recording marks the currents through all ion channels over the membrane of the whole cell. Our data comes from whole-cell experiments that give

better electrical access to the interior of the cell because of a large opening at the tip of the pipette. By carefully selecting the intracellular and extracellular solutions in the patch clamp experiment we are able to isolate the sodium ion channels. More details on the collection of the data can be seen in [36].

2.2 Modified computational model

Our model for $\mathrm{o}(t)$ is described in detail in section 2.7.1 and contains six unknown parameters $\{\theta_i\}_{i=1}^6$. This model was closely borrowed from [21], and first we emphasize a subtlety in this model:

> The parameters of the channel gating dynamics, $\{\theta_i\}_{i=1}^6$, depend on the membrane potential v. We thus write $\boldsymbol{\theta}(v)$ to mean $\{\theta_i\}_{i=1}^6$ given a fixed voltage v.

Our model of $\mathrm{o}(t)$ can be described as follows. Consider a single ion channel that behaves according to the continuous time Markov model in figure 3. We assume N channels independently behave according to this stochastic model, then as $N \to \infty$ the proportion of cells in each state converges to the proposed model (also known as the fluid limit [72]). The model we employ for the proportion of cells in each state differs slightly from [21]. We have removed an inactivation state that occurs only on extremely long lengthscales and is unlikely to be present in our observations. We have also grouped parameters with similar values to the same value. For example, the connection from C_1 to C_2 is slightly different from C_2 to C_3 in their model but not in ours. These alterations produced little change in simulated outputs.

For our analysis we respecify our model by leveraging the specific conditions of the voltage clamp experiment. From time zero onwards, the value of $\mathrm{v}(t)$ is held at a constant. The experiment design thus gives us a useful formulation:

> The parameters of the channel gating dynamics, $\{\theta_i\}_{i=1}^6$, are constant during voltage clamp experiments.

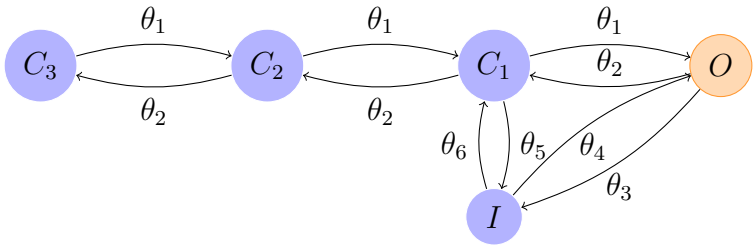

Figure 3: Diagram of the Markov model for the sodium channel. The states C_1, C_2, C_3 represent closed states, O represents the open state, and I represents the inactive state. The variables θ represent dynamic transition rates which depend on membrane potential v.

The above fact simplifies our analysis greatly. Plugging this relation into our model, we get the response from voltage clamp experiments with held membrane potential v can be modeled by

$$\mathrm{i}(t) = G\,\mathrm{o}(t; \boldsymbol{\theta}(v))(v - \mathrm{e}(t)) + G_b(v - \mathrm{e}(t)),$$

where o now depends on a static parameter value $\boldsymbol{\theta}(v)$.

The slow dynamics of intracellular sodium accumulation imply that intracellular sodium occurs during the voltage clamp protocol, possibly due to the natural regularization [19]. Therefore we can restate an approximate model as

$$\mathrm{i}^*(t, v; \boldsymbol{\theta}(v), G, G_b, E) = G\,\mathrm{o}(t; \boldsymbol{\theta}(v))(v - E) + G_b(v - E) \tag{2}$$

where E is a constant value across all clamp experiments. We denote our new function with i^* to emphasize the modifications we have introduced. The motivation for this change is also practical as well as physical. The inference from the data on the parameters of interest $\{\boldsymbol{\theta}(\cdot), G, G_b, E\}$ is conducted using a Gibbs sampler. The function i^* as a function of t depends only on o. The function o is the solution to a set of ordinary linear differential equations and therefore has an explicit form available in a matter of hundredths of a second (Section 2.7.1). We must also acknowledge the exitance of inexactness when using this model. The bias function described in Section 2.3.2 allows us to account for these discrepancies in our inference.

2.3 Statistical modeling

The voltage clamp experiments yields observations denoted $y(t_k, v_j)$ for time points $t_1, \ldots, t_N$ and clamped membrane potentials $v_1, \ldots, v_M$. We will conjecture about the cell properties by taking on the Bayesian viewpoint. From this perspective we can leverage the rich literature history and can account for limited data from physical cells. The major statistical novelty of this work is the ability to infer on a functional parameter $\boldsymbol{\theta}(\cdot)$. Section 2.3.1 establishes a nonparametric statistical model for the functional parameter $\boldsymbol{\theta}(\cdot)$. Given that the data $y(t_k, v_j)$ will deviate from the function $\mathrm{i}^*(t_k, v_j)$, we build a full stochastic model in section 2.3.2.

2.3.1 Functional parameters

Parameters calibrated in previous studies in the literature are scalars (see references in the introduction). So our modeling of parameters must depart here from previous statistical literature on calibration. The relationship between $\boldsymbol{\theta}$ and the membrane potential is a functional dependence as we can observe a response for any clamped membrane potential. [21] assumed that $\boldsymbol{\theta}$ follows convoluted empirical formulae, but we found this too restrictive.

We instead invoke the Bayesian paradigm and place a prior on this function, specifically a Gaussian process prior. The Gaussian process prior is a distribution on a continuous function such that any collection of function evaluations follow a multivariate normal distribution. This prior is used because we anticipate that similar membrane potentials will result in similar ion channel behavior. In other words, the parameters should be continuous about membrane potential v. The use of a Gaussian process model is inspired by the computer experiments literature such as [105] and [26]. We denote this Gaussian process prior as

$$\log \theta_i(\cdot) \overset{\text{indep.}}{\sim} \mathrm{GP}(\mu_i(\cdot), \sigma_i^2 R_i(\cdot, \cdot)),$$

where indep. implies that the prior distribution of θ_i is independent of θ_k if $k \neq i$.

The log transform is used because the parameters ought to be positive. In our case study, we use a Matérn correlation function for R_i with smoothness parameter 2.5 and a lengthscale parameter of 10 mV (see [53]), i.e.

$$R_i(v, v') = (1 + \sqrt{5}\Delta_v + 5\Delta_v^2/3)\exp(-\sqrt{5}\Delta_v),$$

where $\Delta_v = |v - v'|/10$. The lengthscale parameter is determined based on the anticipated change in parameter as voltage is changed. This implies that we anticipate at least two orders of differentiability and $\theta(\cdot)$ should not change drastically over small alterations in voltage. In our example, we take $\mu_i(\cdot)$ to be a linear model with parameters $\boldsymbol{\beta}_i$, a column vector. Specifically, we choose $\mu_i(v) = [1, v - 35]\boldsymbol{\beta}_i$.

2.3.2 Observations

Owing to the ubiquitous [67] Bayesian formulation of the calibration problem, we model the difference between our observations y and i^* with two elements. First we have a stochastic random error ε and secondly we have a discrepancy function b that represents possible differences between i^* and reality. Some potential sources of discrepancy are discussed in section 5.2 but it is by no means an exhaustive list. For example, while conducting the experiment there was a small deviation in the clamped voltage due to controller dynamics [36]. The discrepancy is not known exactly, but if the model is accurate the discrepancy will be small and the converse is true. We thus formulate our observed current for a given time point and clamped membrane potential as

$$y(t_k, v_j) = \mathrm{i}^*(t_k, v_j; \boldsymbol{\theta}(v_j), G, G_b, E) + b(t_k, v_j) + \varepsilon_j(t_k).$$

Our prior on the function b is represented by a Gaussian process with zero mean and a covariance structure defined over both time and voltage. This implies that our prior is that b is a continuous function over both time and voltage. So a small change in either the time or the clamped membrane potential should elicit a similarly small

change in both the model i* and observation y and therefore b. The perturbations represented by ε_j are generated by small environmental influences which will differ upon each subsequent measurement and are continuous in time but not over the repeated subject measurements. We thus consider $\varepsilon_j(\cdot)$ to follow a Gaussian process over time and each ε_j is independent of ε_k if $k \neq j$.

2.3.3 Prior on parameters

Some parameters can be fixed because their value is well studied, determined by physical constants or their effect on the likelihood is minor (implying unidentifiable). For some parameters, we have no *a-priori* knowledge of their value, but their value can be gleaned from the data. In these cases, we can place a prior distribution on the parameters themselves to conjecture about their value through their posterior distribution. In our model, we can place priors on the parameters $\boldsymbol{\beta}_i$, σ_i^2, G, G_b, E, σ_ε^2 and σ_b^2. Our chosen priors with motivation are given in section 2.7.2.

2.4 Analysis

Having established a modeling framework, we now need computational methods for conducting inference. Unfortunately, our parameter space is infinite due to $\boldsymbol{\theta}(\cdot)$ being a functional parameter and we require some developments for analysis. This section will provide those developments in the form of a Gibbs sampler that takes advantage of the structure of our model and data.

Let ϕ be the agglomeration of all parameters besides $\boldsymbol{\theta}(\cdot)$. In the interest of compact notation we denote $\{y(t_k, v_j)\}_{k=1,\ldots,N;j=1,\ldots,M}$ as simply the term "data". Let $\pi(\theta_i(v)|\text{data}, \phi)$ be the posterior density of $\theta_i(v)$ given the data and the other parameters ϕ and v is any membrane potential. First, we have the following simple result because we chose independent Gaussian process priors, $\theta_i(v)$ given $\{\theta_i(v_j)\}_{j=1}^M$ is independent of $\{\theta_k(v_j)\}_{j=1}^M$ if $k \neq i$. Since the data only depends on $\{\boldsymbol{\theta}(v_j)\}_{j=1}^M$ and

ϕ, we can condition as follows

$$\begin{aligned}\pi(\theta_i(v)|\text{data},\phi) &= \int \pi(\theta_i(v)|\{\boldsymbol{\theta}(v_j)\}_{j=1}^M,\text{data},\phi)\pi(\{\boldsymbol{\theta}(v_k)\}_{k=1}^M|\text{data},\phi)\mathrm{d}\{\boldsymbol{\theta}(v_k)\}_{k=1}^M,\\ &= \int \pi(\theta_i(v)|\{\theta_i(v_j)\}_{j=1}^M)\pi(\{\boldsymbol{\theta}(v_k)\}_{k=1}^M|\text{data},\phi)\mathrm{d}\{\boldsymbol{\theta}(v_k)\}_{k=1}^M, \qquad (3)\end{aligned}$$

where π generally represents a density and the vertical lines represent conditioning. The two density terms on the right hand side are the density of $\theta_i(v)$ given $\{\theta_i(v_j)\}_{j=1}^M$ and the posterior distribution of $\{\boldsymbol{\theta}(v_k)\}_{k=1}^M$. Now we employ standard Gaussian process relations. Let $\tilde{\boldsymbol{\theta}}_i = [\log\theta_i(v_1),\ldots,\log\theta_i(v_M)]^\mathsf{T}$, $\tilde{\boldsymbol{\mu}}_i = [\mu_i(v_1),\ldots,\mu_i(v_M)]^\mathsf{T}$, $\boldsymbol{r}_i(v) = [R_i(v,v_1),\ldots,R_i(v,v_M)]^\mathsf{T}$ and $\boldsymbol{R}_i$ be the correlation matrix where the jth, kth element is $R_i(v_j,v_k)$. We know that $\pi(\log\theta_i(v)|\{\theta_i(v_j)\}_{j=1}^M)$ is the same as the density of

$$\mathcal{N}\left(\mu_i(v) + \boldsymbol{r}_i^\mathsf{T}(v)\boldsymbol{R}_i^{-1}\left(\tilde{\boldsymbol{\theta}}_i - \tilde{\boldsymbol{\mu}}_i\right), \sigma_i^2\left(1 - \boldsymbol{r}_i^\mathsf{T}(v)\boldsymbol{R}_i^{-1}\boldsymbol{r}_i(v)\right)\right),$$

where $\mathcal{N}(\mu,\sigma^2)$ stands for a standard normal distribution with mean μ and variance σ^2. Since a normal distribution is simple to sample from, to sample from any value of $\theta_i(v)$ we only need to sample from the posterior distribution of the parameters $\{\boldsymbol{\theta}(v_j)\}_{j=1}^M$.

To infer on the parameters $\{\boldsymbol{\theta}(v_j)\}_{j=1}^M$ given data we leverage the form of our posterior and propose the following Gibbs sampler if we are given ϕ and the data:

Cyclic sampler Suppose we are given initial values of $\{\boldsymbol{\theta}(v_k)^{(n-1)}\}_{k=1}^M$. For $j = 1,\ldots,N$ we draw $\boldsymbol{\theta}(v_j)^{(n)}$ given the data, ϕ, $\{\boldsymbol{\theta}(v_k)^{(n)}\}_{k<j}$, and$\{\boldsymbol{\theta}^{(n-1)}(v_k)\}_{k>j}$ via a Metropolis Hastings step or several Metropolis Hastings steps. The acceptance probability of a step is given in equation (5).

This algorithm cycles though the sampling of the parameter $\boldsymbol{\theta}$ at v_j for all j. The benefit of using the above sampler is instead of attempting to estimate a single $6M$ sized parameter $\{\boldsymbol{\theta}(v_j)\}_{j=1}^M$ we must only tackle several dimension 6 parameters $\boldsymbol{\theta}(v_j)$.

The efficiency gain comes from the fact that the posterior distribution of $\boldsymbol{\theta}(v_j)$ depends heavily on the likelihood of the observations $\{y(t_k, v_j)\}_{k=1,\ldots,N}$ which has strong dependence on $\boldsymbol{\theta}(v_j)$ through i* and only weak dependence on $\{\boldsymbol{\theta}(v_k)\}_{k\neq j}$ through the deviation term b and the prior on the functional parameters.

We use this cyclic sampler in concert with a typical sampler to arrive at our conclusions. The exact algorithm that we use in the next section is outlined below. For our analysis, we have fixed G, E and $\theta_4, \theta_5, \theta_6$, therefore there is no need to sample those values. Thus, at any stage in our sampler, the next parameter values in our Markov chain are drawn as follows:

1. Starting at the last sample of $\{\boldsymbol{\theta}(v_k)\}_{k=1}^M$, use the cyclic sampler to get samples of $\{\boldsymbol{\theta}(v_k)\}_{k=1}^M$ given the data and the latest sample of all other parameters.

2. For each $i = 1, \ldots, 3$, use the distribution defined at the end of Section 2.7.3 to draw σ_i^2 directly from the conditional posterior of σ_i^2 given the data and the latest sample of all other parameters.

3. For each $i = 1, \ldots, 3$, use the distribution defined at the end of Section 2.7.3 to draw $\boldsymbol{\beta}_i$ directly from the conditional posterior of $\boldsymbol{\beta}_i$ given the data and the latest sample of all other parameters.

4. Draw σ_ε^2 and σ_b^2 using Metropolis Hastings steps on conditional posterior of σ_ε^2 and σ_b^2 given the data and the latest sample of all other parameters.

5. Draw G_b using Metropolis Hastings steps on conditional posterior of G_b given the data and the latest sample of all other parameters.

After enough iterations of this sampler, we can establish what should be a nearly independent samples using standard Metropolis Hastings arguments [45]. In the following section, our analysis drew every 10th sample to collect 400 samples and established an average effective sample size of 128 for all parameters $\theta_i(v_j)$ [66].

Lastly, we relegate the discussion of the prediction of a new response, $y(t, v)$, to the section, see equation (6).

2.5 Case study: The effect of aberrant sialylation

2.5.1 Experimental data

This section leverages our statistical developments to investigate two groups of physical cells denoted "wild-type" and "ST3Gal4-deficient". The ST3Gal4-deficient subjects are lacking the enzyme β-galactoside α-2,3-sialyltransferase 4 (ST3Gal4) which is 1 of 20 sialyltransferase that increase sialic acids in galactose residues [83, 37, 34]. The ST3Gal4-deficient strain has been used to investigate the general problem of aberrant glycosylation in cardiovascular function. The wild-type subjects were generated with proper production of the ST3Gal4 enzyme.

In this work we use a total of seven wild-type and nine ST3Gal4-deficient cells selected from [36] that were deemed successful experiments based on visual inspection. Each cell was studied via clamped voltage excitation with clamped membrane potentials ranging from -70 mV to 0 mV in 5 mV increments. We leave out the response with a clamped membrane potential of -40 mV for a posterior check seen in Figure 2.5.2. The data collection was documented and first analyzed in [36]. The response current was saved at 150 non-uniformly spaced samples in time to account for the greater deal of variability in the run-up. Following the analysis of [36], the response y was normalized to account for cell size differences by dividing by a separately measured capacitance value. This allows us to set fix our G value as outlined in Section 2.7.2. It also implies that our y values and i^* values will be reported in units of $\mu A/\mu C$.

Efforts were made to insure that the data is properly aligned but we acknowledge this is inexact. Therefore, we add an additional offset parameter that indicates the start time of the stepped membrane potential for each voltage step in the experiment.

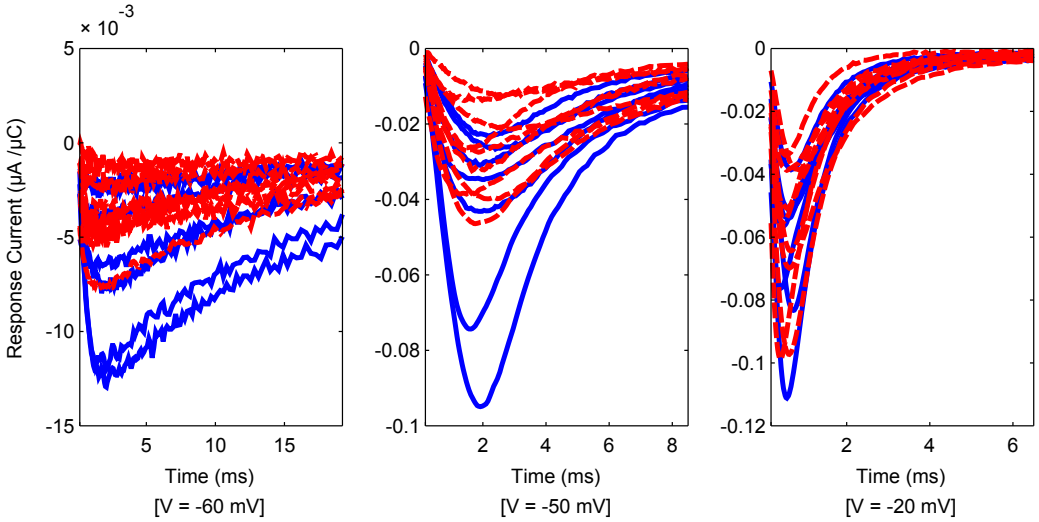

Figure 4: Collected responses for wild-type (solid) and ST3Gal4-deficient (dashed) during three separate clamped voltage experiments.

The response i* given this new offset parameter is a time shifted version of the original i*. This value has a prior distribution of normal with a mean of 0ms (since we attempted calibration) and a variance of .1ms. The sampling of this value takes place in an added fourth step in our Gibbs sampler described in Section 2.4.

Figure 4 shows the responses for three different membrane potentials. There is significant subject to subject difference. To account for this observation we will use our Bayesian method to sample from the posteriors generated from individual cell data (or equivalently, our prior assumes each subject is independent of others).

The ST3Gal4-deficient cells appear to behave separately from the wild-type cells especially in low membrane potentials. Our goal here is to reach some conclusions about the possible mechanism behind this effect. The prevailing paradigm for this type of study involves two steps. First, one conjectures a model for differences between the cells. Second, through a series of empirical comparisons one shows that the conjectured model is a superior representation of the modified cells. For example, [21] study the effect of a wholly different mutation on action potential and give biophysical justifications for a new model for the ΔKPQ mutated cells. Parameters of their

Markov model are fixed in both wild-type and mutated cells. This allows claims such as "[m]utant channels activate more quickly". The justifications are based on the parameters of a fitted exponential models that may not represent the data. Here we make four observations that distinguish our analysis from the previous literature: (i) the longitudinal responses do not follow any single exponential model but instead follow a computational model, (ii) the parameters of the computational model are uncertain, (iii) there is possibly a gap between our model and reality and (iv) the specific behavior of a cell can radically differ between subjects and is propagated through parameters of the computational model. The Bayesian framework we propose incorporates all of these observations.

2.5.2 Results and conclusions

The success of our analysis hinges on the adequacy of our model described in sections 5.2 and 2.3. As an initial model check, Figure 2.5.2 shows a predictive extrapolation using our proposed approach and the standard least squares calibration using the parameterized form of $\theta(\cdot)$. We predict the response to a clamped membrane potential of -5mV using the observations from clamped membrane potentials ranging from -70mV to 0mV. Our predictive accuracy and uncertainty quantification is significantly improved over the more traditional statistical approach.

Now we check whether the anticipated response, i^*, is sufficient to describe the observed behavior. If i^* is extremely different from y, the resultant posterior distributions of parameters such as $\boldsymbol{\theta}(\cdot)$ will be of little significance (and likely indistinguishable from the prior distribution). We will investigate the *deviation function* defined by $y - \mathrm{i}^*$ to detect if there is a problem with i^*. For a given t_k and v_j, the deviation function has a posterior density of equivalent to $y - \mathrm{i}^*$ when parameters are drawn from the posterior. The upper half of the plot in Figure 5 shows a comparison of posterior distributions of deviation functions using data from two different cells. In the

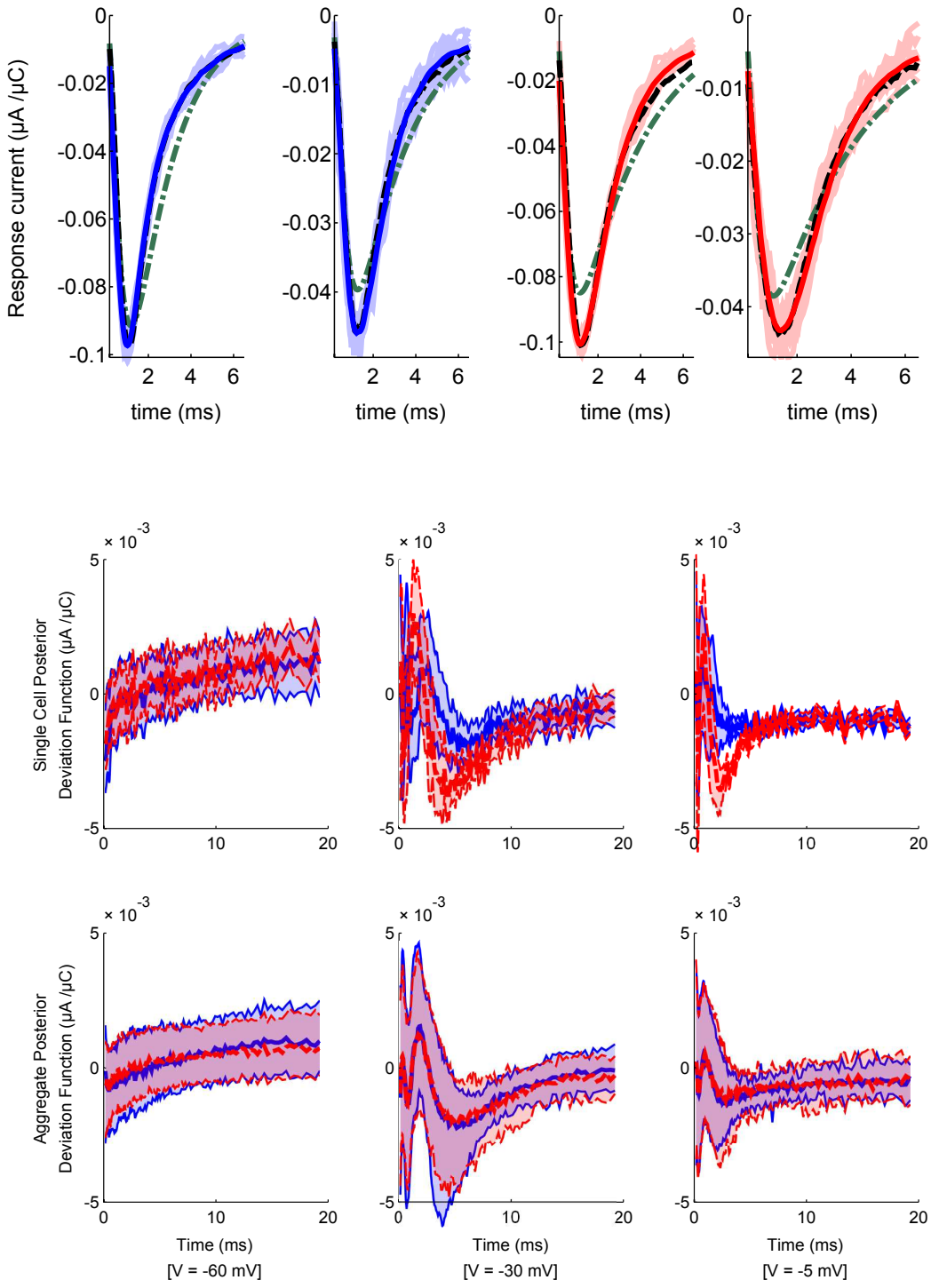

Figure 5: Pointwise medians and 90 % credible intervals of the deviation function's posterior distribution for three different clamped membrane potentials. The top three plots correspond to two individual cell posteriors and the bottom plots display the information for aggregated posteriors. The solid lines are from wild-type cells and the dashed lines are from ST3Gal4-deficient cells.

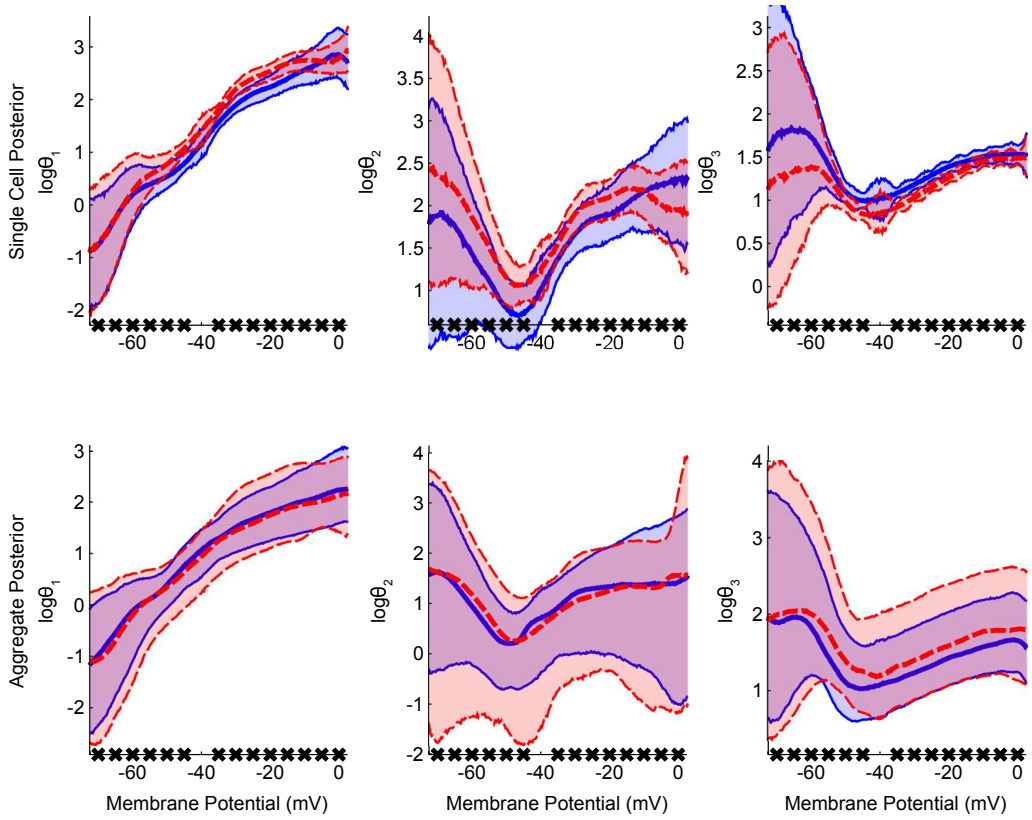

Figure 6: Pointwise medians and 5 %, 95 % quantiles of the posterior distribution of $\theta_1(\cdot)$, $\theta_2(\cdot)$ and $\theta_3(\cdot)$. The top three plots correspond to two individual cell posteriors and the bottom plots display the information for aggregated posteriors. The solid lines are from wild-type cell(s) and the dashed lines are from ST3Gal4-deficient cell(s) The 'x' tick marks represent points for which we had observations from voltage clamp experiments.

lower half of Figure 5, we display the *aggregate posterior* which is constructed by randomly choosing a cell from each group and drawing from the posterior corresponding to that individual cell's data. The magnitude of the posterior deviation function is reasonable considering our prior variance on b and ε. The deviation function appears continuous, but not necessarily smooth, which is also consistent with our prior model. Figure 5 also supports the idea that the described model from [21] may be sufficient for describing both wild-type and ST3Gal4-deficient cells.

We have established that our stochastic model can be used for both groups of cells. We can now move to addressing the question of which element in i^* is the root

of the observed differences present in Figure 4. In Figure 6, we have displayed our posterior distribution of θ_1, θ_2 and θ_3 to show representative single cell and aggregated posteriors. The values of $\theta_1(v), \theta_2(v)$ and $\theta_3(v)$ represent the rates at which channels move between states in our model at a membrane potential v, also called transition rates. The values of θ_1 and θ_2 represent the rates at which states open and close from closed and open states, respectively. The value of θ_3 represents the rate at which states inactivates from open states. Because we have defined our parameters as functions, we can easily visualize their posteriors.

We can now conjecture about the differences between the two groups of cells in terms of our posited model. To do so, we define $\delta_i(v)$ as the ratio of $\theta_i(v)$ for all ST3Gal4-deficient cells to the average of $\theta_i(v)$ for all wild-type cells. The posterior distribution of this value is seen in Figure 7. A concentration of the posterior around a value of δ_i much bigger than 1 implies that ST3-deficient cells have a higher value of θ_i and a value less than 1 implies the opposite.

The first note of interest is that θ_1, the transition rate from closed to open, is decreased in ST3Gal4-deficient cells. Additionally, ST3Gal4-deficient cells may have an increased transition rate from open to closed as θ_2 is increased at mid-level membrane potentials, but this effect is slightly reversed at high membrane potentials. Based on the 90% intervals being very wide, the effect we see is not strong. Inactivation from open channels appears to be faster in ST3Gal4-deficient cells. This observation contradicts [36] who used the same data to note that ST3Gal4-deficient cells "inactivated more slowly". The difference in conclusions may be due to the differences in our respective analyses. In [36], "the decaying portion of each current trace was fit with a single exponential function". Their method might induce problems because the responses do not follow exponential shapes (see Figure 4). Another reason for the different conclusions might be because our data is a subset of their data. We do not include four wild-type cells and four ST3Gal4-deficient cells because their responses

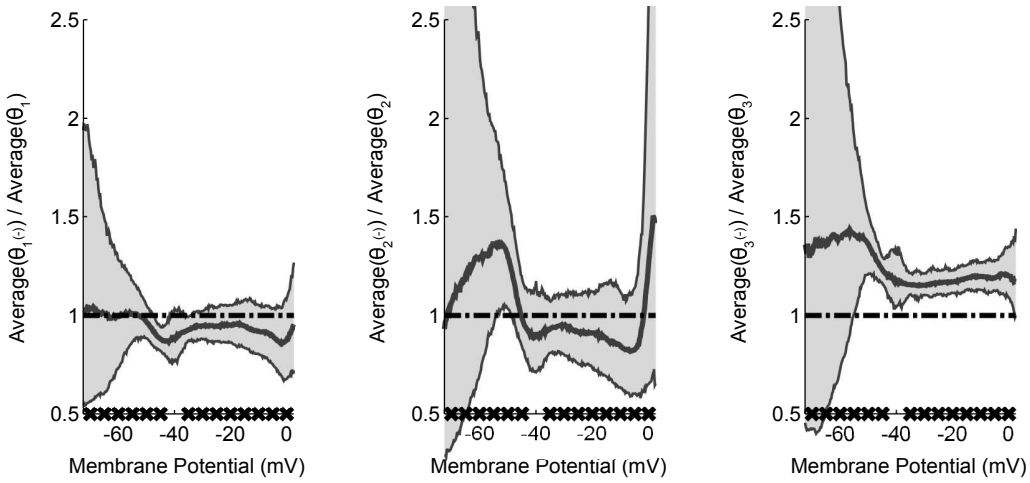

Figure 7: Pointwise medians and 90 % credible interval for $\delta_1(\cdot)$, $\delta_2(\cdot)$ and $\delta_3(\cdot)$. The 'x' tick marks represent points for which we had observations from voltage clamp experiments.

indicated an experiment failure or were not recorded properly at one or more of the membrane potentials.

2.6 Concluding remarks

This chapter has developed a method to infer on a functional parameter using Gaussian process prior distributions. We were able to a investigate model of the sodium ion channels in cardiac cells. Going forward, it will be interesting to investigate several other electrophysiological challenges. Action potential simulation requires the full functional form of $\theta(\cdot)$. It is not clear how to leverage our functional parameter's posterior distribution into a computational model where the voltage can change. A simple implementation would use our mean as a plug-in estimate. However, the uncertainty quantification will likely require some type of Monte Carlo method. Another complexity of single-cell experiments is that similar single-cell parameters might produce very different aggregate responses (e.g. [49]). Thus methods to translate individual cell results to larger systems would likely be very impactful.

Using a Bayesian approach for electrophysiological systems we can also investigate

the uncertainty in our estimated parameters. This could be critical for electrophysiological systems as there has been observations that multiple parameter combinations can produce similar outputs (see [1] for an example involving neurons). A wide posterior indicates that creating a good parameter estimate would prove difficult based on the existing data. A narrower posterior could assure a researcher that we have enough data to conjecture about a parameter. Thus, this framework has the potential to make distinctions of significant differences in parameters easier for a researcher. As a comparison, [17] uses thousands of parameter combinations as a search mechanism to find appropriate model parameters. The Bayesian approach that we outline here could serve as a substitute for that method. This analogy becomes more apparent if our sampling scheme was replaced with Approximate Bayesian Computation [25]. In this work, we have addressed the inter-subject variability by isolating specific parameters for each cell in our study and sampling from their individual posterior. The review of [107] indicates that variability expressed in the observations could be explained through parameters for electrophysiological cell models.

In addition to the future study of the electrical activity of cardiac cells, work is needed to infer on functional parameters in general cases. The conditional distributions used for equation (3) rely on leveraging the voltage clamp experiments, but often experiments where the environmental condition is held constant are impossible. In these cases, more flexible tools are required. One idea for conducting inference is to use only function values corresponding to inputs on a fine mesh, thus making the problem finite dimensional. Research is needed to investigate sampling mechanisms that coordinate well with this framework [22]. Moreover, even in the case stated in the chapter, there is doubt as to whether the cyclic sampling approach to functional parameters is optimal. Since our prior induces a correlation between function evaluations of our parameter, e.g. $\theta_i(v_1)$ and $\theta_i(v_2)$, they will also have posterior correlation. Therefore, approaches such as Hamiltonian Monte Carlo [86] could outperform our

method. This becomes especially important if the function of interest is a very smooth Gaussian process as the correlations will be very high. Hamiltonian Monte Carlo does come with the computational cost of gradient evaluation.

2.7 Details

2.7.1 Differential equation model for $\mathrm{o}(t)$

Our differential equation system defined by

$$\frac{\mathrm{d}\boldsymbol{x}}{\mathrm{dt}} = \boldsymbol{A}(\boldsymbol{\theta})\boldsymbol{x},$$

where $\boldsymbol{x}$ is the vector of the proportion of ion channels in the first closed, open, inactive, second closed and third closed states and

$$\boldsymbol{A}(\boldsymbol{\theta}) = \begin{bmatrix} -\theta_5 - \theta_1 - \theta_2 & \theta_2 & \theta_6 & \theta_1 & 0 \\ \theta_1 & -\theta_3 - \theta_2 & \theta_4 & 0 & 0 \\ \theta_5 & \theta_3 & -\theta_6 - \theta_4 & 0 & 0 \\ \theta_2 & 0 & 0 & -\theta_1 - \theta_2 & \theta_1 \\ 0 & 0 & 0 & \theta_2 & -\theta_1 \end{bmatrix}.$$

Now we consider the voltage clamped experiments. Let $\boldsymbol{x}_0$ be the solution at time 0. For section Section 2.5 we set $\boldsymbol{x}_0 = [0, 0, 0, 0, 1]^\mathsf{T}$ since the cells were held at a very low voltage (-100 mV) prior to being held at the clamped voltage, thereby insuring most ion channels are in the "most" closed position. Then we have that

$$o(t; \boldsymbol{\theta}) = [0, 1, 0, 0, 0] \exp(t\boldsymbol{A})\boldsymbol{x}_0.$$

2.7.2 Prior parameters

For our model parameters θ_i, we must specify a prior on σ_i and μ_i for each $i = 1, \ldots, 6$. To set these, we broke the parameters θ_i into two categories based on the parameter values given in [21]. The first three values are expected to be much larger than the other three for all voltages. The smaller the value in the outlined Markov model, the

less of a chance that any ion channels will change between those two states. Thus, these small values can be fixed without affecting our output i^*. Therefore θ_4, θ_5 and θ_6 are fixed to 0, 0.0084 and 0, close to the values found in [21]. In terms of our model, can induce this in our model by setting $\sigma_i^2 = 0$ for $i = 4, 5, 6$. For $i = 1, 2, 3$, we choose a prior on σ_i^2 of an inverse gamma with shape parameter 1 and rate parameter 4. This distribution is a broad with mean 1/4. For the prior on each $\boldsymbol{\beta}$ we choose multivariate normal with mean $[-1, 0]^\mathsf{T}$ and covariance matrix $\text{diag}(4, (2/35)^2)$ which gives us a broad prior.

We have priors of

$$b \sim \text{GP}\left(0, \sigma_b^2 R_b\right)$$

and

$$\varepsilon_j \overset{\text{indep.}}{\sim} \text{GP}\left(0, \sigma_\varepsilon^2 R_\varepsilon\right).$$

The hyperparameters consist parameters for the priors on parameters σ_ϵ and σ_b and correlation functions labeled R_ε and R_b. We explicitly list the priors for these values below:

- σ_ε^2 has an inverse gamma prior with shape parameter 50 and rate parameter 0.0000125. This distribution is a broad with mean $0.0000125/50 = (.05/100)^2$ $(\mu A/\mu C)^2$.
- R_ε is the i.i.d. model, or $R_\varepsilon(t, t') = \mathbb{1}_{t=t'}$.
- σ_b^2 has inverse gamma prior with shape parameter 50 and rate parameter 0.0003125. This distribution is a broad with mean $0.0003125/50 = (.25/100)^2$ $(\mu A/\mu C)^2$.
- R_b is the outer product of two Matérn covariance functions with smoothness parameter 2.5,

  $$R_b((t, v), (t', v')) = (1+\sqrt{5}\Delta_t+5\Delta_t^2/3)(1+\sqrt{5}\Delta_v+5\Delta_v^2/3)\exp(-\sqrt{5}(\Delta_v+\Delta_t)),$$

where $\Delta_t = .1^{-1}|\log(t+.4) - \log(t'+.4)|$ and $\Delta_v = 5^{-1}|v - v'|$. The first difference function, Δ_t, was chosen to allow for more time varying behavior in the run-up present in observations seen in Figure 4, possibly generated by the dynamics of the controller.

Lastly, we have the parameters G, G_b and E. We set them as outlined below

- G is set to a fixed value of 0.01 $\mu A/\mu C$ to prevent any identifiability issues with the parameters channel dynamic parameters. This is consistent with the numbers used in [15].

- G_b has a prior of log-normal with $\mu = \log(G/10^4)$ and $\sigma = 2$ where μ and σ are the parameters of the associated normal distribution. This implies we expect some value close to the median of this distribution $\exp(\log(G/10^4)) = G/10^4$. The value of σ is very large, which gives us a broad prior distribution. This number was chosen to insure that our prior implies the we expect the background current to be significantly smaller than the dynamic current.

- E is set to a fixed value of +20mV based on the the ionic concentrations in our experiment.

2.7.3 MCMC and prediction details

For simplicity, denote $\boldsymbol{y}_j$ as the vectorization of $\{y(t_k, v_j)\}_{k=1,\ldots,N}$. We have that from Bayes rule

$$\pi(\log \boldsymbol{\theta}(v_j)|\text{data}, \phi, \{\log \boldsymbol{\theta}(v_k)\}_{k\neq j}) \propto \pi(\boldsymbol{y}_j|\{\boldsymbol{y}_k\}_{k\neq j}, \phi, \{\log \boldsymbol{\theta}(v_k)\}_{k=1}^{M}) \\ \pi(\log \boldsymbol{\theta}(v_j)|\{\log \boldsymbol{\theta}(v_k)\}_{k\neq j}, \phi, \{\boldsymbol{y}_k\}_{k\neq j})$$

and from our prior model

$$\pi(\log \boldsymbol{\theta}(v_j)|\{\log \boldsymbol{\theta}(v_k)\}_{k\neq j}, \phi, \{\boldsymbol{y}_k\}_{k\neq j}) = \pi(\log \boldsymbol{\theta}(v_j)|\{\log \boldsymbol{\theta}(v_k)\}_{k\neq j}). \tag{4}$$

Using equation (4), we have that

$$\begin{aligned}\pi(\log \boldsymbol{\theta}(v_j)|\{\log \boldsymbol{\theta}(v_k)\}_{k\neq j}) &\propto p_1(\log \boldsymbol{\theta}(v_j)|\{\log \boldsymbol{\theta}(v_k)\}_{k\neq j}) \\ &:= \exp\left(-\frac{1}{2}\sum_{i=1}^{6}\sigma_i^{-2}\left(1-\boldsymbol{r}_{i;-j}^{\mathsf{T}}\boldsymbol{R}_{i;-j}^{-1}\boldsymbol{r}_{i;-j}\right)^{-1}\left(\log\theta_i(v_j)-\mu_{i;j|-j}\right)^2\right),\end{aligned}$$

where

$$\mu_{i,j|-j} = \mu_i(v_j) + \boldsymbol{r}_{i;-j}^{\mathsf{T}}\boldsymbol{R}_{i;-j}^{-1}\left(\tilde{\boldsymbol{\theta}}_{i;-j} - \tilde{\boldsymbol{\mu}}_{i;-j}\right),$$

$\tilde{\boldsymbol{\mu}}_{i;-j}$, $\tilde{\boldsymbol{\theta}}_{i;-j}$ and $\boldsymbol{r}_{i;-j}$ are the same as their not subscripted counterparts in Section 2.4 with the jth element removed and $\boldsymbol{R}_{i;-j}$ is the prior correlation matrix of $\tilde{\boldsymbol{\theta}}_{i;-j}$.

We have left to find the conditional distribution of $\boldsymbol{y}_j$. Let $\boldsymbol{b}_j$ be the vectorization of $\{b(t_k, v_j)\}_{k=1,\ldots,N}$, $\boldsymbol{\varepsilon}_j$ be the vectorization of $\{\varepsilon(t_k, v_j)\}_{k=1,\ldots,N}$ and let $\boldsymbol{i}(\boldsymbol{\theta})$ be the vectorization of $\{\mathrm{i}^*(t_k, v_j)\}_{k=1,\ldots,N}$ given $\boldsymbol{\theta}$ and parameter ϕ. Then we have that

$$\boldsymbol{y}_j = \boldsymbol{i}(\boldsymbol{\theta}(v_j)) + \boldsymbol{b}_j + \boldsymbol{\varepsilon}_j.$$

The value of $\boldsymbol{i}(\boldsymbol{\theta}(v_j))$ given $\boldsymbol{\theta}(v_j)$ and ϕ is deterministic. The distribution of $\boldsymbol{\varepsilon}_j$ is independent of $\{\boldsymbol{y}_k\}_{k\neq j}$ and $\{\boldsymbol{\theta}(v_k)\}_{k=1}^M$. Therefore $\boldsymbol{\varepsilon}_j$ given $\{\boldsymbol{y}_k\}_{k\neq j}$ and $\{\boldsymbol{\theta}(v_k)\}_{k=1}^M$ follows its prior distribution given by

$$\mathcal{N}\left(\mathbf{0}, \sigma_\varepsilon^2\boldsymbol{R}_\varepsilon\right),$$

where $\boldsymbol{R}_\varepsilon$ is the correlation matrix corresponding to $\boldsymbol{\varepsilon}_j$, $\mathbf{0}$ is a column vector of zeroes and for this section $\mathcal{N}$ represents both univariate and multivariate normal distributions. Lastly we have to find the distribution $\boldsymbol{b}_j$ given $\{\boldsymbol{y}_k\}_{k\neq j}$ and $\{\boldsymbol{\theta}(v_k)\}_{k=1}^M$. By our prior, we have that this distribution is given by

$$\mathcal{N}\left(\hat{\boldsymbol{b}}_{b_j|y_{-j}}, \sigma_b^2\boldsymbol{R}_{b_j|y_{-j}}\right),$$

where

$$\hat{\boldsymbol{b}}_{b_j|y_{-j}} = \sigma_b^2 \boldsymbol{R}_{b_j,b_{-j}} \left(\sigma_b^2 \boldsymbol{R}_{b_{-j}} + \sigma_\epsilon^2 \boldsymbol{I} \otimes \boldsymbol{R}_\varepsilon\right)^{-1} \left(\boldsymbol{y}_{-j} - \boldsymbol{i}_{-j}\right), \boldsymbol{y}_{-j} = \begin{bmatrix} \boldsymbol{y}_1 \\ \vdots \\ \boldsymbol{y}_{j-1} \\ \boldsymbol{y}_{j+1} \\ \vdots \\ \boldsymbol{y}_M \end{bmatrix}, \boldsymbol{i}_{-j} = \begin{bmatrix} \boldsymbol{i}(\boldsymbol{\theta}(v_1)) \\ \vdots \\ \boldsymbol{i}(\boldsymbol{\theta}(v_{j-1})) \\ \boldsymbol{i}(\boldsymbol{\theta}(v_{j+1})) \\ \vdots \\ \boldsymbol{i}(\boldsymbol{\theta}(v_M)) \end{bmatrix}$$

and

$$\boldsymbol{R}_{b_j|y_{-j}} = \boldsymbol{R}_{b_j} - \sigma_b^2 \boldsymbol{R}_{b_j,b_{-j}} \left(\sigma_b^2 \boldsymbol{R}_{b_{-j}} + \sigma_\epsilon^2 \boldsymbol{I} \otimes \boldsymbol{R}_\varepsilon\right)^{-1} \boldsymbol{R}_{b_j,b_{-j}}^{\mathsf{T}},$$

where $\otimes$ represents the Kronecker product. The matrix $\boldsymbol{I}$ is an identity matrix, $\boldsymbol{R}_{b_j}$ is the correlation matrix corresponding to $\boldsymbol{b}_j$, the matrix $\boldsymbol{R}_{b_{-j}}$ is the correlation matrix corresponding to

$$\boldsymbol{b}_{-j} = \begin{bmatrix} \boldsymbol{b}_1 \\ \vdots \\ \boldsymbol{b}_{j-1} \\ \boldsymbol{b}_{j+1} \\ \vdots \\ \boldsymbol{b}_M \end{bmatrix},$$

and the matrix $\boldsymbol{R}_{b_j,b_{-j}}$ is the cross correlation matrix between the vectors. Combining the above, since the three terms $\boldsymbol{i}(\boldsymbol{\theta}(v_j))$,$\boldsymbol{b}_j$ and $\boldsymbol{\varepsilon}_j$ are independent and multivariate normal, we have that

$$\begin{aligned} &\pi(\boldsymbol{y}_j|\{\boldsymbol{y}_k\}_{k\neq j}, \phi, \{\boldsymbol{\theta}(v_k)\}_{k=1}^M) \propto p_2(\boldsymbol{\theta}(v_j)|\text{data}, \phi, \{\boldsymbol{\theta}(v_k)\}_{k\neq j}) \\ &:= \exp\left(-\frac{1}{2}\left(\boldsymbol{y}_j - \boldsymbol{i}(\boldsymbol{\theta}(v_j)) - \hat{\boldsymbol{b}}_{b_j|y_{-j}}\right)^{\mathsf{T}} \left(\sigma_b^2 \boldsymbol{R}_{b_j|y_{-j}} + \sigma_\varepsilon^2 \boldsymbol{R}_\varepsilon\right)^{-1} \left(\boldsymbol{y}_j - \boldsymbol{i}(\boldsymbol{\theta}(v_j)) - \hat{\boldsymbol{b}}_{b_j|y_{-j}}\right)\right). \end{aligned}$$

Finally we can establish the rejection rate used in our wave sampler described in

Section 2.4 as

$$\begin{aligned}
&\frac{\pi\left(\log\boldsymbol{\theta}(v_j)^{(n)}|\text{data},\phi^{(n-1)},\{\log\boldsymbol{\theta}(v_k)^{(n)}\}_{k<j},\{\log\boldsymbol{\theta}(v_k)^{(n-1)}\}_{k>j}\right)}{\pi\left(\log\boldsymbol{\theta}(v_j)^{(n-1)}|\text{data},\phi^{(n-1)},\{\log\boldsymbol{\theta}(v_k)^{(n)}\}_{k<j},\{\log\boldsymbol{\theta}(v_k)^{(n-1)}\}_{k>j}\right)}\\
=&\frac{p_1\left(\log\boldsymbol{\theta}(v_j)^{(n)}|\{\log\boldsymbol{\theta}(v_k)^{(n)}\}_{k<j},\{\log\boldsymbol{\theta}(v_k)^{(n-1)}\}_{k>j}\right)}{p_1\left(\log\boldsymbol{\theta}(v_j)^{(n-1)}|\{\log\boldsymbol{\theta}(v_k)^{(n)}\}_{k<j},\{\log\boldsymbol{\theta}(v_k)^{(n-1)}\}_{k>j}\right)}\\
&\times\frac{p_2\left(\boldsymbol{\theta}(v_j)^{(n)}|\text{data},\phi^{(n-1)},\{\boldsymbol{\theta}(v_k)^{(n)}\}_{k<j},\{\boldsymbol{\theta}(v_k)^{(n-1)}\}_{k>j}\right)}{p_2\left(\boldsymbol{\theta}(v_j)^{(n-1)}|\text{data},\phi^{(n-1)},\{\boldsymbol{\theta}(v_k)^{(n)}\}_{k<j},\{\boldsymbol{\theta}(v_k)^{(n-1)}\}_{k>j}\right)}.
\end{aligned}\tag{5}$$

Let $\boldsymbol{y}^\mathsf{T}=[\boldsymbol{y}_1^\mathsf{T},\ldots,\boldsymbol{y}_M^\mathsf{T}]$ and let $\boldsymbol{R}_b$ be correlation matrix corresponding to $\boldsymbol{b}^\mathsf{T}=[\boldsymbol{b}_1^\mathsf{T},\ldots,\boldsymbol{b}_M^\mathsf{T}]$. We have that the covariance matrix of $\boldsymbol{y}$ is

$$\boldsymbol{\Sigma}_y=\sigma_b^2\boldsymbol{R}_b+\sigma_\varepsilon^2\boldsymbol{I}\otimes\boldsymbol{R}_\varepsilon.$$

Let $\boldsymbol{o}(\boldsymbol{\theta}(v_j))$ be the vectorization of $\{\mathrm{o}(t_k;\boldsymbol{\theta}(v_j))\}_{k=1}^N$.

Now say we are trying to predict $y(t,v)$ where v was not included in our experiment. Then we can use the relation

$$\pi(y(t,v)|\text{data})=\int\pi(y(t,v)|\boldsymbol{\theta}(v),\phi,\text{data})\pi(\boldsymbol{\theta}(v),\phi|\text{data})\mathrm{d}\{\boldsymbol{\theta}(v),\phi\},$$

where we can sample from $\boldsymbol{\theta}(v)$ using equation (3) and ϕ using our proposed sampler. Similar to above, we have the predictive distribution $\pi(y(t,v)|\boldsymbol{\theta}(v),\phi,\text{data})$ is the same as

$$\mathcal{N}\left(\mathrm{i}^*(t,v;\boldsymbol{\theta}(v),\phi)+\hat{b}_{b|y}(t,v),\sigma_{b|y}^2+\sigma_\varepsilon^2\right),\tag{6}$$

where

$$\hat{b}_y(t,v)=\sigma_b^2\boldsymbol{r}_b(t,v)\left(\sigma_b^2\boldsymbol{R}_b+\sigma_\varepsilon^2\boldsymbol{I}\otimes\boldsymbol{R}_\varepsilon\right)^{-1}(\boldsymbol{y}-\boldsymbol{i})$$

and

$$\sigma_{b|y}^2=\sigma_b^2-\sigma_b^4\boldsymbol{r}_b(t,v)\left(\sigma_b^2\boldsymbol{R}_b+\sigma_\varepsilon^2\boldsymbol{I}\otimes\boldsymbol{R}_\varepsilon\right)^{-1}\boldsymbol{r}_b^\mathsf{T}(t,v).$$

The matrix $\boldsymbol{I}$ is an identity matrix, $\boldsymbol{R}_b$ is the correlation matrix corresponding to $\boldsymbol{b}$, and the vector $\boldsymbol{r}_b(t,v)$ is cross correlation between $b(t,v)$ and $\boldsymbol{b}$.

Let the prior on σ_i^2 be inverse gamma with shape parameter α_i and rate parameter γ_i. Say we are given all other parameters. We can use standard analysis tools to find

the posterior distribution is inverse gamma with shape parameter

$$\alpha_i + \frac{NM}{2}$$

and rate parameter

$$\gamma_i + \frac{1}{2}\left(\tilde{\boldsymbol{\theta}}_i - \tilde{\boldsymbol{\mu}}_i\right)^{\mathsf{T}} \boldsymbol{R}_i^{-1}\left(\tilde{\boldsymbol{\theta}}_i - \tilde{\boldsymbol{\mu}}_i\right).$$

Let the prior on $\boldsymbol{\beta}_i$ be multivariate normal with mean $\boldsymbol{\mu}_{\beta_i}$ and standard deviation $\boldsymbol{\Sigma}_{\beta_i}$. Say we are given all other parameters. Let $\boldsymbol{X} = (1, (v_i + 35))_{i=1}^{M}$. Then we have that the posterior distribution is multivariate normal with mean

$$\left(\sigma_i^{-2}\boldsymbol{X}^{\mathsf{T}}\boldsymbol{R}_i^{-1}\boldsymbol{X} + \boldsymbol{\Sigma}_{\beta_i}^{-1}\right)^{-1}\left(\boldsymbol{\Sigma}_{\beta_i}^{-1}\boldsymbol{\mu}_{\beta_i} + \sigma_i^{-2}\boldsymbol{X}^{\mathsf{T}}\boldsymbol{R}_i^{-1}\tilde{\boldsymbol{\theta}}_i\right)$$

and covariance matrix

$$\left(\sigma_i^{-2}\boldsymbol{X}^{\mathsf{T}}\boldsymbol{R}_i^{-1}\boldsymbol{X} + \boldsymbol{\Sigma}_{\beta_i}^{-1}\right)^{-1}.$$

Chapter III

FAST PREDICTION OF DETERMINISTIC FUNCTIONS USING SPARSE GRID EXPERIMENTAL DESIGNS

In this chapter, we investigate a new approach to resolve this problem: by restricting ourselves a general class of designs, accurate non-approximative predictors can be with found with significantly less computational expense. The results require a separable covariance function, see the first chapter of this thesis. This work was published and is typeset differently in [94].

This class of experimental designs is termed *sparse grid designs* and is based on the structure of eponymic interpolation and quadrature rules. Sparse grid designs [112] have been used with in conjunction with polynomial rules [126, 7, 132, 88, 131], but these designs have not gained popularity among users of random field models. Here, we encourage the use of sparse grid designs by demonstrating computational procedures to be used with these designs where the predictor can be computed very quickly.

Section 1.3 will briefly describe two broad types of existing designs and identify deficiencies of those existing types. Section 3.1 will explain the definition of sparse grid designs and then the following sections will discuss three important topics:

- Section 3.2 explains how we can exploit the structures used in building sparse grid designs to achieve extreme computational gains when building the predictor. Our algorithm computes $\boldsymbol{w}$ by inverting several small matrices versus one large matrix. This algorithm is derived from the result that $\hat{y}(\boldsymbol{x}_0)$ can be written as the tensor product of linear operators, see Theorem 3.6.1 in Sppendix 3.6.2.

- Section 3.3 goes on to demonstrate that we can estimate unknown parameters of the random field with similar computational quickness. Of note is Theorem 3.3.1, which gives an expression for the determinant of the matrix $\mathbf{\Sigma}$ that can be evaluated quickly.

- Section 3.4 illustrates that sparse grid designs perform well even when the input is high dimensional. We conduct empirical comparisons that demonstrate good performance of these designs which supports the positive asymptotic arguments proven previously [120].

Section 3.5 will offer some discussions on the role of these designs and the creation of optimal sparse grid designs.

3.1 Sparse grid designs

This section will discuss the construction of sparse grid experimental designs which are closely associated with sparse grid interpolation and quadrature rules. To build these designs first specify a nested sequence of one dimensional experimental designs for each $i = 1, \ldots, d$ denoted $\mathcal{X}_{i,j}$, where $\mathcal{X}_{i,j} \subseteq \mathcal{X}_{i,j+1}, j = 0, 1, 2, \ldots$, and $\mathcal{X}_{i,0} = \emptyset$. Designs defined for a single dimension, e.g. $\mathcal{X}_{i,j}$, are termed *component designs* in this work. The nested feature of these sequences is important for our case. The general literature related to sparse grid rules does not require this property. Here, it is necessary for the stated results to hold.

Sparse grid designs are therefore defined as

$$\mathcal{X}_{SG}(\eta) = \bigcup_{\vec{j} \in \mathbb{G}(\eta)} \mathcal{X}_{1,j_1} \times \mathcal{X}_{2,j_2} \times \cdots \times \mathcal{X}_{d,j_d}, \tag{7}$$

where $\eta \geq d$ is an integer that represents the level of the construction and $\mathbb{G}(\eta) = \left\{ \vec{j} \in \mathbb{N}^d | \sum_{i=1}^{d} j_i = \eta \right\}$. Here we use the overhead arrow to distinguish the vector of indices, $\vec{j} = [j_1, \ldots, j_d]$ from a scalar index. Increasing the value of η results in denser

Table 1: Sample size of sparse grid designs with level of construction η, dimension d and $\#\mathcal{X}_{i,j} = h(j)$ for all i. The values of c and c_0 are some constant integers bigger than zero. The last line is from [126].

$h(j), j > 0$	$N_{SG}(\eta)$	Bound on $N_{SG}(\eta)$
cj	$c^d\binom{\eta}{d}$	$(c\eta)^d/d!$
$c(j-1)+1$	$\sum_{k=0}^{\min(d,\eta-d)} c^k\binom{d}{k}\binom{\eta-d}{k}$	$c^{\eta-d}\binom{\eta}{d}$ if $\eta \leq 2d$
$c_0(c^j-1)$	$c_0^d(c-1)^d\sum_{j=0}^{\eta-d} c^j\binom{j+d-1}{d-1}$	$c_0^d\,(c-1)^{d-1}\,c^{\eta-d+1}\binom{\eta-1}{d-1}$

designs. Figure 8 illustrates the construction of the two dimensional designs seen in Figure 9. The details of the component designs can be seen in section 3.6.1.

Unlike many other design alternatives, sparse grid designs are not defined via a given sample size. The sample size of the resulting sparse grid design is a complicated endeavor to compute *a-priori*. After the dimension d and level of construction η, a major contributing factor to the sample size, $N_{SG}(\eta) := \#\mathcal{X}_{SG}(\eta)$, is the sizes of the component designs. The sample size of a sparse grid design is given by

$$N_{SG}(\eta) = \sum_{\vec{j}\in\mathbb{J}(\eta)} \prod_{i=1}^{d} \#\mathcal{X}_{i,j} - \#\mathcal{X}_{i,j-1},$$

where $\mathbb{J}(\eta) = \left\{\vec{j} \in \mathbb{N}^d | \sum_{i=1}^d j_i \leq \eta\right\}$. Table 2 presents some shortcut calculations of the sample size along with some bounds when $\#\mathcal{X}_{i,j} = \#\mathcal{X}_{k,j} = h(j)$ for all i and k.

The proper selection of the points in the component designs is essential to achieving good performance of the overall sparse grid design. Establishing good component designs can lead to a good sparse grid design, but interaction between dimensions is an important consideration.

3.2 Fast prediction with sparse grid designs

This section will propose an algorithm that shows the major advantage of sparse grid designs: the availability of fast predictors. Section 3.6.2 justifies the proposed algorithm by describing a predictor in the form of a tensor product of linear maps and then Theorem 3.6.1 demonstrates that conjectured predictor is the same as $\hat{y}(\boldsymbol{x}_0)$.

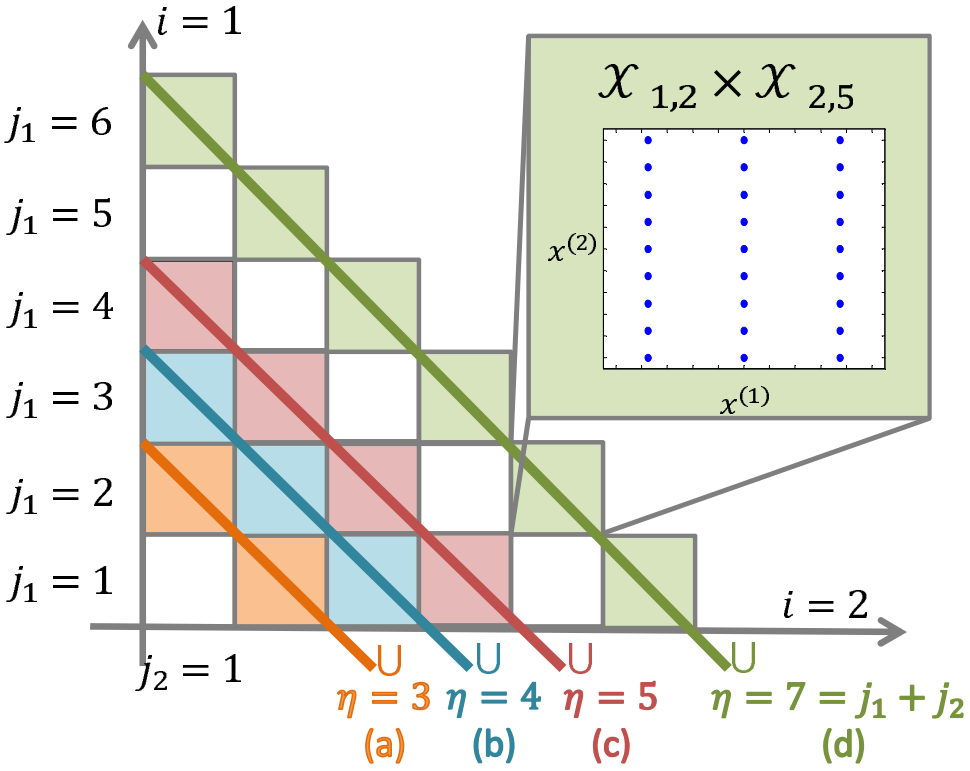

Figure 8: Diagram of the construction of the two dimensional designs seen in Figure 9. Each box represents $\mathcal{X}_{1,j_1} \times \mathcal{X}_{2,j_2}$. The dark lines pass through lattice designs creating the union of the sets featured in Figure 9.

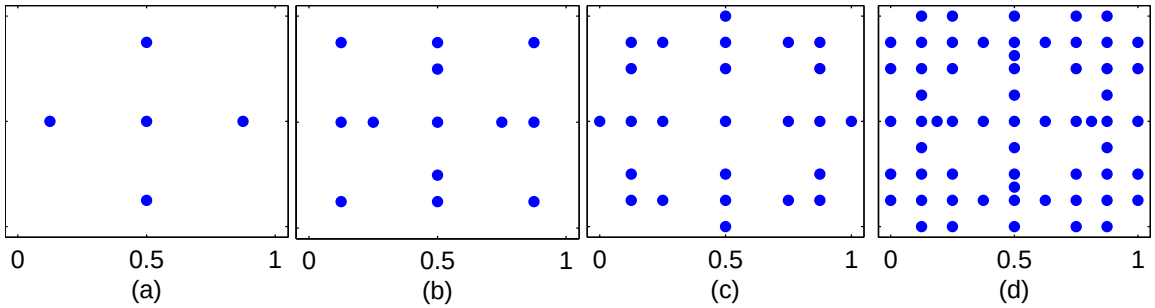

Figure 9: Sparse grid designs associated with Figure 8 where $d = 2$ and $\eta = 3$ (a), 4 (b), 5 (c), and 7 (d). The details of the component designs used for this figure can be seen in section 3.6.1.

Here we show how to build the weight vector, $\boldsymbol{w}$, by inverting covariance matrices associated with the component designs which are relativity small compared to $\boldsymbol{\Sigma}$. Therefore, our proposed method results in a faster computation of $\boldsymbol{w}$ than a method that computes $\boldsymbol{w}$ from direct inversion of $\boldsymbol{\Sigma}$ when N is large. This is the same mechanism that is used to construct fast predictors with lattice designs. But unlike lattice designs, we show sparse grid designs perform well in cases where the input is high dimensional in Section 3.4.

Algorithm 1 lists the proposed algorithm for computing $\boldsymbol{w}$. In the algorithm, the matrices $\boldsymbol{S}_{i,j}$ are composed of elements $C_i(x, x')$, for all $x, x' \in \mathcal{X}_{i,j}$. Also, the vectors $\boldsymbol{y}_{\vec{j}}$, $\boldsymbol{\mu}_{\vec{j}}$ and $\boldsymbol{w}_{\vec{j}}$ denote subvectors of $\boldsymbol{y}$, $\boldsymbol{\mu}$ and $\boldsymbol{w}$ at indices corresponding to $\mathcal{X}_{1,j_1} \times \mathcal{X}_{2,j_2} \times \ldots \times \mathcal{X}_{d,j_d}$ for all $\vec{j} \in \mathbb{J}(\eta)$.

Algorithm 1 Proposed algorithm for the fast computation of $\boldsymbol{w}$ when the design is $\mathcal{X}_{SG}(\eta)$. Here, $a(\vec{j}) = (-1)^{\eta - |\vec{j}|} \binom{d-1}{\eta - |\vec{j}|}$ and $\mathbb{P}(\eta) = \left\{ \vec{j} \in \mathbb{N}^d | \max(d, \eta - d + 1) \leq \sum_{i=1}^{d} j_i \leq \eta \right\}$.

Initialize $\boldsymbol{w} = \boldsymbol{0}$
For all $\vec{j} \in \mathbb{P}(\eta)$
$\quad \boldsymbol{w}_{\vec{j}} = \boldsymbol{w}_{\vec{j}} + a(\vec{j}) \left(\bigotimes_{i=1}^{d} \boldsymbol{S}_{i,j_i}^{-1} \right) \left(\boldsymbol{y}_{\vec{j}} - \boldsymbol{\mu}_{\vec{j}} \right)$

Another important feature of predictors in general is the presence of a predictive variance,

$$\mathbb{E}_{\boldsymbol{Y}=\boldsymbol{y}} \left(\hat{y}(\boldsymbol{x}_0) - Y(\boldsymbol{x}_0) \right)^2 = C(\boldsymbol{x}_0, \boldsymbol{x}_0) - \boldsymbol{\sigma}^{\mathsf{T}}(\boldsymbol{x}_0) \boldsymbol{\Sigma}^{-1} \boldsymbol{\sigma}(\boldsymbol{x}_0),$$

where the subscript $\boldsymbol{Y} = \boldsymbol{y}$ on the expectation implies we condition on that case. As noted before, computation of $\boldsymbol{\Sigma}^{-1}$ is an undesirable operation. Luckily, this operation can be avoided by using sparse grid designs. As demonstrated in Section 3.6.3, when employing a sparse grid design the predictive variance is given by

$$\mathbb{E}_{\boldsymbol{Y}=\boldsymbol{y}} \left(\hat{y}(\boldsymbol{x}_0) - Y(\boldsymbol{x}_0) \right)^2 = C(\boldsymbol{x}_0, \boldsymbol{x}_0) - \sum_{\vec{j} \in \mathbb{J}(\eta)} \prod_{i=1}^{d} \Delta_{i,j_i}(\boldsymbol{x}_0), \tag{8}$$

where $\Delta_{i,j}(\boldsymbol{x}_0) = \varepsilon_{i,j-1}(\boldsymbol{x}_0) - \varepsilon_{i,j}(\boldsymbol{x}_0)$ and $\varepsilon_{i,j}$ is defined as the expected squared prediction error in one dimension with covariance C_i and design $\mathcal{X}_{i,j}$. After substituting

known relations, we have that

$$\varepsilon_{i,j}(\boldsymbol{x}_0) = C_i(x_0^{(i)}, x_0^{(i)}) - \boldsymbol{s}_{i,j}^{\mathsf{T}}(x_0^{(i)})\boldsymbol{S}_{i,j}^{-1}\boldsymbol{s}_{i,j}(x_0^{(i)}),$$

where the elements of the vector $\boldsymbol{s}_{i,j}(x_0^{(i)})$ are $C_i(x_0^{(i)}, x)$ for all $x \in \mathcal{X}_{i,j}$.

3.3 Fast prediction with unknown parameters

The previous section assumed that both mean, $\mu(\cdot)$, and covariance, $C(\cdot,\cdot)$, are exactly known. This is often not assumed in practical situations. Instead, these functions are given general structures with unknown parameters which we denote θ. Two major paradigms exist for prediction when θ is unknown: (i) simply use an estimate for θ based on the observations and predict using (1) or (ii) Bayesian approaches [106]. For either method, the typical formulae require computation of both the determinant and inverse of $\boldsymbol{\Sigma}$, which are costly when N is large. This section develops the methods to avoid these computations. For expositional simplicity, this section will outline the first method and leave the full Bayesian method for future work.

The estimate of θ we consider will be the *maximum likelihood estimate* (MLE), which is denoted $\hat{\theta}$. We therefore term the predictor that uses this estimate as the *MLE-predictor*, which will be used for comparisons in Section 3.4.2. We first explain the typical general structures of $\mu(\cdot)$ and $C(\cdot,\cdot)$ in Section 3.3.1 and then we describe the traditional forms of the estimate $\hat{\theta}$ and problems with them in Section 3.3.2. Section 3.3.3 then explains fast methods to find $\hat{\theta}$ in this setting.

3.3.1 General setting

The structures of μ, C and θ in this section are borrowed from [106] and are widely employed. We assume that the mean is a linear combination of $p \geq 1$ basis functions, $f_1(\cdot), \ldots, f_p(\cdot)$, and the covariance function is scaled such that $C(\cdot,\cdot) = \sigma^2 R(\cdot,\cdot;\phi)$, where $R(\boldsymbol{x}_1, \boldsymbol{x}_2; \phi) = \prod_{i=1}^{d} R_i(x_1^{(i)}, x_2^{(i)}; \phi)$ is a correlation function and σ^2 represents the variance of $y(\boldsymbol{x}) - \mu(\boldsymbol{x})$. The parameter ϕ is a general parameter or group of

parameters that can represent unknown aspects of $R(\cdot, \cdot; \phi)$ that affect the lengthscale and differentiability of the realized response $y(\cdot)$. We now have the following case

$$Y(\cdot) \sim GP\left(\sum_{k=1}^{p} \beta_k f_k(\cdot), \sigma^2 R(\cdot, \cdot; \phi)\right),$$

where $\theta = \{\beta_1, \ldots, \beta_p, \sigma^2, \phi\}$ is the set of unknown parameters.

3.3.2 Traditional computation of the MLE

The logarithm of the probability density of the observations $\boldsymbol{y}$ with $\theta = \{\beta_1, \ldots, \beta_p, \sigma^2, \phi\}$, called the *log-likelihood*, is given by (up to a constant)

$$L(\boldsymbol{\beta}, \sigma^2, \phi) = -\frac{1}{2}\left(N \log(\sigma^2) + \log|\boldsymbol{R}_\phi| + (\boldsymbol{y} - \boldsymbol{F}\boldsymbol{\beta})^\mathsf{T} \boldsymbol{R}_\phi^{-1} (\boldsymbol{y} - \boldsymbol{F}\boldsymbol{\beta}) / \sigma^2\right),$$

where $|\boldsymbol{A}|$ represents the determinant of a matrix $\boldsymbol{A}$, $\boldsymbol{R}_\phi$ is the $N \times N$ correlation matrix of $\boldsymbol{y}$ when parameter ϕ is used, $\boldsymbol{\beta} = [\beta_1, \ldots, \beta_p]^\mathsf{T}$, and

$$\boldsymbol{F} = \begin{bmatrix} f_1(\boldsymbol{x}_1) & \ldots & f_p(\boldsymbol{x}_1) \\ f_1(\boldsymbol{x}_2) & \ldots & f_p(\boldsymbol{x}_2) \\ \vdots & \vdots & \vdots \\ f_1(\boldsymbol{x}_N) & \ldots & f_p(\boldsymbol{x}_N) \end{bmatrix}.$$

Our goal is to solve the optimization problem

$$\hat{\theta} = \text{argmax}_{\beta, \sigma^2, \phi}\, L(\boldsymbol{\beta}, \sigma^2, \phi).$$

There are closed form maximum likelihood estimates for both $\boldsymbol{\beta}$ and σ^2 given ϕ which we denote $\hat{\boldsymbol{\beta}}_\phi$ and $\hat{\sigma}^2_\phi$. They are

$$\hat{\boldsymbol{\beta}}_\phi = \left(\boldsymbol{F}^\mathsf{T} \boldsymbol{R}_\phi^{-1} \boldsymbol{F}\right)^{-1} \boldsymbol{F}^\mathsf{T} \boldsymbol{R}_\phi^{-1} \boldsymbol{y}$$

and

$$\hat{\sigma}^2_\phi = N^{-1}\left(\boldsymbol{y} - \boldsymbol{F}\hat{\boldsymbol{\beta}}_\phi\right)^\mathsf{T} \boldsymbol{R}_\phi^{-1}\left(\boldsymbol{y} - \boldsymbol{F}\hat{\boldsymbol{\beta}}_\phi\right).$$

Then, ϕ is found by generic numerical maximization, i.e.

$$\hat{\phi} = \text{argmax}_\phi\, L(\hat{\boldsymbol{\beta}}_\phi, \hat{\sigma}^2_\phi, \phi).$$

The problem with using these methods directly is that $\hat{\boldsymbol{\beta}}_\phi$ and $\hat{\sigma}^2_\phi$ require inversion of the $N \times N$ matrix $\boldsymbol{R}_\phi$. Additionally, $L(\hat{\boldsymbol{\beta}}_\phi, \hat{\sigma}^2_\phi, \phi)$ still contains the term $\log|\boldsymbol{R}_\phi|$, which is often as cumbersome as finding $\boldsymbol{R}_\phi^{-1}$. The remainder section proposes alternatives to these methods that are faster. Specifically, we will be able to compute $\hat{\boldsymbol{\beta}}_\phi$, $\hat{\sigma}^2_\phi$ and $\log|\boldsymbol{R}_\phi|$ without ever storing or operating directly on $\boldsymbol{R}_\phi$.

3.3.3 Proposed fast computation of the MLE

To introduce our fast-to-compute maximum likelihood estimate, we first describe a generalization of Slgorithm 1 seen in Slgorithm 2. Algorithm 2 computes

$$Q\left(\boldsymbol{A}; \otimes_{i=1}^{d} C_i\right) := \boldsymbol{\Sigma}^{-1}\boldsymbol{A},$$

where $\boldsymbol{A}$ is any $N \times m$ matrix and m is any positive integer. The notation "$\otimes_{i=1}^{d} C_i$" implies we have a separable covariance with each covariance function being represented by $C_i(\cdot,\cdot)$. The computations in Slgorithm 2 do not require the direct inversion of $\boldsymbol{\Sigma}$ and therefore avoid the major computational problems of the traditional method. The validity of Slgorithm 2 is implied by the validity of Slgorithm 1.

Algorithm 2 Fast computation of $Q(\boldsymbol{A}; \otimes_{i=1}^{d} C_i) = \boldsymbol{\Sigma}^{-1}\boldsymbol{A}$ when the design is $\mathcal{X}_{SG}(\eta)$ and $\boldsymbol{A}$ is any $N \times m$ matrix where m is any positive integer. The notation "$\otimes_{i=1}^{d} C_i$" implies we have a separable covariance with each covariance function being represented by $C_i(\cdot,\cdot)$. The notation $\boldsymbol{A}_{\vec{j},\cdot}$ means the matrix with rows that correspond to $\mathcal{X}_{1,j_1} \times \mathcal{X}_{2,j_2} \times \ldots \times \mathcal{X}_{d,j_d}$ and all columns of $\boldsymbol{A}$. Section 3.2 defines $\mathbb{P}(\eta)$, $a(\vec{j})$, and $\boldsymbol{S}_{i,j}$.

Initialize $\tilde{\boldsymbol{A}}$ as an $N \times m$ matrix with all 0 entries.
For all $\vec{j} \in \mathbb{P}(\eta)$
$\quad \tilde{\boldsymbol{A}}_{\vec{j},\cdot} = \tilde{\boldsymbol{A}}_{\vec{j},\cdot} + a(\vec{j}) \left(\bigotimes_{i=1}^{d} \boldsymbol{S}_{i,j_i}^{-1}\right) \boldsymbol{A}_{\vec{j},\cdot}$
Output $\tilde{\boldsymbol{A}}$.

Now we can establish our maximum likelihood estimates for $\boldsymbol{\beta}$ and σ^2 that do not require inversion of the $N \times N$ matrix $\boldsymbol{R}_\phi$. For a given ϕ,

$$\hat{\boldsymbol{\beta}}_\phi = \left(\left[Q(\boldsymbol{F}; \otimes_{i=1}^{d} R_i(\phi))\right]^{\mathsf{T}} \boldsymbol{F}\right)^{-1} \left[Q(\boldsymbol{F}; \otimes_{i=1}^{d} R_i(\phi))\right]^{\mathsf{T}} \boldsymbol{y}$$

and

$$\hat{\sigma}_\phi^2 = N^{-1}\left[Q(\boldsymbol{y} - \boldsymbol{F}\hat{\boldsymbol{\beta}}_\phi; \otimes_{i=1}^d R_i(\phi))\right]^\mathsf{T}\left(\boldsymbol{y} - \boldsymbol{F}\hat{\boldsymbol{\beta}}_\phi\right).$$

The last step to find the MLE requires maximization of $L(\hat{\boldsymbol{\beta}}_\phi, \hat{\sigma}_\phi^2, \phi)$ with respect to ϕ. This expression contains the term $\log|\boldsymbol{R}_\phi|$ which, as mentioned before, is expensive to compute. Therefore, we demonstrate the following theorem related to the expression of the determinant that only involves determinants of component covariance matrices, $\boldsymbol{S}_{i,j}$. The proof lies in the section.

Theorem 3.3.1. *If $\mathcal{X} = \mathcal{X}_{SG}(\eta)$, then*

$$\log|\boldsymbol{\Sigma}| = \sum\nolimits_{\vec{j}\in\mathbb{J}(\eta)} \sum\nolimits_{i=1}^d \left(\log|\boldsymbol{S}_{i,j_i}| - \log|\boldsymbol{S}_{i,j_i-1}|\right) \cdot \prod\nolimits_{k\neq i} \#\mathcal{X}_{k,j_k} - \#\mathcal{X}_{k,j_k-1}$$

where $|\boldsymbol{S}_{i,0}| := 1$ for all i.

By using $R(\boldsymbol{x}_1, \boldsymbol{x}_2; \phi) = \prod_{i=1}^d R_i(x_1^{(i)}, x_2^{(i)}; \phi)$ as the covariance function in the formula in the above theorem, we gain an expression for $\log|\boldsymbol{R}_\phi|$ without directly computing the determinant of an $N \times N$ matrix. Once $\hat{\phi}$ is found, this gives us $\hat{\theta} = \{\hat{\boldsymbol{\beta}}_{\hat{\phi}}, \hat{\sigma}_{\hat{\phi}}^2, \hat{\phi}\}$.

3.4 Prediction performance comparisons

Thus far, this chapter has established that we can build predictors quickly when sparse grid designs are used. However, an issue of critical importance is how well the resulting predictors perform. This section seeks to compare the predictive performance resulting from sparse grid designs to the more common designs discussed in Section 1.3. Our core findings can be summarized as follows: (i) both sparse grid and space-filling designs outperform lattice designs, (ii) sparse grid designs appear competitive with space-filling designs for smooth functions and inferior to space-filling designs for very rough functions, and (iii) the time taken to find the MLE-predictor using sparse grid designs can be orders of magnitude less than the time taken using the traditional methods.

Before we begin numerical comparisons, it might be helpful to take a historical look at sparse grid designs. The prevalence of sparse grid designs in the numerical approximation literature can be owed to the demonstrated efficiency of the designs even when the input is of high dimension. It has been shown if $X = [0,1]^d$, using sparse grid designs with component designs of the form $\mathcal{X}_{i,j} = \{1/2^j, \ldots, (2^j - 1)/2^j\}$ is an asymptotically efficient design strategy under the symmetric separable covariance structure [120, 130, 103]. These designs are also known as *hyperbolic cross points*. The key point discovered in the previous analysis is that sparse grid designs are asymptotically efficient regardless of dimension and lattice designs become increasingly inefficient as the dimension grows large. Therefore, we anticipate that sparse grid designs outperform lattices in high dimensions.

The sparse grid designs used in this section were constructed from component designs that are symmetric across dimensions and details of the component designs are in section 3.6.1. These appeared to be at least competitive if not superior to hyperbolic cross points in a simulation study comparable to Section 3.4.1. The space-filling designs were constructed by using the scrambled Sobol sequence described in [78]. Maximin Latin hypercube designs that were generated via the R package `lhs` produced inferior distance metrics for large sample sizes but the same conclusions as the ones presented in this section. The lattice design designs used for comparison in Section 3.4.1 were $\{1/4, 3/4\}^{10}$, $\{0, 1/2, 1\}^{10}$, and $\{0, 1/3, 2/3, 1\}^{10}$.

3.4.1 Comparison via average prediction error

This subsection will investigate the mean square prediction error resulting from various experimental designs when the mean and covariance structures are known. This can be thought of as the average mean squared prediction error over all possible sample paths, $y(\cdot)$, drawn from a Gaussian process with a specified covariance function.

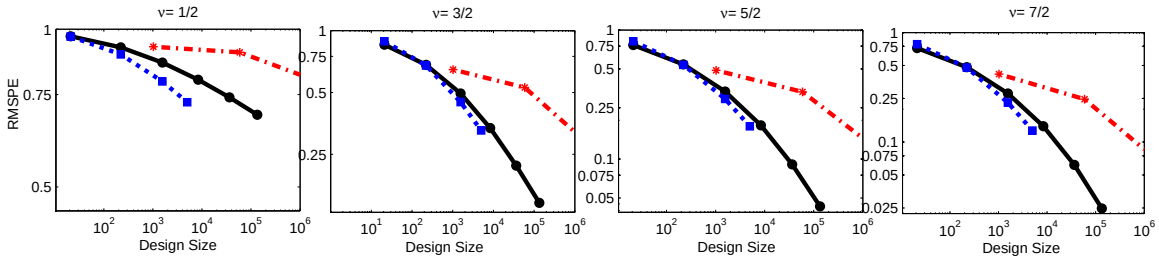

Figure 10: Root mean square prediction errors (RMSPE) associated with sparse grid designs (solid), space-filling designs (small dashes), and lattice designs (dashed-dotted) for the simulation discussed in Section 3.4.1. The random fields are located in $[0,1]^{10}$ and defined with a Matérn covariance function where $\phi = .75$ and ν varies.

Furthermore, we seek to examine the impact of the smoothness of $y(\cdot)$ on the effectiveness of the design strategies. To allow for the introduction of varying levels of smoothness, this section will use the Matérn class of covariance functions,

$$C_i(x,x') = \frac{1}{2^{\nu-1}\Gamma(\nu)}\left(\sqrt{2\nu}h\right)^{\nu}\mathcal{K}_{\nu}\left(\sqrt{2\nu}h\right), \tag{9}$$

where $\mathcal{K}_{\nu}$ is the modified Bessel function of order $\nu > 0$ and $h = |x - x'|/\phi$. The use of this covariance class allows us to independently adjust a smoothness parameter ν, where the sample paths are $\lceil \nu - 1 \rceil$ times differentiable [53]. For simplicity, this subsection uses homogenous covariance in every dimension. For the case when $d = 10$ and $\phi = .75$, Figure 10 compares the average root mean squared prediction error (RMSPE) resulting from the design strategies computed through 1000 Monte Carlo samples on $[0,1]^{10}$. Note if $N > 3000$, the RMSPEs for the space-filling designs were not recorded due to numerical instability when inverting the large covariance matrix.

Figure 10 indicates sparse grid designs yield superior performance to lattice designs. The results also demonstrate the similarity of the sparse grid designs and space-filling designs in cases of the existence of at least one derivative. However, sparse grid designs appear inferior to the space-filling designs if the sample path has almost surely no differentiability.

3.4.2 Comparison via deterministic functions

This section will compare the performance of sparse grid designs and space-filling designs on a set of deterministic test functions. For both methods, we assume the mean and covariance structures of the deterministic functions are unknown and use the *MLE-predictor*. For μ, we use a constant mean structure, $\mu(\boldsymbol{x}) = \beta$, and for the covariance function we use a scaled Matérn with $\nu = 5/2$ and single lengthscale parameter ϕ for all dimensions i. This analysis will report the median absolute prediction error, which is more robust to extreme observations compared to the mean square prediction error. The median absolute prediction error will be estimated by the sample median of the absolute prediction error at 1000 randomly selected points in the input space. We consider the following functions: Franke's function [41], the Borehole function [85], the product peak function given by $y(\boldsymbol{x}) = \prod_{i=1}^{d}(1 + 10(x^{(i)} - 1/4)^2)^{-1}$, the corner peak function given by

$$y(\boldsymbol{x}) = \left(1 + d^{-1}\sum\nolimits_{i=1}^{d} x^{(i)}\right)^{-d-1},$$

and the Rosenbrock function given by

$$y(\boldsymbol{x}) = 4\sum\nolimits_{i=1}^{d-1}(x^{(i)} - 1)^2 + 400\sum\nolimits_{i=1}^{d-1}((x^{(i+1)} - .5) - 2(x^{(i)} - .5)^2)^2.$$

With the exception of the Borehole function, all domains are $X = [0, 1]^d$ (the Borehole function was scaled to the unit cube). For the space-filling designs, designs sizes were restricted to cases where memory constraints in MATLAB were not violated on the author's computer.

Figure 11 presents the results of the study. Most functions were similarly estimated using either design strategy. While Franke's function has significantly more bumps and ridges compared to the other functions, making it more difficult to estimate, good prediction of Franke's function based on few observations is possible because the input to the function is located in a 2 dimensional space. At the other

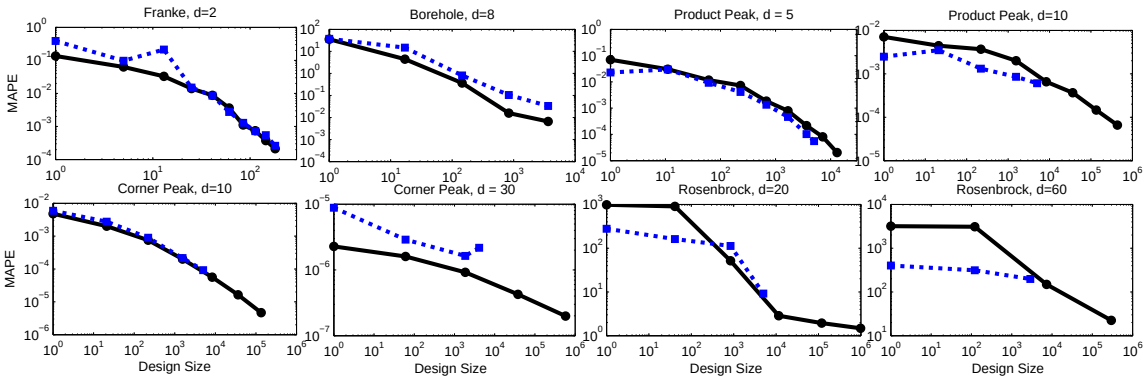

Figure 11: Median absolute prediction errors (MAPE) of the MLE-predictor from Section 3.3 with sparse grid designs (circles, solid line) and space-filling designs (squares, dashed line).

extreme, while the corner peak function is smooth, estimating the function when $d = 30$ is a very challenging task. Using a space-filling design of size 4000 does not do an adequate job of estimating the function as it produces median absolute prediction error of about 10 times more than the best that can be achieved using a sparse grid design with a much larger design size. Similar effects are seen when attempting to estimate the Rosenbrock function in 60 dimensions.

Figure 12 compares the computational time needed to find both $\hat{\theta}$ and the weights $\boldsymbol{w}$ using both the traditional method and the proposed method for the MLE-predictors used to produce Figure 11. The method to find the MLE-predictor was described in Section 3.3. There was three cases where the cost of the traditional algorithm with a design size of less than 5000 was more than the proposed algorithm with a design size of nearly a million. While the design sizes attempted for the traditional algorithm were limited for memory and numerical stability reasons, some extrapolation emphasizes the problem with using the traditional algorithm on experiments with huge sample sizes. A sample size of a million points would require roughly 10^7 seconds, or 115.7 days, to find the MLE-predictor. By using a sparse grid design, we are able to compute the MLE-predictor based on a million observations in a fraction of that time, about

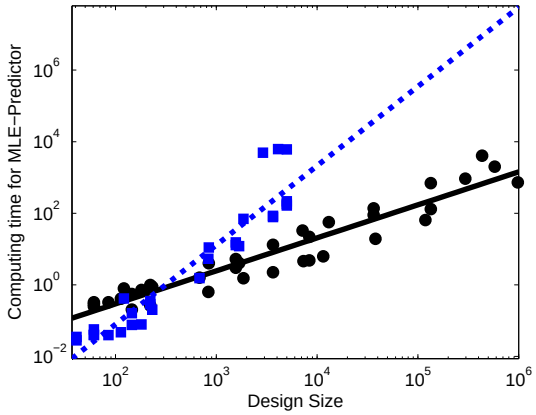

Figure 12: Computation time (in seconds) needed to find the MLE-predictor from Section 3.3 using the proposed method for sparse grid experimental designs (circles) and the traditional method with space-filling designs (squares). The solid line (sparse grid designs) and the dashed line (space-filling designs) represent least squares fits of the model log computational time $= \beta_0 + \beta_1 \log N$ to the respective data.

15 minutes (770 seconds).

3.5 Discussion

The proposed sparse grid designs deviate from the traditional space-filling framework and utilize lattice structures to construct efficient designs. Sparse grid designs appear to be competitive with common space-filling designs in terms of prediction, but space-filling designs appear to outperform sparse grid designs in simulations where the underlying function has no differentiability. Based on the discussions at the end of Section 3.4.1, these early results may extend to cases that can be classified as rough functions.

Sparse grid designs are an enormously flexible framework and this work has not yet realized their full potential. A topic not discussed at length in this work are *optimal* sparse grid designs, which might be able to close any small performance gaps between sparse grid and space-filling designs. Optimality is dictated by the choice of design criteria, which has previously focused on distance measures such as the minimum distance between any two design points. When the sample size grows large, this can

become an expensive metric to compute as it requires $\mathcal{O}(N^2)$ arithmetic operations. Therefore, using the shortcut calculations for integrated prediction error, see equation (8), or maximum entropy, see Theorem 3.3.1, might be faster criteria to compute (see [105] for more information on these criteria).

A problem not yet solved using the proposed designs occurs when the covariance function is not separable. The study of these situations merits more work. As an example, if the sample path contains distinct areas with differing behavior, the assumption of local separability might be a more apt modeling strategy. Using only local separability assumptions, methods similar to [47] could be employed with local sparse grid designs that study heterogeneous sections of the function.

3.6 Details

One notational difference between the body of the chapter and these appendices: Since the proofs for theorems 1 and 2 are demonstrated through induction by treating the level of construction η and the dimension d as variables, we use the indexing (η, d) for the design $\mathcal{X}_{SG}(\eta, d)$, the design size $N_{SG}(\eta, d)$, the index sets $\mathbb{J}(\eta, d)$ and $\mathbb{P}(\eta, d)$, and the covariance matrix $\mathbf{\Sigma}(\eta, d)$. Also, the symbol $\setminus$ means 'set-minus', i.e. $A \setminus B$ is the elements in A that are not in B.

3.6.1 Component designs used for the sparse grid designs in this work

The sparse grid design in Figure 1, subplot c, was created where $d = 2$, $\eta = 6$ and $\mathcal{X}_{i,1}, \mathcal{X}_{i,2} \setminus \mathcal{X}_{i,1}, \mathcal{X}_{i,3} \setminus \mathcal{X}_{i,2}$, $\mathcal{X}_{i,4} \setminus \mathcal{X}_{i,3}$ and $\mathcal{X}_{i,5} \setminus \mathcal{X}_{i,4}$ are $\{.5\}, \{0, 1\}, \{.25, .75\}, \{.375, .625\}$ and $\{.125, .875\}$ respectively for $i = 1$ and 2.

The sparse grid design in Figure 9 was created with component designs such that $\mathcal{X}_{i,1}$, $\mathcal{X}_{i,2} \setminus \mathcal{X}_{i,1}$, $\mathcal{X}_{i,3} \setminus \mathcal{X}_{i,2}$, $\mathcal{X}_{i,4} \setminus \mathcal{X}_{i,3}$, $\mathcal{X}_{i,5} \setminus \mathcal{X}_{i,4}$, $\mathcal{X}_{i,6} \setminus \mathcal{X}_{i,5}$, and $\mathcal{X}_{i,7} \setminus \mathcal{X}_{i,6}$ are $\{.5\}$, $\{.125, .875\}$, $\{.25, .75\}$, $\{0, 1\}$, $\{.375, .625\}$, $\{0.1875, 0.8125\}$, and $\{0.0625, 0.9375\}$ respectively for all i. These component designs were chosen through an ad-hoc method, but are essentially based on maintaining good spread of points as η increases.

The component designs used in Figure 9 are used to construct higher dimensional designs used in Section 3.4.

3.6.2 Proof that Slgorithm 1 produces correct $\boldsymbol{w}$

The correctness of $\boldsymbol{w}$ produced by Slgorithm 1 is difficult to understand without the use of linear operators, therefore we will rephrase $\hat{y}(\boldsymbol{x}_0)$ discussed in Chapter 1 in terms of a linear operator. Let $\mathcal{F}$ be a function space of functions that map X to $\mathbb{R}$. Let $\mathcal{P} : \mathcal{F} \to \mathbb{R}$ be a *predictor operator* with respect to $\mathcal{X} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$ if $\mathcal{P}f = \sum_{k=1}^{N} q_k f(\boldsymbol{x}_k)$ where $q_k \in \mathbb{R}$. The following definition explains an *optimal* predictor operator.

Definition 3.6.1. *A predictor operator $\mathcal{P}$ is termed* optimal *with respect to $\boldsymbol{x}_0$ and $\mathcal{X} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$ if $\mathcal{P}f = \sum_{k=1}^{N} q_k f(\boldsymbol{x}_k)$ and*

$$\{q_1, \ldots, q_N\} = \text{argmin}_{\{\alpha_1,\ldots,\alpha_N\}\in\mathbb{R}^N} \mathbb{E}\left(\mu(\boldsymbol{x}_0) + \sum_{k=1}^{N} \alpha_k[Y(\boldsymbol{x}_k) - \mu(\boldsymbol{x}_k)] - Y(\boldsymbol{x}_0)\right)^2.$$

A predictor operator $\mathcal{P}$ is termed optimal because

$$\hat{y}(\boldsymbol{x}_0) = \mu(\boldsymbol{x}_0) + \mathcal{P}[y - \mu],$$

is the *best linear unbiased predictor* of $y(\boldsymbol{x}_0)$ given the observations $\boldsymbol{Y} = \boldsymbol{y}$ when $\mathcal{P}$ is optimal with respect to $\boldsymbol{x}_0$ and $\mathcal{X}$ [106].

In general, the optimal predictor operator is when q_k is the kth element in $\boldsymbol{\sigma}^{\mathsf{T}}(\boldsymbol{x}_0)\boldsymbol{\Sigma}^{-1}$. There are cases where the predictor operator is unique. Therefore, we only need to show a clever form of the optimal predictor operator that agrees with the $\boldsymbol{w}$ produced by Slgorithm 1 to complete our argument.

Now we define a sequence, $j = 0, 1, 2, \ldots$, of predictor operators, $\mathcal{P}_{i,j}$, for each dimension i. These are the optimal predictor operators with respect to $x_0^{(i)}$ and $\mathcal{X}_{i,j}$ when the dimension of the input is 1 and the covariance function is C_i.

To find the desired form of the optimal predictor operator with respect to sparse grid designs, one could guess that the quadrature rule of [112] will be of great use. In

our terms, the [112] quadrature rule can be interpreted as the predictor operator

$$\mathcal{P}(\eta, d) = \sum_{\vec{j} \in \mathbb{J}(\eta,d)} \bigotimes_{i=1}^{d} \mathcal{P}_{i,j_i} - \mathcal{P}_{i,j_i-1}, \tag{10}$$

where the $\otimes$ symbol for linear operators is the tensor product. .

While this form of $\mathcal{P}(\eta, d)$ is known, the optimality of $\mathcal{P}(\eta, d)$ in the situation discussed has not yet to been proved to the author's knowledge. [126] study the case where $\mathcal{X}_{i,j} = \mathcal{X}_{k,j}$ for all i and k. They show an optimality property with respect to an L_∞ norm, which they term *worst case*. [126] go on to state in passing that one could verify that (10) is mean of the predictive distribution and therefore optimal in our setting, but they do not demonstrate it in that work. Here, we formally state and demonstrate this result.

Theorem 3.6.1. *The predictor operator $\mathcal{P}(\eta, d)$ is optimal with respect to $\boldsymbol{x}_0$ and $\mathcal{X}_{SG}(\eta, d)$. Furthermore, $\mathcal{P}(\eta, d)$ can be written in the form*

$$\mathcal{P}(\eta, d) = \sum_{\vec{j} \in \mathbb{P}(\eta,d)} a(\vec{j}) \bigotimes_{i=1}^{d} \mathcal{P}_{i,j_i}, \tag{11}$$

where $a(\vec{j}) = (-1)^{\eta - |\vec{j}|} \binom{d-1}{\eta - |\vec{j}|}$ *and* $\mathbb{P}(\eta, d) = \left\{ \vec{j} \in \mathbb{N}^d | \max(d, \eta - d + 1) \leq \sum_{i=1}^{d} j_i \leq \eta \right\}$.

The different statements of (10) and (11) are important to note. The predictor operator in (10) is theoretically intuitive as it geometrically explains how we maintain orthogonality as η grows and allows for the subsequent proof. However, if we were to attempt to use (10) directly, each term in the sum would require us to sum 2^d terms after expansion, which may temper any computational advantages the lattice structure yields. [126] show that (10) can be written of the form (11). This result is simply an algebraic manipulation and requires no conditions regarding optimality, but the result allows us to easily use (10).

The fact that (11) is the optimal predictor operator verifies that Slgorithm 1 produces correct $\boldsymbol{w}$.

3.6.2.1 Proof of Theorem 3.6.1

Proof. Let

$$\mathcal{P}(\eta, d)f = \sum_{k=1}^{N} q_k f(\boldsymbol{x}_k)$$

where $q_k \in \mathbb{R}$ and $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\} = \mathcal{X}_{SG}(\eta, d)$. We need to show that

$$\{q_1, \ldots, q_N\} = \text{argmin}_{\{\alpha_1,\ldots,\alpha_N\}\in\mathbb{R}^N} \mathbb{E}\left(\mu(\boldsymbol{x}_0) + \sum_{k=1}^{N} \alpha_k[Y(\boldsymbol{x}_k) - \mu(\boldsymbol{x}_k)] - Y(\boldsymbol{x}_0)\right)^2.$$

Since $\mathbb{E}(Y(\boldsymbol{x}) - \mu(\boldsymbol{x})) = 0$, the objective function is minimized when the covariance between $\sum_{k=1}^{N} \alpha_k Y(\boldsymbol{x}_k) - Y(\boldsymbol{x}_0)$ and values of Y at all points in $\mathcal{X}_{SG}(\eta, d)$ is 0. Thus, we need to show that

$$\text{cov}(\mathcal{P}(\eta, d)Y - Y(\boldsymbol{x}_0), Y(\boldsymbol{x}_k)) = 0,$$

for all $\boldsymbol{x}_k \in \mathcal{X}_{SG}(\eta, d)$.

If $d = 1$, the theorem is clearly true for all $\eta \geq d$. Assume that the theorem is true for $d-1$ and all $\eta \geq d-1$; we will show that it is true for d and η. This demonstrates the result by an induction argument.

We have that

$$\text{cov}(\mathcal{P}(\eta, d)Y - Y(\boldsymbol{x}_0), Y(\boldsymbol{x}_k)) = -C(\boldsymbol{x}_0, \boldsymbol{x}_k) + \mathcal{P}(\eta, d)\mathbb{E}\left[\{Y(\boldsymbol{x}_k) - \mu(\boldsymbol{x}_k)\}(Y - \mu)\right]. \tag{12}$$

Observe that

$$\begin{aligned}
\mathcal{P}(\eta, d)\mathbb{E}\left[\{Y(\boldsymbol{x}_k) - \mu(\boldsymbol{x}_k)\}(Y - \mu)\right] &= \mathcal{P}(\eta, d)C\left(\cdot, \boldsymbol{x}_k\right) \\
&= \sum_{\vec{j}\in\mathbb{J}(\eta,d)} \bigotimes_{i=1}^{d} \mathcal{P}_{i,j_i} C_i\left(\cdot, x_k^{(i)}\right) - \mathcal{P}_{i,j_i-1} C_i\left(\cdot, x_k^{(i)}\right) \\
&= \sum_{\vec{j}\in\mathbb{J}(\eta-1,d-1)} \prod_{i=1}^{d-1} \mathcal{P}_{i,j_i} C_i\left(\cdot, x_k^{(i)}\right) - \mathcal{P}_{i,j_i-1} C_i\left(\cdot, x_k^{(i)}\right) \\
&\quad \cdot \sum_{j_d=1}^{\eta-|\vec{j}|} \mathcal{P}_{d,j_d} C_d\left(\cdot, x_k^{(d)}\right) - \mathcal{P}_{d,j_d-1} C_d\left(\cdot, x_k^{(d)}\right).
\end{aligned} \tag{13}$$

Since $\mathcal{P}_{i,j}$ is the optimal predictor operator with respect to $x_0^{(i)}$ and $\mathcal{X}_{i,j}$, $\mathcal{P}_{i,j}C_i(\cdot,x)-C_i(x_0^{(i)},x)=0$ if $x\in\mathcal{X}_{i,j}$. Let

$$\mathbb{K}=\{\vec{j}|\boldsymbol{x}_k\in\mathcal{X}_{i,j_1}\times\mathcal{X}_{2,j_2}\times\cdots\times\mathcal{X}_{d,j_d},\vec{j}\in\mathbb{J}(\eta,d)\},$$

and let $\vec{a}=\text{argmin}_{\vec{j}\in\mathbb{K}}|\vec{j}|$. Since sparse grid designs have nested component designs, if $j_i\geq a_i$, then $\mathcal{P}_{i,j_i}C_i\left(\cdot,x_k^{(i)}\right)=\mathcal{P}_{i,j_1+1}C_i\left(\cdot,x_k^{(i)}\right)=C_i(x_0^{(i)},x_k^{(i)})$, since $x_k^{(i)}\in\mathcal{X}_{i,j_i}\subset\mathcal{X}_{i,j_i+1}$. This implies if $\vec{j}\not\leq\vec{a}$, then $\prod_{i=1}^{d}\left(\mathcal{P}_{i,j_i}C_i\left(\cdot,x_k^{(i)}\right)-\mathcal{P}_{i,j_i-1}C_i\left(\cdot,x_k^{(i)}\right)\right)=0$. Then (13) can be rewritten as

$$\begin{aligned}\mathcal{P}(\eta,d)\mathbb{E}\left[\{Y(\boldsymbol{x}_k)-\mu(\boldsymbol{x}_k)\}(Y-\mu)\right]=&\\ \sum_{\vec{j}\in\mathbb{J}(\eta-1,d-1)}\prod_{i=1}^{d-1}&\left(\mathcal{P}_{i,j_i}C_i\left(\cdot,x_k^{(i)}\right)-\mathcal{P}_{i,j_i-1}C_i\left(\cdot,x_k^{(i)}\right)\right)\\ \cdot\sum_{j_d=1}^{\max(\eta-a_1-\cdots-a_{d-1},\eta-|\vec{j}|)}&\mathcal{P}_{d,j_d}C_d\left(\cdot,x_k^{(d)}\right)-\mathcal{P}_{d,j_d-1}C_d\left(\cdot,x_k^{(d)}\right)\end{aligned}\quad(14)$$

Also, if $j_d>a_d$ then $\mathcal{P}_{d,j_d}C_d\left(\cdot,x_k^{(d)}\right)-\mathcal{P}_{d,j_d-1}C_d\left(\cdot,x_k^{(d)}\right)=0$ and

$$\sum_{i=1}^{d}a_i\leq\eta\Rightarrow a_d\leq max(\eta-a_1-\cdots-a_{d-1},\eta-j_1-\cdots-j_{d-1}),$$

which implies

$$\begin{aligned}\sum_{j_d=1}^{\max(\eta-a_1-\cdots-a_{d-1},\eta-|\vec{j}|)}&\mathcal{P}_{d,j_d}C_d\left(\cdot,x_k^{(d)}\right)-\mathcal{P}_{d,j_d-1}C_d\left(\cdot,x_k^{(d)}\right)\\ &=\sum_{j_d=1}^{a_d}\mathcal{P}_{d,j_d}C_d\left(\cdot,x_k^{(d)}\right)-\mathcal{P}_{d,j_d-1}C_d\left(\cdot,x_k^{(d)}\right)\\ &=C_d\left(x_0^{(d)},x_k^{(d)}\right).\end{aligned}\quad(15)$$

Plugging (15) into (14) yields

$$\begin{aligned}\mathcal{P}(\eta,d)\mathbb{E}\left[\{Y(\boldsymbol{x}_k)-\mu(\boldsymbol{x}_k)\}(Y-\mu)\right]=&\\ C_d\left(x_0^{(d)},x_k^{(d)}\right)\cdot\sum_{\vec{j}\in\mathbb{J}(\eta-1,d-1)}\prod_{i=1}^{d-1}&\mathcal{P}_{i,j_i}C_i\left(\cdot,x_k^{(i)}\right)-\mathcal{P}_{i,j_i-1}C_i\left(\cdot,x_k^{(i)}\right).\end{aligned}$$

By the induction assumption, the theorem is true for $d-1$, which means that for $\eta-1$ and $d-1$, (12) is equal to zero. Therefore,

$$\sum_{\vec{j}\in\mathbb{J}(\eta-1,d-1)}\prod_{i=1}^{d-1}\mathcal{P}_{i,j_i}C_i\left(\cdot,x_k^{(i)}\right)-\mathcal{P}_{i,j_i-1}C_i\left(\cdot,x_k^{(i)}\right)=\prod_{i=1}^{d-1}C_i\left(x_0^{(i)},x_k^{(i)}\right).$$

This gives us the desired result for d,

$$\mathcal{P}(\eta,d)\mathbb{E}\left[\{Y(\boldsymbol{x}_k)-\mu(\boldsymbol{x}_k)\}(Y-\mu)\right]=\prod_{i=1}^{d}C_i\left(x_0^{(i)},x_k^{(i)}\right)=C(\boldsymbol{x}_0,\boldsymbol{x}_k).$$

Inserting this into (12) yields the major result that (10) is the optimal predictor operator.

In [126], they demonstrate through combinatorial relations and algebraic manipulations that (10) can be simplified to (11). □

3.6.3 Proof that (8) is the MSPE

Due to Theorem 3.6.1,

$$\begin{aligned}\mathbb{E}\left(\hat{Y}(\boldsymbol{x}_0)-Y(\boldsymbol{x}_0)\right)^2&=\operatorname{var}(Y(\boldsymbol{x}_0))-\mathcal{P}(\eta,d)\mathbb{E}\left[\{Y(\boldsymbol{x}_0)-\mu(\boldsymbol{x}_0)\}(Y-\mu)\right]\\&=C(\boldsymbol{x}_0,\boldsymbol{x}_0)-\sum_{\vec{j}\in\mathbb{J}(\eta,d)}\prod_{i=1}^{d}\mathcal{P}_{i,j}C_i(\cdot,x_0^{(i)})-\mathcal{P}_{i,j-1}C_i(\cdot,x_0^{(i)}).\end{aligned}$$

Because $\mathcal{P}_{i,j}$ is the optimal predictor operator in one dimension with respect to $x_0^{(i)}$ and $\mathcal{X}(\eta,d)$, we have

$$\mathbb{E}\left(\hat{Y}(\boldsymbol{x}_0)-Y(\boldsymbol{x}_0)\right)^2=\operatorname{var}(Y(\boldsymbol{x}_0))-\sum_{\vec{j}\in\mathbb{J}(\eta,d)}\prod_{i=1}^{d}\Delta_{i,j_i},$$

where $\Delta_{i,j}$ is defined in Section 3.2.

Lastly, we have that since $\hat{Y}(\cdot)$ is an affine map from $\boldsymbol{Y}$ and $Y(\cdot)$ follows a Gaussian process, $\hat{Y}(\boldsymbol{x}_0)-Y(x_0)$ and $\boldsymbol{Y}$ are jointly multivariate normal. By Theorem 3.6.1, there is 0 covariance between them. Therefore $\hat{Y}(\boldsymbol{x}_0)-Y(x_0)$ is independent of $\boldsymbol{Y}$ and we can condition the expectation on the left-hand-side on $\boldsymbol{Y}=\boldsymbol{y}$ without affecting the right-hand-side.

3.6.4 Proof of Theorem 3.3.1

Proof. If $d = 1$, the theorem is clearly true for all $\eta \geq d$. We now prove this result by induction. Assume the theorem is true for $d-1$ and all $\eta \geq d-1$.

To demonstrate this result, we require the use of the Schur complement. Let $\boldsymbol{M} = \left[\boldsymbol{A}, \boldsymbol{B}; \boldsymbol{B}^{\mathsf{T}}, \boldsymbol{C}\right]$. The Schur complement of $\boldsymbol{M}$ with respect to $\boldsymbol{A}$, expressed $\boldsymbol{M}/\boldsymbol{A}$, is defined by $\boldsymbol{C} - \boldsymbol{B}^{\mathsf{T}}\boldsymbol{A}^{-1}\boldsymbol{B}$ (if $\boldsymbol{A}$ is invertible). The determinant quotient property of the Schur complement is $|\boldsymbol{M}/\boldsymbol{A}| = |\boldsymbol{M}||\boldsymbol{A}|^{-1}$. The theorem can be rewritten as

$$|\boldsymbol{\Sigma}(\eta, d)| = \prod_{\vec{j} \in \mathbb{J}(\eta,d)} \prod_{i=1}^{d} |\boldsymbol{S}_{i,j_i} / \boldsymbol{S}_{i,j_i-1}|^{\prod_{k \neq i} \#\mathcal{X}_{k,j_k} - \#\mathcal{X}_{k,j_k-1}}.$$

We also require following result:

$$|\boldsymbol{A} \otimes \boldsymbol{B}| = |\boldsymbol{A}|^m |\boldsymbol{B}|^n, \tag{16}$$

where $\boldsymbol{A}$ and $\boldsymbol{B}$ are $n \times n$ and $m \times m$ sized matrices, respectively.

We will use the notation $\boldsymbol{\Sigma}(\eta, d; \mathcal{X}_0)$ to denote the submatrix of $\boldsymbol{\Sigma}(\eta, d)$ with respect to the elements which correspond to $\mathcal{X}_0 \subset \mathcal{X}$. Expanding the term $|\boldsymbol{\Sigma}(\eta, d)|$ with respect the quotient property

$$\begin{aligned} |\boldsymbol{\Sigma}(\eta, d)| = &|\boldsymbol{\Sigma}(\eta, d) / \boldsymbol{\Sigma}(\eta, d; \mathcal{X}_{SG}(\eta-1, d-1) \times \mathcal{X}_{d,1} \setminus \mathcal{X}_{d,0})| \\ &|\boldsymbol{\Sigma}(\eta, d; \mathcal{X}_{SG}(\eta-1, d-1) \times \mathcal{X}_{d,1} \setminus \mathcal{X}_{d,0})|. \end{aligned} \tag{17}$$

Let

$$\boldsymbol{Q} = \boldsymbol{\Sigma}(\eta, d) / \boldsymbol{\Sigma}(\eta, d; \mathcal{X}_{SG}(\eta-1, d-1) \times \mathcal{X}_{d,1} \setminus \mathcal{X}_{d,0}).$$

Now, observe the elements of $\boldsymbol{Q}$ that correspond to $\mathcal{X}_{SG}(\eta-2, d-1) \times \mathcal{X}_{d,2} \setminus \mathcal{X}_{d,1}$,

$$\boldsymbol{Q}(\mathcal{X}_{SG}(\eta-2, d-1) \times \mathcal{X}_{d,2} \setminus \mathcal{X}_{d,1}) = \boldsymbol{A} - \boldsymbol{B}^{\mathsf{T}}\boldsymbol{C}^{-1}\boldsymbol{B},$$

where $\boldsymbol{A}$ is a covariance matrix corresponding to $\mathcal{X}_{SG}(\eta-2, d-1) \times \mathcal{X}_{d,2} \setminus \mathcal{X}_{d,1}$, $\boldsymbol{C}$ is a covariance matrix corresponding to $\mathcal{X}_{SG}(\eta-1, d-1) \times \mathcal{X}_{d,1} \setminus \mathcal{X}_{d,0}$, and $\boldsymbol{B}$ is the

cross covariance. Since $\mathcal{X}_{SG}(\eta-2,d-1)\subset\mathcal{X}_{SG}(\eta-1,d-1)$ and $\mathcal{X}_{d,1}\subset\mathcal{X}_{d,2}$,

$$\boldsymbol{Q}(\mathcal{X}_{SG}(\eta-2,d-1)\times\mathcal{X}_{d,2}\setminus\mathcal{X}_{d,1})=\boldsymbol{\Sigma}\left(\eta-2,d-1\right)\bigotimes\left(\boldsymbol{S}_{d,2}/\boldsymbol{S}_{d,1}\right).$$

So,

$$|\boldsymbol{Q}(\mathcal{X}_{SG}(\eta-2,d-1)\times\mathcal{X}_{d,2}\setminus\mathcal{X}_{d,1})|=|\boldsymbol{\Sigma}(\eta,d;\mathcal{X}_{SG}(\eta-2,d-1)\times\mathcal{X}_{d,2}\setminus\mathcal{X}_{d,1})|,$$

which can be used with (17) to show

$$\begin{aligned}\boldsymbol{\Sigma}(\eta,d)=&|\boldsymbol{\Sigma}(\eta,d)/\boldsymbol{\Sigma}(\eta,d;\mathcal{X}_{SG}(\eta-1,d-1)\times\mathcal{X}_{d,1}\setminus\mathcal{X}_{d,0})\\&/\boldsymbol{\Sigma}(\eta,d;\mathcal{X}_{SG}(\eta-2,d-1)\times\mathcal{X}_{d,2}\setminus\mathcal{X}_{d,1})|\\&|\boldsymbol{\Sigma}(\eta,d;\mathcal{X}_{SG}(\eta-1,d-1)\times\mathcal{X}_{d,1}\setminus\mathcal{X}_{d,0})|\,|\boldsymbol{\Sigma}(\eta,d;\mathcal{X}_{SG}(\eta-2,d-1)\times\mathcal{X}_{d,2}\setminus\mathcal{X}_{d,1})|.\end{aligned}$$

Iterating the expansion to $\eta-d+1$ yields,

$$\begin{aligned}|\boldsymbol{\Sigma}(\eta,d)|=&|\boldsymbol{\Sigma}(\eta,d)/\boldsymbol{\Sigma}(\eta,d;\mathcal{X}_{SG}(\eta-1,d-1)\times\mathcal{X}_{d,1}\setminus\mathcal{X}_{d,0})\\&/\boldsymbol{\Sigma}(\eta,d;\mathcal{X}_{SG}(\eta-2,d-1)\times\mathcal{X}_{d,2}\setminus\mathcal{X}_{d,1})\ /\cdots\\&/\boldsymbol{\Sigma}(\eta,d;\mathcal{X}_{SG}(d-1,d-1)\times\mathcal{X}_{d,\eta-d+1}\setminus\mathcal{X}_{d,\eta-d})\ |\\&\prod_{j_d=1}^{\eta-d+1}|\boldsymbol{\Sigma}(\eta,d;\mathcal{X}_{SG}(\eta-j_d,d-1)\times\mathcal{X}_{d,j_d}\setminus\mathcal{X}_{d,j_d-1})|.\end{aligned}$$

The term outside of the product is the Schur complement of a positive definite matrix with itself, which is an empty matrix. By the Leibniz formula, the determinant is 1. Therefore,

$$|\boldsymbol{\Sigma}(\eta,d)|=\prod_{j_d=1}^{\eta-d+1}|\boldsymbol{\Sigma}(\eta,d;\mathcal{X}_{SG}(\eta-j_d,d-1)\times\mathcal{X}_{d,j_d}\setminus\mathcal{X}_{d,j_d-1})|.$$

With (16), we have

$$|\boldsymbol{\Sigma}(\eta,d)|=\prod_{j_d=1}^{\eta-d+1}|\boldsymbol{S}_{d,j_d}/\boldsymbol{S}_{d,j_d-1}|^{N_{SG}(\eta-j_d,d-1)}\,|\boldsymbol{\Sigma}(\eta-j_d,d-1)|^{\#\mathcal{X}_{d,j_d}-\#\mathcal{X}_{d,j_d-1}}.$$

And by (7) and the induction assumption

$$
\begin{aligned}
|\boldsymbol{\Sigma}(\eta, d)| &= \prod_{j_d=1}^{\eta-d+1} \prod_{\vec{j}\in\mathbb{J}(\eta-j_d,d-1)} |\boldsymbol{S}_{d,j_d}/\boldsymbol{S}_{d,j_d-1}|^{\prod_{k\neq d}\#\mathcal{X}_{k,j_k}-\#\mathcal{X}_{k,j_k-1}} \\
&\quad \prod_{i=1}^{d-1} |\boldsymbol{S}_{i,j_i}/\boldsymbol{S}_{i,j_i-1}|^{\prod_{k\neq i}\#\mathcal{X}_{k,j_k}-\#\mathcal{X}_{k,j_k-1}}, \\
&= \prod_{\vec{j}\in\mathbb{J}(\eta,d)} \prod_{i=1}^{d} |\boldsymbol{S}_{i,j_i}/\boldsymbol{S}_{i,j_i-1}|^{\prod_{k\neq i}\#\mathcal{X}_{k,j_k}-\#\mathcal{X}_{k,j_k-1}},
\end{aligned}
$$

which demonstrates the result. □

Chapter IV

BUILDING ACCURATE EMULATORS FOR STOCHASTIC SIMULATIONS VIA QUANTILE KRIGING

4.1 Background

This chapter describes the development of emulators through a framework termed *quantile kriging*, which allows for non-parametric representations of the stochastic behavior of the output. The first step in the framework is running a designed experiment with replications at different sets of inputs. Using this information, we establish an empirical predictive distribution by using the Gaussian process prediction conditioned on the estimated quantiles at each set of inputs in the experiment.

While this two step procedure can be considered informal from the traditional Bayesian viewpoint, this framework results in emulators with an explicit predictive distribution and an associated fast sampling scheme. Furthermore, this work studies asymptotic properties of this methodology that yields practical insights. For example, experiments consisting of replications at sets of different inputs is nearly universally accepted among users of simulations [2], but the rational is not always justified. We demonstrate, under certain regulatory conditions, a result that can be summarized as follows (see Section 4.4):

> *By using an experiment that has the appropriate ratio of replications to sets of different inputs, we can achieve an optimal rate of convergence.*

To the authors' knowledge, this is the first result of this type for stochastic emulators.

The basic idea of the proposed framework is to estimate the underlying *quantiles* of the distribution while accounting for epistemic uncertainty. After discussing

the modeling strategy in Section 5.2, we propose a method to develop predictive distributions in Section 4.3. Sections 4.4 and 4.5 demonstrate the advantages of this framework by investigating the asymptotic efficiency and two illustrations, respectively. Section 4.6 briefly discusses some conclusions, comparisons to other work and possible extensions of this work.

4.2 Simulation Metamodeling

As mentioned in Section 5.1, emulators are traditionally developed using random field metamodels [105, 106] which provide the ability to model simulation output without the restrictive linear or low-order polynomial assumptions. Additionally, random field metamodels provide the ability to account for both the *aleatoric* uncertainty, which differs on each realization of the simulation, and *epistemic* uncertainty, representing the uncertainty caused by unknown aspects of the system.

The basic idea of the traditional metamodel [69, 2, 93] is to assume the output is the sum of a deterministic, but unknown, mean $M(x)$ and a random variable $\varepsilon(x)$ representing the stochastic behavior of the simulation, i.e.

$$Y(x) = M(x) + \varepsilon(x).$$
$$\varepsilon(x) \sim \mathcal{N}(0, \sigma^2(x)),$$

where $\sigma^2(x)$ represents the variance of the output, which is a function of the inputs. The value of $Y(x)$ represents a single draw from the simulation with inputs x. In the interest of generality, we assume only independent samples are drawn from the simulation model. Since $M(x)$ is unknown but deterministic, the framework from deterministic simulations is adopted, e.g. [105], and a distributional assumption is placed on $M(x)$ that represents our uncertainty. The deterministic value of M has a prior distribution of $\mathcal{GP}(\mu(\cdot), C(\cdot,\cdot))$, where $\mathcal{GP}$ denotes a Gaussian process with a trend function $\mu(x)$ and a covariance structure $cov\left(M(x), M(x')\right) = C(x, x')$. Here,

the distribution of $\varepsilon(x)$ represents the *aleatoric* uncertainty, caused by the stochasticity present in the simulation, and the distribution of $M(x)$ represents the *epistemic* uncertainty, a result of our lack of knowledge about the true mean function.

However, this approach is limited by the normality assumption on $\varepsilon(x)$, which, as mentioned in the introduction, is often invalid. Let $Q_\alpha(x)$ represents the α quantile of the distribution of $Y(x) = M(x) + \varepsilon(x)$, i.e. $Q_\alpha(x) = \inf\{t : P(Y(x) \leq t) \geq \alpha\}$. Here, we establish the key idea of the proposed framework: *since $Q_\alpha(x)$ is an unknown function, we ought to attempt to estimate it as a function of x.* Therefore, we model the quantiles as unknown functions of x, and we further assume they are continuous. Emulators typically work by exploiting the continuity, or higher orders of differentiability, of the output (discussed in, for example, [106]). Without any assumptions of this variety, creating predictive distributions would prove futile.

This means that if we observed two replications from the same input x, we assume their respective simulation outputs may differ, but the distribution is the same. Two replications from x and $x' \neq x$ would have *differing* distributions of the simulation output, but similar inputs (measured in distance) implies similar distributions. Therefore, in this metamodel, the aleatoric variation need not be Gaussian, but we assume that the distribution of the simulation output is continuous, meaning as $x \to x_0$, the distribution of the output at x approaches the distribution at x_0.

As an example, let the output, $Y(x)$, be the failure time for a product, which is often modeled as exponentially distributed. Define the mean of $Y(x)$ as $\nu(x) > 0$ and assume the function $\nu(x)$ is smooth. The quantiles of the exponential distribution are given by

$$Q_\alpha(x) = -\ln(1-\alpha)/\nu(x),$$

and output quantiles, clearly, are continuous as a function of x.

Under general assumptions on the distribution of the output, we can establish

the continuity of the quantiles for a wide variety of problems. The following proposition demonstrates this under a broad class of assumptions (proof is located in the supplementary materials).

Proposition 4.2.1. *Let F_x be defined as the cumulative probability distribution such that $Y(x) \sim F_x$. Suppose $F_x(y)$ is continuous with respect to x and y and $F_x(y)$ is a strictly monotonic function with respect to y, then $Q_\alpha(x)$ is a continuous function with respect to x.*

However, this is not an exclusive characterization; the output $Y(x)$ does not need to be a continuous random variable. Consider the following simplified case: you flip a coin, you win x if it lands heads side up and lose x if it lands tails side up. This example is characterized by the following simple distribution

$$F_x(y) = .5\mathbb{1}\{-x \leq y\} + .5\mathbb{1}\{x \leq y\},$$

which corresponds to a discrete distribution. Here, $F_x(y)$ is *not* a continuous or strictly increasing function of y, and therefore does not fit the criteria listed in Proposition 4.2.1, but the quantiles, $Q_\alpha(x) = -x + 2x\mathbb{1}\{\alpha \geq .5\}$, are continuous with respect to x.

4.3 Predictive Distribution

This section outlines the creation of predictive distributions from a designed experiment. We assume there is an experiment that comprises n sets of inputs, denoted $\mathcal{X} = \{x_1, \ldots, x_n\}$, with m replications, which results in a set of observations $y_1(x), y_2(x), \ldots, y_m(x)$ for each $x \in \mathcal{X}$. The choice of $\mathcal{X}$ for use with random field models has been studied in several contexts, and the authors point to [106] and the references therein for more information. In general, the selection of space-filling Latin hypercube designs has yielded positive results.

Using this experiment, this work seeks to develop a predictive distribution for a new input $x_0 \notin \mathcal{X}$, i.e. $\hat{F}_{x_0}$, which is close to the true distribution of the simulation output, $Y(x_0) \sim F_{x_0}$. After some preliminaries, the explicit predictive distribution is described (Section 4.3.2). Discussions of the practical matter of estimation of parameters associated with the Gaussian process model can be seen in Section 4.3.3. The asymptotic analysis of the framework is outlined in Section 4.4.

4.3.1 Expository Development

For simplicity, first consider the case where only a single level α exists and we observe $Q_\alpha(x)$ is known for each $x \in \mathcal{X}$. Since Q_α is assumed continuous, a reasonable prior distribution is that Q_α follows a Gaussian process with mean $\mu(\cdot)$ and covariance $C(\cdot, \cdot)$. From this, an estimate of the α quantile at x_0 s

$$\mu(x_0) + \boldsymbol{\sigma}^\mathsf{T}(x_0)\boldsymbol{\Sigma}^{-1}\left(\boldsymbol{Q}_\alpha - \boldsymbol{\mu}\right), \tag{18}$$

where $\boldsymbol{\Sigma}$ is a matrix composed of elements $C\left(x, x'\right)$ for all $x, x' \in \mathcal{X}$, $\boldsymbol{\sigma}(x_0)$ is a vector composed of elements $C\left(x_0, x\right)$ for all $x \in \mathcal{X}$, $\boldsymbol{\mu}$ is a vector composed of elements $\mu(x)$ for all $x \in \mathcal{X}$ and $\boldsymbol{Q}_\alpha$ is a vector consisting of $Q_\alpha(x)$ for all $x \in \mathcal{X}$. This estimate follows directly from the extensive work in Gaussian process models, e.g. [106].

Now we can develop our approximative procedure. The case being considered assumes $Q_\alpha(x)$ is unknown, but we can draw m replicates from the distribution. Thus, one could replace $\boldsymbol{Q}_\alpha$ with $\tilde{\boldsymbol{Q}}_\alpha = [\tilde{Q}_\alpha(x_1), \ldots, \tilde{Q}_\alpha(x_n)]^\mathsf{T}$, where $\tilde{Q}_\alpha(x)$ is the *estimated* α quantile at a point $x \in \mathcal{X}$. Since Q_α is approximated by $\tilde{Q}_\alpha$, we need to introduce a nugget term to the metamodel to incorporate the random difference between Q_α and $\tilde{Q}_\alpha$, which involves using a covariance function of the form $C(x, x') + \rho^2 \mathbb{1}\{x = x'\}$. The value of ρ^2 is a value that represents the variation of $Q_\alpha(x) - \tilde{Q}_\alpha(x)$. Now, an estimate of $Q_\alpha(x_0)$ is given by

$$\mu(x_0) \qquad + \qquad \boldsymbol{\sigma}^\mathsf{T}(x_0)\left(\boldsymbol{\Sigma} + \rho^2\boldsymbol{I}\right)^{-1}\left(\tilde{\boldsymbol{Q}}_\alpha - \boldsymbol{\mu}\right) \tag{19}$$

The value of ρ is a choice given to the user that should be the standard deviation of $Q_\alpha(x) - \tilde{Q}_\alpha(x)$. Because this value is not known exactly, the choice of this value discussed in Sections 4.3.3 and 4.4.

Since the distribution of the output of a complex simulation often cannot be placed in a parametric class, we propose using the empirical quantile estimates for $\tilde{Q}_\alpha(x)$, i.e.

$$\tilde{Q}_\alpha(x) = \inf\left\{t; \sum_{i=1}^{m} \mathbb{1}(y_i(x) \leq t) \geq m\alpha\right\}. \tag{20}$$

Let $y_{(k)}(x)$ represent the kth order statistic from the m replications at x. First, we recognize $\alpha \in [(k-1)/m, k/m)$ implies that $\tilde{\boldsymbol{Q}}_\alpha = \boldsymbol{y}_{(k)}$ where $\boldsymbol{y}_{(k)}$ is a vector of the kth order statistic from each set of inputs in the experiment, i.e. $\boldsymbol{y}_{(k)} = [y_{(k)}(x_1), \ldots, y_{(k)}(x_n)]^\mathsf{T}$.

4.3.2 Explicit Predictive Distribution

We now establish the proposed predictive distribution. Given a value of α, consider the case where $Q_\alpha(x)$ is known for all α and $x_1, \ldots, x_n$. The predictive distribution of $Q_\alpha(x_0)$ conditioned on the known values yields a normal prediction based on the traditional Gaussian process predictive distribution seen in (18). Suppose also that we have a Markov-like property in the form of $[Q_\alpha(x_0)|\{Q_{\alpha'}(x)|\alpha' \in (0,1), x \in \mathcal{X}\}] = [Q_\alpha(x_0)|\boldsymbol{Q}_\alpha]$ where the brackets indicate a density. We could then establish the predictive distribution as the distribution of $Q_\alpha(x_0)$ given by $[Q_\alpha(x_0)|\boldsymbol{Q}_\alpha][\alpha]$, where $\alpha \sim U(0,1)$. In our work, we do not know $\boldsymbol{Q}_\alpha$ but have estimated $\tilde{\boldsymbol{Q}}_\alpha$. Our two stage approach suggests plugging in $\tilde{\boldsymbol{Q}}_\alpha$ for $\boldsymbol{Q}_\alpha$ and using the predictive distribution in Chapter 1. This yields the following predictive distribution

$$\hat{F}_{x_0}(y) = \frac{1}{m}\sum_{i=1}^{m} F_{\mathcal{N}}(y; a_i(x_0), v(x_0)), \tag{21}$$

where $F_{\mathcal{N}}(y; a, v)$ is a normal distribution with mean a and variance v,

$$a_i(x_0) = \mu(x_0) + \boldsymbol{\sigma}^\mathsf{T}(x_0)\left[\boldsymbol{\Sigma} + \rho^2\boldsymbol{I}\right]^{-1}\left(\boldsymbol{y}_{(i)} - \boldsymbol{\mu}\right)$$

and

$$v(x_0) = C(x_0, x_0) - \boldsymbol{\sigma}^{\mathsf{T}}(x_0)\left[\boldsymbol{\Sigma} + \rho^2 \boldsymbol{I}\right]^{-1} \boldsymbol{\sigma}(x_0) + \rho^2.$$

Because the predictive distribution is a mixture of normal distributions, samples can be quickly drawn. While the predictive distribution is a linear combination of normal distributions, the number of normal distributions increases with the sample size, creating a large number of basis functions for the predictive distribution. Also, the size of the basis functions, measured by the variance of the normal distributions, will naturally shrink as the quantiles are better estimated (n and m are increased). These two features give this predictive distribution the ability to be extremely close to a variety of nonnormal distributions, including heavy tailed and bimodal distributions. We discuss the asymptotic consistency and efficiency $a_{\lfloor \alpha m \rfloor}(x_0)$ as an approximation of $Q_\alpha(x_0)$ under general conditions in Section 4.4.

4.3.3 Choice of C

The assumed properties of the quantiles with respect to x depend on the choice of covariance function C. While C is required to be positive definite, there is a broad array of choices for the covariance function, and the most widely used are the Matérn and Gaussian classes of stationary correlation functions. Covariance functions are typically endowed with a set of parameters, $\boldsymbol{\theta}$, which represent properties of the response surface including lengthscale, differentiability and the Hausdorff dimension. The parameter ρ^2 is often included in $\boldsymbol{\theta}$ because it is unknown, even though it is not explicitly a covariance parameter but an estimation parameter that should change based on the number of observations (see Section 4.4).

We propose estimating these parameters from the data via cross-validation criteria, which has been shown to be an effective implementation strategy for emulators, e.g. [26]. The cross-validation criteria measures the squared prediction error if an observation or set of observations is ignored. Therefore, selection of parameters

by cross-validation intuitively results in parameters with good predictive properties. Here, the predictive performance is measured by the prediction of an estimated quantile from the previous section, i.e. if $\alpha \in [(k-1)/m, k/m)$ then $\tilde{\boldsymbol{Q}}_\alpha = \boldsymbol{y}_{(k)}$. Let $\boldsymbol{\Sigma}$ be defined as in Section 4.3.2, the leave-one-out cross validation for each observation can be quickly calculated using

$$e_{ij}(\boldsymbol{\theta}) = \frac{\left(\boldsymbol{\Sigma}^{-1}(\boldsymbol{\theta})\right)_j}{\left(\boldsymbol{\Sigma}^{-1}(\boldsymbol{\theta})\right)_{jj}} \left(\boldsymbol{y}_{(i)} - \boldsymbol{\mu}\right),$$

where $()_j$ is the jth row and $()_{jj}$ is the jth diagonal element. Therefore, we select covariance parameters as $\hat{\boldsymbol{\theta}} = argmin \sum_{i=1}^{n} \sum_{i=1}^{m} e_{ij}^2(\boldsymbol{\theta})$.

4.4 Asymptotic Efficiency

While the next section will outline an example of the practical benefits of the proposed methodology, this section will show the proposed method is asymptotically consistent and efficient, i.e., under some regulatory conditions, no other framework can do better as $n, m \to \infty$. While sample size restrictions prevent the asymptotic results from being directly utilized, the consistency of the prediction is critical to gauging the performance of the proposed two stage framework.

In this section we assume the Gaussian process model is *stationary*, which implies that the covariance function is only a function of the distance between two sets of inputs, i.e. $C(x, x') = C(x - x')$. Without loss of generality, we further assume the observations are normalized, i.e. zero mean and $C(0) = 1$. The process is then defined by a correlation function, and we emphasize these assumptions on C by denoting a correlation function $\Phi(h)$, $h \in \mathbb{R}^d$. Suppose the design region $x \in \Omega$ is a convex and compact subset of $\mathbb{R}^d$. Since prior distributions for surfaces such as Gaussian processes are difficult to confirm, we demonstrate our results in a general function space. We assume that the underlying true function $Q_\alpha(x)$ lies in the reproducing kernel Hilbert space generated by Φ, denoted as $\mathcal{N}_\Phi(\Omega)$ (for more background on these function spaces, refer to [127]). We will prove that as the sample size increases, the

predictive mean of the Gaussian process model in Chapter 1 converges in probability to the true quantile at the optimal rate.

For $0 < \alpha < 1$, we assume that $Q_\alpha(x_i)$ is estimated by the empirical distribution, as seen in (20), denoted $\tilde{Q}_\alpha(x_i)$. Invoking the representer theorem [124, 108], the predictive mean of the kriging model with a nugget effect, seen in Chapter 1, equals to the solution to the following minimization problem for some $\lambda_{m,n}^2 > 0$

$$\hat{Q}_\alpha(\cdot) = \underset{f \in \mathcal{N}_\Phi(\Omega)}{\operatorname{argmin}} \frac{1}{n} \sum_{i=1}^{n} (\tilde{Q}_\alpha(x_i) - f(x_i))^2 + \lambda_{m,n}^2 \|f\|_{\mathcal{N}_\Phi(\Omega)}^2. \tag{22}$$

Next, we demonstrate the efficiency of $\hat{Q}_\alpha(\cdot)$ for given α under certain regularity conditions:

(A1) $x_i \overset{i.i.d.}{\sim} U(\Omega)$, the uniform distribution over Ω.

(A2) Let F_x be defined as $Y(x) \sim F_x$. For each $x \in \Omega$, there exists $\epsilon > 0$, such that F_x is twice differentiable on interval $B_\epsilon(\alpha, x) = (Q_\alpha(x) - \epsilon, Q_\alpha(x) + \epsilon)$ for every $x \in \Omega$ with first and second derivatives denoted as $f_x(\cdot)$ and $f_x'(\cdot)$ respectively. Furthermore, we assume $c_1 := \inf_{x \in \Omega, t \in B_\epsilon(\alpha,x)} f(x,t) > 0$, and $c_2 := \sup_{x \in \Omega, t \in B_\epsilon(\alpha,x)} |f'(x,t)| < \infty$.

(A3) There exist constants τ with $\lfloor \tau \rfloor > d/2$ and $c_3 > 0$ such that $\bar{\Phi}(w) \leq c_3(1 + \|w\|^2)^{-\tau}$ for $w \in \mathbf{R}^d$, where $\bar{\Phi}$ is the Fourier transformation of Φ.

(A4) $c_4 m^{2\tau/d} \leq n \leq c_5 m^\gamma$ for constants $c_4, c_5 > 0$ and $\gamma \in (2\tau/d, \infty)$.

The next theorem formally states the asymptotic efficiency (proof is located in the supplementarily materials):

Theorem 4.4.1. *Suppose (A1)-(A4) are met. If* $\lambda_{m,n}^2 \sim (mn)^{-2\tau/(2\tau+d)}$ *as* $m, n \to \infty$, *then* $\|\hat{Q}_\alpha(\cdot) - Q_\alpha(\cdot)\|_{L^2(\Omega)} = O_p((mn)^{-\tau/(2\tau+d)})$.

Here, (A1) insures the points in the design $\mathcal{X}$ will eventually fill the space as n grows, (A2) insures the consistency and asymptotic normality of the sample quantile,

(A3) is required to embed the reproducing kernel Hilbert space into a Sobolev space, and (A4) insures a proper ratio of m and n to achieve efficiency.

The bound we establish agrees with the known optimal bounds [117] for non-parametric regression, which implies that as $n, m \to \infty$, $a_{\lfloor \alpha m \rfloor}(x_0)$ approaches $Q_\alpha(x_0)$, and it does so at the fastest rate possible in terms of observed data. While previously developed techniques require the simulation output to be normally distributed, the efficiency shown in this section is *not* limited to the case when the simulation output is Gaussian.

Furthermore, this result addresses the question of replications in experiments for emulators. If the number of replications are properly related to the number of different inputs in the experiment, i.e. $n \asymp m^\gamma$ where $\gamma > 2\tau/d$, we lose *no* efficiency in the emulator. Since τ is a measure of smoothness of the quantiles with respect to x, where large τ represents smooth quantiles, this result can be interpreted as: *if quantiles have little smoothness with respect to x, the experiment should consist of more replications.* This result is somewhat surprising because the information gained by increasing n when studying a rough function is less than the information gained from replications.

The estimate in (22) differs slightly from the one discussed in the section 4.3 because the covariance function is assumed specified. We present the following corollary, a direct result of Theorem 1, which explains the results in a more general context (similar ideas were presented in [123]):

Corollary 4.4.1. *Suppose (A1)-(A4) hold and $Q_\alpha \in \mathcal{N}_\Phi(\Omega)$. Suppose that $\hat{Q}^*_\alpha$ is estimated by*

$$\hat{Q}^*_\alpha(x) = \underset{f \in \mathcal{N}_{\Phi^*}(\Omega)}{\operatorname{argmin}} \frac{1}{n} \sum_{i=1}^{n} (\tilde{Q}_\alpha(x_i) - f(x_i))^2 + \lambda^2_{m,n} \|f\|^2_{\mathcal{N}_{\Phi^*}(\Omega)},$$

where $\Phi^* \leq c_6 \Phi$ for some $c_6 > 0$ and satisfies (A3) with a τ^*. If $\lambda^2_{m,n} \sim (mn)^{-2\tau^*/(2\tau^*+d)}$ as $m, n \to \infty$, the following results hold:

- *If $\tau^* = \tau$, then $\|\hat{Q}^*_\alpha(\cdot) - Q_\alpha(\cdot)\|_{L^2(\Omega)} = O_p((mn)^{-\tau/(2\tau+d)})$.*

- *If* $\tau^* < \tau$ *and* $c_4 m^{2\tau^*/d} \leq n \leq c_5 m^{\gamma}$, *then* $\|\hat{Q}^*_\alpha(\cdot) - Q_\alpha(\cdot)\|_{L^2(\Omega)} = O_p((mn)^{-\tau^*/(2\tau^*+d)})$.

This demonstrates that even if the covariance function is misspecified, we can achieve the optimal convergence if we have correctly estimated the general behavior of the covariance function, measured by τ. If we err on the conservative side and choose a covariance function with a small τ^*, e.g. the exponential covariance function, we sacrifice efficiency for robustness. Importantly, the result for $\tau^* > \tau$ is not included above. Although it is not shown in Theorem 1, this condition is likely to result in an inconsistent estimate.

4.5 Illustrations

Here, two examples are presented to illustrate the power of the proposed approach. The first deals with the crack propagation model discussed in Section 5.1. The second example provides a comparison between the proposed approach and the approach of [2] using the basic queueing system discussed in their work. Further details for implementation of the proposed method can be seen in the supplementary materials.

4.5.1 Material Fatigue

Fatigue of materials remains an important and challenging problem for engineers designing many structures from highways to turbine engines. Variability in loadings and material fatigue strength creates the need for a model that incorporates stochasticity. The study of fracture mechanics has recently focused on computational methods to understand propagation of faults in heterogenous materials. Examples include piezoelectric materials [75] and complex composites [50]; extensions to 3-dimensional fractures have further increased the complexity of computational models [61]. The inclusion of stochasticity in these models makes computational techniques burdensome for fine mesh models. Here, we study the simplified case of one-dimensional crack propagation under transverse cyclic loading.

A classic deterministic model for the crack growth in this setting is the Forman equation [40],

$$\frac{\mathrm{d}\ell}{\mathrm{d}t} = G(\ell) = f\frac{C_0(\Delta K(\ell))^n}{(1-R)K_c + \Delta K(\ell)}$$

where C_0 and n are constants, R is the ratio of the minimum (S_{min}) and maximum (S_{max}) pressures exerted on the material, f is the frequency of the cyclic loading, K_c is the fracture toughness, and $\Delta K(\ell) = (S_{max} - S_{min})\sqrt{\ell}\alpha(\ell)$. Here, $\alpha(\ell)$ is a function of the geometry and if the width of the structure is much larger than the crack size, $\alpha(\ell)$ can be approximated as 1. Though nonlinear, the Forman equation has the capability to represent both stable and accelerated growth rates [115]. An extension of the deterministic model to account for variation is achieved multiplying the above growth rate by a stochastic process with unit mean [76, 134, 113]. Specifically, the model considered is

$$\mathrm{d}\ell = G(\ell)\mathrm{d}t + G(\ell)\mathrm{d}W(t),$$

where $W(t)$ is a realization of a Wiener process with variance σ^2. The material properties and conditions used in this simulation are borrowed from [59], which studied a plate of 7075 aluminum alloy with an initial crack length of 2.54 mm. This experiment will emulate the crack length after 2000 cycles under various stress ratios (further details can be seen in the supplementary materials).

Figure 13 shows emulators created with varying n and m. Subplot (a) shows an emulator with a small value of both n and m, which is not a good estimator of the true simulation seen in (d). Subplots (b) and (c) represent the improvements that occur as we increase n and m respectively. From (a) to (b), n is increased and the shape of the quantiles as a function x is closer to the true shape of the quantiles shown in (d). From (b) to (c), m is increased allowing for better estimation of the individual quantiles (d) (compare the estimated quantiles at $R = 0$). For most inference, an emulator such as the one given in (c) will be sufficient for emulating the simulation.

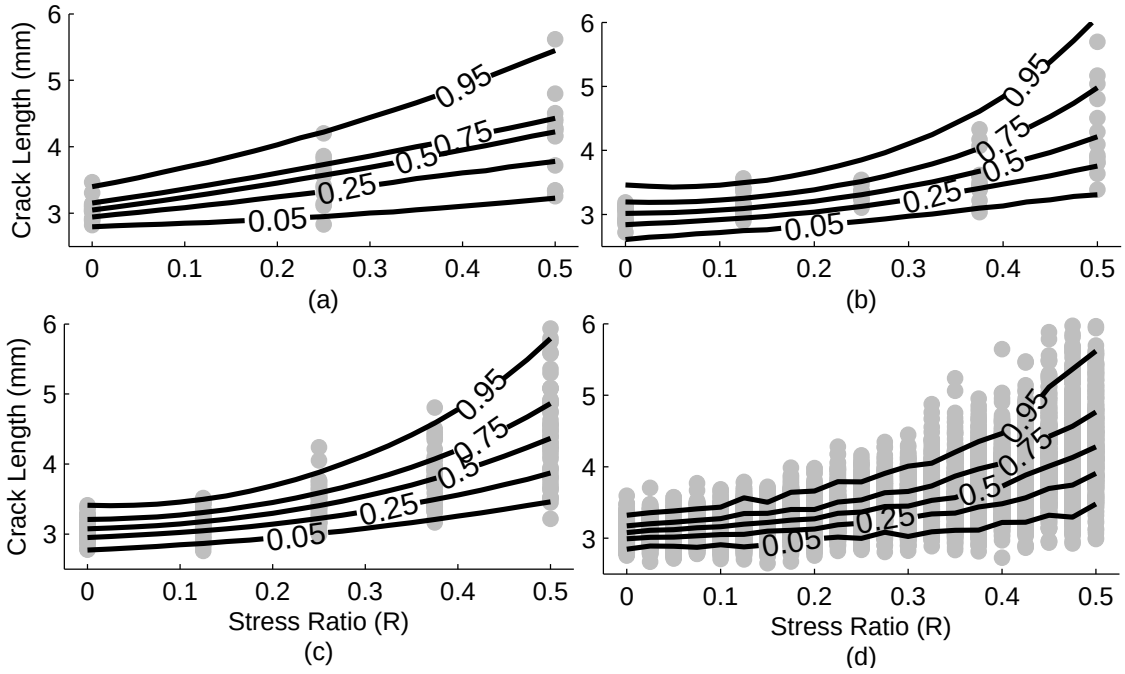

Figure 13: Example of emulation for Section 4.5.1; the light gray dots represent observations. Subplots (a), (b), and (c) contains the quantiles of the predictive distribution (solid line) with $n = 3$, $m = 15$ (a); $n = 5$, $m = 15$ (b); and $n = 5$, $m = 50$ (c). Subplot (d) represents empirical quantiles are generated by simulating 400 observations at 20 points, requiring $8,000$ samples.

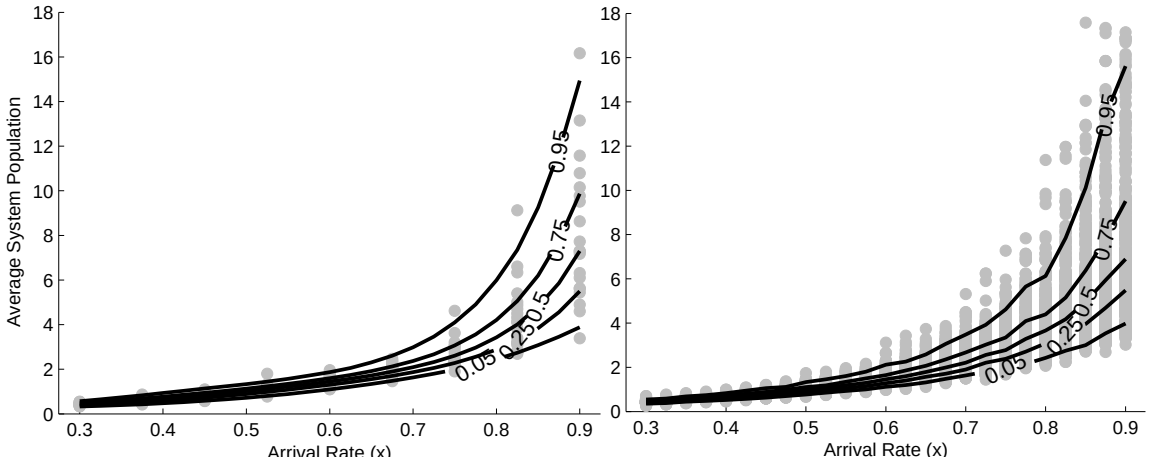

Figure 14: Example of emulation for Section 4.5.2; the light gray dots represent observations. The left hand plot contains the quantiles of the predictive distribution (solid line) with $n = 9$ and $m = 20$. The right hand empirical quantiles are generated by simulating 400 observations at 27 points, requiring $10,600$ samples.

4.5.2 Queueing System Example

We will now compare to [2] under their example, indicating the advantages of the proposed technique compared to the traditional metamodeling framework which assumes the simulation output follows a normal distribution. Here, we study a first-in first-out M/M/1 queue, i.e. a queue with one server and exponentially distributed interarrival (with mean x) and service times (with unit mean). The simulation output is the average system population in the system from time 0 to 1000. [2] defines the simulation output as a long term average, but this work considers this a finite horizon problem, which are commonly encountered (e.g. [33]). The experiment consists of n evenly spaced design points on $[.3, .9]$ with m replications.

Figure 14 compares the predictive density of the proposed method compared to estimates established through dense sampling. A closer inspection of the predictive density at three arrival rates can be seen in Figure 15, where the non-Gaussian behavior of the simulation is modeled significantly better by the proposed approach compared to the traditional metamodeling framework. This indicates the superiority

of the proposed method for creation of an accurate *emulator* for stochastic experiments.

We quantitatively compare predictive distributions using *integrated quadratic distance* (IQD), which is a proper divergence score given by

$$\int_{-\infty}^{\infty} \left(F(y) - G(y)\right)^2 \mathrm{d}y,$$

where F is the predictive distribution and G is the actual distribution. Under some regulatory conditions (see [121]), IQD scores are equivalent to the metric

$$\mathbb{E}|R - S| - \frac{1}{2}\mathbb{E}|R - R'| - \frac{1}{2}\mathbb{E}|S - S'|,$$

where R and R' are independent copies from F and S and S' are independent copies from G. A score of 0 indicates perfect emulation, and smaller values are preferred. Here, our goal is to develop a predictive distribution for a sample at a value x. Therefore, we create an average IQD (AIQD) by sampling a value of x from $[.3, .9]$. We create 400 replicates of the output at $[.25, .275, \ldots, .925, .95]$ to create estimates of $\mathbb{E}|R - S|$ and $\mathbb{E}|S - S'|$.

Table 2 presents a comparison of AIQD using the proposed framework. The comparison is made using differing levels of m and n, and a smaller value represents superior prediction. Since the simulation is stochastic, it is difficult to compare values directly across m and n, though in general, increasing m and n results in better prediction. Clearly, the proposed method outperforms the traditional metamodeling framework, which is at least partially caused by instability in estimating $\sigma^2(x)$ as mentioned in [2].

4.6 Concluding remarks

Here, a framework is established for building emulators of stochastic simulations via *quantile kriging*, which enables a computationally attractive alternative to running

Table 2: Performance of the proposed technique for the example in Section 4.5. ()'s designate the score using [2].

n	m	AIQD
5	10	**.0300** (.0850)
	20	**.0284** (.3600)
	40	**.0077** (.2397)
9	10	**.0221** (.0543)
	20	**.0258** (.1408)
	40	**.0081** (.1032)
17	10	**.0135** (.0504)
	20	**.0076** (.0467)
	40	**.0065** (.0470)

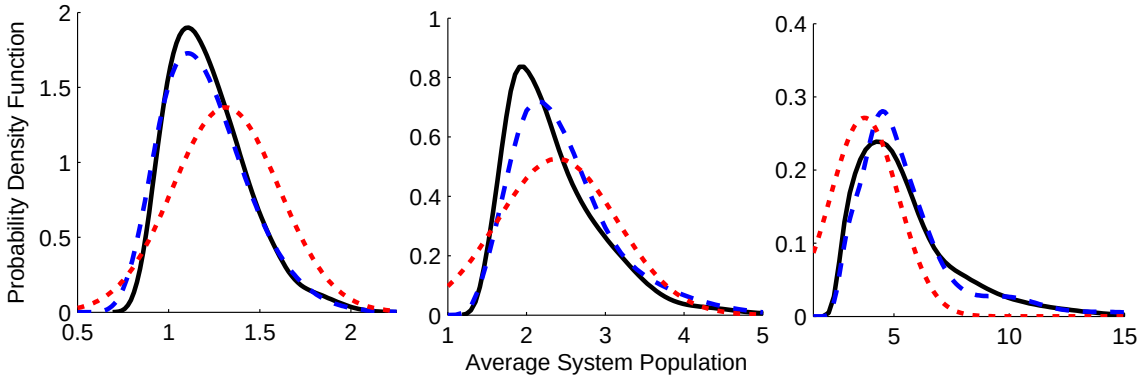

Figure 15: Predictive distributions of the average population in a queueing system over 1000 time units with an arrival rate of .55 (left), .70 (middle) and .85 (right) and $n = 9$ and $m = 40$. The proposed approach is marked by long dashes, and the solid line represents the distribution from 400 independent samples, and for comparison the method described in [2] is marked by shorter dashes.

the simulation model for every possible set of inputs. While emulators have been discussed to approximate specific models (e.g. [133]), this work discusses constructions that do not rely on knowledge of the structure of the simulation. Other methods have been proposed to predict distributions, see [27] and [35], but the focus in those works is on linear modeling, which can exclude large classes of responses. One key element of the proposed methodology is the characterization of both aleatoric and epistemic uncertainty, which differs from a large class of approaches known as quantile regression, which has been studied at great depth (for more information see the text [70]). The focus of these techniques is adjustment of the loss function from a squared error loss to a piecewise linear loss to find estimates of individual quantiles. Of the works in quantile regression, [74] is the closest to our work; the concept of their work is to add an additional penalty term representing the norm of a reproducing kernel Hilbert space, which is closely tied to Gaussian processes. Besides ignoring epistemic uncertainty, this method incurs significant computational cost as the method requires quadratic programming to find each quantile, which can be burdensome for large amounts of data. This difficulty is exacerbated when finding unknown parameters, which requires hundreds or thousands of quadratic programs. Another relevant technique by [38], often termed the "double kernel" approach, uses a similar two stage mechanism as quantile kriging, but uses procedures involving locally defined polynomials. We yield to the comments of [125] on the chapter [18] who explain the difference in the modeling strategies; our work uses models that are defined over the entire region in lieu of locally defined models.

However, the quantile kriging approach is not without drawbacks. The asymptotic efficiency is demonstrated, but small sample results could not be reached. Fully Bayesian approaches might provide comfort to a user concerned with small sample results, but the technique mentioned in this work purposely avoids the use of Dirichlet processes to insure simplicity of implementation through avoidance of complex

Markov Chain Monte Carlo algorithms. Also, while Sections 5.2 and 4.5 demonstrate some examples of implementation, lingering questions about the metamodel assumptions are inescapable. However, the continuity of the first moment, typically assumed by most metamodels, is similarly restrictive. Extensive simulation-based studies of the applicability of this model in a broader set of contexts are left for future work.

This work shows an asymptotic convergence rate of $O_p((nm)^{-\tau/(2\tau+d)})$, where τ is a measure of smoothness and d is the dimension of the input. This indicates that developing emulators will require a large sample size in high dimensional scenarios, which means inversion of a large $n \times n$ matrix. Since inversion is the major obstacle in practice, we focus on its computational cost. For the proposed method, the number of arithmetic operations grows according to $O(n^3)$, which is an improvement over the methods such as [74] which require $O(m^3 n^3)$ operations when using scattered data. However, problems still arise if n enlarges. Works studying similar problems for deterministic computer codes, e.g. [52] and [94], might provide some insight into solutions.

The choice of sets inputs and number of replications for the designed experiment is outside of the scope of this chapter, but challenges remain. In Section 4.4, we demonstrate an optimal rate of convergence by selecting the inputs via a uniform distribution and a symmetric number of replications. However, one would expect to achieve better small sample results using space-filling designs, such as those in [106], to select sets of inputs for the experiment. Additionally, [2] emphasize the allocation of more replications to sets of inputs that produce high variation in the output. A similar approach might provide benefits here as well, but while this approach is justified when predicting the simulation output mean with normally distributed variations, the more general approach taken in this work adds complications.

4.7 Details

4.7.1 Proof of Proposition 1

Proof. Denoting the norm of the difference in quantiles to be $\|Q_\alpha(x) - Q_\alpha(x_0)\|$, our goal is to show for all $\epsilon > 0$ there exists a ball $B(x)$, centered at x, for which $x \in B(x)$ implies $\|Q_\alpha(x) - Q_\alpha(x_0)\| \leq \epsilon$. Let the quantile function be written in terms of the inverse cumulative distribution $F_x^{-1}(\alpha) = \inf_{Y \in \mathbb{R}}\{F_x(Y) \geq \alpha\}$. Since F_x is continuous and strictly monotonic, $F_x^{-1}(F_x(Y)) = Y$. Therefore the norm becomes

$$\|F_x^{-1}(\alpha) - F_{x_0}^{-1}(\alpha)\| = \|F_x^{-1}(\alpha) - F_x^{-1}(F_x(F_{x_0}^{-1}(\alpha)))\|, \tag{23}$$

and since F_x is continuous and strictly monotonic with respect to Y, F_x^{-1} is continuous with respect to α. Therefore, there exists an $\epsilon' > 0$ such that

$$\|\alpha - F_x(F_{x_0}^{-1}(\alpha))\| < \epsilon'$$

implies (23). From this, we can rewrite the result as for all $\epsilon' > 0$ there exists a ball $B(x)$ such that for all $x \in B(x)$,

$$\|\alpha - F_x(Y_0)\| = \|F_{x_0}(Y_0) - F_x(Y_0)\| < \epsilon',$$

where $Y_0 = F_{x_0}^{-1}(\alpha)$. Since F_x is continuous with respect to x, we have established the existence of a ball $B(x)$.

4.7.2 Proof of Theorem 1

The proof of this result is available in the supplementary materials of [95].

4.7.3 Quantile Kriging Implementation Details for Section 5

The framework discussed in this chapter allows for a wide variety of choices to improve the performance in implementation. First, the simulation output for the queueing model is considered to be the logarithm of the average number of customers in the system; this resulted in significant gains in prediction and stability of metamodel

fit. The trend function μ, following [2], was estimated as a constant; the estimate used in this study was $\hat{\mu} = \sum_{i=1}^{m} \sum_{j=1}^{n} y_i(x_j)/mn$. We assume a Gaussian covariance structure $C(x, x') = \sigma^2 \exp(-\theta(x-x')^2)$, with θ found via the cross validation estimate discussed in section 3 and σ^2 set to the cross-validation estimate

$$\hat{\sigma}^2 = \frac{1}{mn} \sum_{i=1}^{m} \sum_{j=1}^{n} \left(\boldsymbol{\Sigma}^{-1}(\boldsymbol{\theta})\right)_{ii} e_{ij}^2,$$

which represents the ratio of e_{ij}^2 over the prediction normalized prediction variance (e_{ij} is defined in section 3). All computations are done using the numerical programming environment MATLAB.

Chapter V

DEFECT PATTERNS: ESTIMATION AND TESTING USING NONPARAMETRIC POISSON PROCESS MODELS

5.1 Introduction

This work is motivated by data collected during the production of steel rolled bars. A rolling mill is a production process used to control dimensions of a long workpiece through compressive forces applied by a set of rolls. The production of steel bars using a rolling mill represents one of the oldest manufacturing processes, but inspection of this system remains a challenging problem. Long tracks of rolled steel can develop defects throughout the manufacturing process. The extreme temperatures and hostile manufacturing environment make inspection of the mill in person difficult and dangerous. Recently, developments have been made that offer detection of surface level defects during production [73]. This technology employs specially designed cameras combined with novel algorithms to detect different types of defects including bleeds (when the molten interior core leaks through the hardened surface) and seams (fractures that are hardened further down the mill). While limited defects are acceptable, the reduction of defects in a rolled bar results in better tensile, compressive, and flexural strength. Ultimately, these features determine the application and usability of the item. Therefore, corrective action that results in reduction or prevention of defects yields significantly reduced material and energy waste.

Knowledge of the defect pattern for this system is critical for diagnosis of the underlying cause. For example, seam formation concentrated near the ends of a rolled bar can represent problems in the locking mechanisms used to steady the bar during downstream finishing. The knowledge of this eliminates the need to check

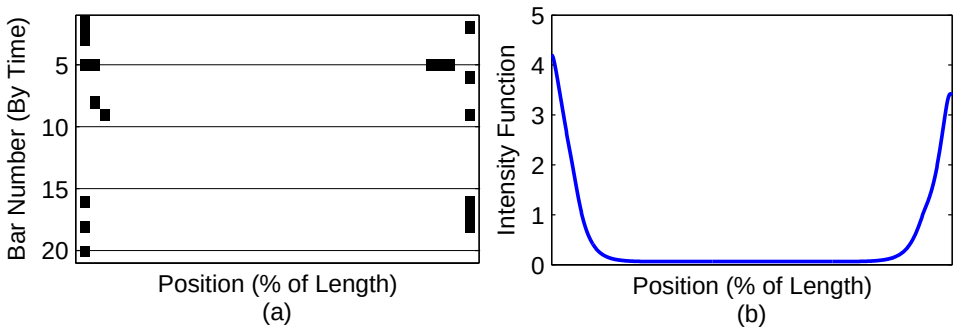

Figure 16: An example of defect locations (a) and the estimated intensity function using the proposed technique (b). For (a), a black box in an area indicates at least one defect.

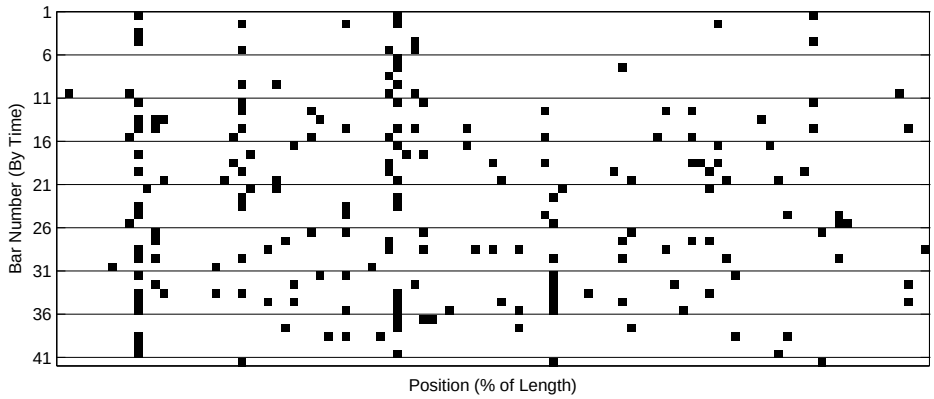

Figure 17: A subtle pattern of defects, a black box in an area indicates at least one defect.

individual rollers along the production line, thus saving hours of inspection and lost production time. Figure 16 illustrates an example of defects that are localized at the ends of the product. The estimated intensity function indicates the concentration of defects with sharp peaks at both ends. Figure 17 presents a more challenging example. A single rolled bar of approximately 60 meters has typically between 2-7 defects located throughout the bar, which makes determining a pattern of defects from a single bar extremely difficult. While a single bar does not exhibit an obvious pattern, the accumulation of several bars clearly indicates a pattern. In this example, there is an increase in defects near 10% and 40% of total bar length.

With this motivation in mind, section 2 describes the modeling philosophy with

an emphasis on the meaning of the stochastic point process modeling for defects. Sections 3 and 4 describe the functional estimation procedure based on penalized maximum likelihood estimation and the resulting hypothesis testing. Section 5 will derive the asymptotic rate of convergence with some discussion of the asymptotic efficiency. Illustrations in section 6 demonstrate the performance of the proposed estimate and hypothesis test and compare our approach to another common non-parametric estimate. Some conclusions and discussions of these types of problems are offered in section 7.

5.2 Stochastic Modeling

The measurements from the ith product is a series of m_i defects, labeled $x_{i1}, \ldots, x_{im_i}$, and n products are observed. Here, we model the set of defects from each product as an independent realization of a point process, which is a class of stochastic processes where samples consist of a set of isolated points. Statistical estimation of point process models have been extensively studied; applications include physics modeling [31], disaster occurrences [30], epidemiology [43], tomography [20] and crime events [81]. There are important distinctions between point–processes for temporal systems and our problem. In temporal problems, the common goal is to predict what will occur for a set number of epochs immediately succeeding the current one, i.e. outside of the observed space. However, due to the nature of our problem, this has no relevance; this would be akin to predicting the probability of defects not on the product. Instead, our primary interest is estimation of the current pattern of defects on a product, i.e. inside the observed space. A distinction must also be made between this problem and what are known as hazard models [80], where the data is often right–censored because of the ultimate outcome of these studies. Additionally, in hazard models, only a single point is typically observed per subject. In our problem, the right–censoring of the data does not occur and multiple defects can occur on a single product. The fact that

the scenarios and goals described in this work differ from the traditional examples illuminates why new methodology for estimation is needed.

For simplicity of exposition, this work will consider the *null* pattern to be a *no* pattern, i.e. an expectation of even defects throughout the product. We do this because (1) this objective is tasked to us in the described application and (2) extensions to other null patterns follow from this case. This work will probabilistically describe the existence of defect patterns by leveraging differences between what are known as *homogenous* processes and *inhomogenous* processes. A homogenous stochastic process assumption implies the probability of defects is similar no matter which section of the product is examined. In the context of our problem, this would be considered *no pattern*. Conversely, an inhomogeneous process would imply certain regions of the product have increased chances of defects, or there exists a *pattern* of defects.

In keeping with the proposed application, each product is assumed one dimensional and unit length, making the domain of the point process $[0, 1]$. To state our assumptions more exactly, let the point process be defined by a random function g that maps subsets of $[0, 1]$ to the number of defects in that section. Here, we employ Poisson processes (see [65] for more details), which are defined by the following two properties based on regions of the product, $X \in [0, 1]$:

- $N(X_1), \ldots, N(X_m)$ are independent when $X_1, \ldots, X_m$ are disjoint.
- $N(X)$ has a Poisson distribution with parameter $\int_X f(x)\mathrm{d}x$, where $f(x)$ is a non-negative function.

The function f is termed an *intensity function*, which indicates the rate at which faults occur in a given region. In the absence of a pattern of defects, each product can be assumed to follow a homogenous random process, which is defined as for any subset of the product, X, the distribution is identical to the distribution of the shifted process. This implies $f(x) = \alpha$, $\alpha > 0$.

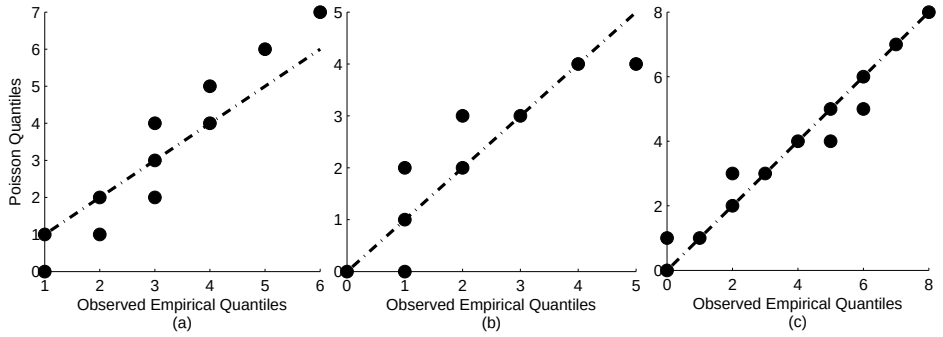

Figure 18: Observed empirical quantiles of the number of defects over the regions $[0, .5]$ (a), $[.375, .625]$ (b), and $[.25, 1]$ (c) versus the quantiles from the Poisson distribution.

The standard model outlined above has tremendous flexibility, but might not always be applicable. There are extensions of this model, termed Cox models (see [65]), where the underlying intensity function, $f(x)$, is itself random. However, in our example, there are not enough observations on a single product to either confirm or deny that assumption, and we error on the side of simplicity by using the above model. Furthermore, the distributional assumptions imposed by the Poisson process model needs to be investigated. Figure 18 demonstrates the Poisson quantile-quantile plot of the number of defects over regions of various sizes for the products in Figure 17. The linear nature of the observations on the plot indicates the distribution of the number of defects in each region is at least approximately Poisson distributed.

5.3 Penalized Maximum Likelihood Estimation of f

This section will describe the general strategy for estimation of the intensity function f. Our estimation scheme is designed to produce an interpretable and stable estimate that is computationally tractable.

Section 5.3.1 will motivate the discussion of our ideas by describing previous attempts at similar problems and explain some deficiencies of the most common approaches for estimating these functions. Section 5.3.2 will describe the proposed approach. Our estimate can be found via a Newton-Raphson solver, detailed in appendix 5.8, which allows for fast estimation. Section 5.3.3 provides insight into practical matters, such as parameter estimation and choice of kernel functions.

5.3.1 Review of Estimation Procedures

Based on figures 16 and 17, the use of a parametric model for $f(x)$ would result in inadequate estimators. As the number of observed products grows large, a parametric framework would miss several potential patterns present in the data. Currently, the most prevalent strategy for nonparametric estimation of the intensity function is termed a local kernel smoothed estimate [98, 29], which is analogous to the local smoothed kernel density estimates [104], and the use of these methods persist today [23, 82, 122, 51, 81]. However, as will be shown in comparisons later, this approach leads to undesirable properties in the estimate. This is because convergence of the estimate relies on shrinkage of the lengthscale of the basis functions, which will naturally cause sharp spikes and dips in the estimated function.

Here, we base our estimation procedures on maximizing the likelihood. This is not a new concept, but few works have reported usable estimation procedures in the general nonparametric setting for these types of point processes. Several works that have studied methods to estimate the underlying intensity function by thresholding wavelet coefficients from a single realization of the process [71, 128, 101], whereas this work studies the case of multiple realizations. [13] studied this thresholding approach when several shifted Poisson processes are observed, a case that differs from the one considered here. Also of note are the works on hazard models, e.g. [3], [139] and [91]. The differences between hazard models and the models studied here were discussed

in Section 5.2. The works of [64] and [109] contain similar ideas to the one presented here, though they differ in key parts such as function space assumptions and penalization choices. More importantly, the lack of tractable computational methods with previous estimates seems to have limited the impact of previous penalized approaches, whereas the estimate to be proposed can be found with a broad array of convex optimization methods.

5.3.2 Proposed Estimation Technique

Consider nonparametric setting where $f(x)$ is in a general function space, $\mathcal{F}$. A simple approach would produce an estimate of f, termed $\hat{f}(x)$, by directly maximizing the log–likelihood,

$$L(f) = \sum_{i=1}^{n} \sum_{j=1}^{m_i} \log\{f(x_{ij})\} - n \int_0^1 f(x) \mathrm{d}x.$$

However, if one maximizes L in a general function space, the resultant estimate will be a function that has extreme spikes at x_{ij} and is as close to 0 everywhere else. For example, $\hat{f}(x) = \sum_{i=1}^{n} \sum_{j=1}^{m_i} \delta(x - x_{ij})$ would maximize the log–likelihood in a space that included translations of δ. This is a function composed only of infinite spikes at the observed defects. From a practical point of view, this does not represent any additional information gained through estimation. From a stochastic point of view, this estimate is extremely unstable, with infinitely large variances.

There are a bevy of choices available for $\mathcal{F}$ and here we consider the functions to come from a general nonparametric family termed reproducing kernel Hilbert spaces (RKHS) [124, 127],

$$\mathcal{G} = \left\{ g(\cdot) = \sum_{i=1}^{\infty} \beta_i \kappa(\cdot, z_i) \middle| \beta_i \in \mathbb{R}, z_i \in [0,1], \|g\| < \infty \right\},$$

where $\kappa(x, y)$ is a positive (semi-)definite function termed a kernel and

$$\|g\| = \sum_{i=1}^{\infty} \sum_{j=1}^{\infty} \beta_i \beta_j \kappa(z_i, z_j)$$

represents the complexity of the function. This function space contains large classes of functions where the complexity (or specifically $\|g\|$) is restricted. For example, the L_2 is not a RKHS because the associated reproducing kernel would be the δ function, but this does not have a bounded L_2 norm. However, without modification of $L(f)$, the maximizer of $L(f)$ can be made arbitrarily close to the function $\sum_{i=1}^{n}\sum_{j=1}^{m_i}\delta(x-x_{ij})$. The estimate proposed by this work will be a *penalized* version of the likelihood and is given by

$$\underset{f\in\mathcal{F}}{argmax}\, L(f) - \lambda_n J(f),$$

where $J(f)$ is a complexity penalty and λ_n is a parameter depending on the number of observed products.

Since the function f must be positive, we define the space of possible functions as

$$\mathcal{F} = \left\{ g^2 \,|g \in \mathcal{G} \right\}.$$

The practice of squaring the function emulates similar methods used for penalized estimates for density estimation [129], and the theoretical ramifications from this will be discussed in detail in Section 5.5.

Our goal is now to find the maximizer of L in terms of the function g,

$$\hat{f}^{1/2} = \underset{g\in\mathcal{G}}{argmax}\, L(g^2) - \lambda_n\|g\|,$$

Maximizing the above under the proposed function assumptions proves difficult since there are an infinite number of basis functions. Typically, a representation theorem is employed [124] to avoid this very dilemma. The generalized representation theorem shown in [108] implies the following result:

Theorem 5.3.1. *For any function c that maps from $[0,1]^n$ to $\mathbb{R}\cup\{-\infty\}$,*

$$\hat{g} = \underset{g\in\mathcal{G}}{argmax}\, c(g(z_1),\ldots,g(z_n)) - \lambda_n\|g\|,$$

admits representation of the form $g(\cdot) = \sum_{i=1}^{n}\beta_i\kappa(\cdot,z_i)$.

However, in our formulation the objective function $L(g^2)$ is not of the form used in the representation theorem; $\int_0^1 g^2(x)\mathrm{d}x$ depends on the entire function. We propose approximating this value by $\int_0^1 f(x)\mathrm{d}x \approx n_q^{-1}\sum_{i=1}^{n_q} f(\tilde{x}_i)$, where the points $\tilde{x}_1, \ldots, \tilde{x}_{n_q}$ are determined by the user. Section 5.5 discusses sampling these points from $U[0,1]$, but any approach that represents a quadrature rule, such as equispaced points, will likely suffice in the majority of cases. Denote $\tilde{L}(g)$ as

$$\tilde{L}(g) = 2\sum_{i=1}^{n}\sum_{j=1}^{m_i}\log\{g(x_{ij})\} - nn_q^{-1}\sum_{i=1}^{n_q} g^2(\tilde{x}_i) - \lambda_n\|g\|.$$

Now, the representation theorem can be invoked, which results in the estimate

$$\hat{f}^{1/2} = \underset{g\in\mathcal{H}}{argmax}\,\tilde{L}(g) - \lambda_n\|g\|, \tag{24}$$

where H is a function space with *finite* parameters

$$\mathcal{H} = \left\{ h \,\middle|\, h = \sum_{i=1}^{n}\sum_{j=1}^{m_i}\beta_{ij}\kappa(\cdot, x_{ij}) + \sum_{i=1}^{n_q}\gamma_i\kappa(\cdot,\tilde{x}_i) \right\}.$$

We demonstrate in appendix 5.8 that this is a *convex* problem and therefore its optimization is possible through a variety of methods. Appendix 5.8 goes on to describe Newton-Raphson methods. The described methods worked very quickly in our examples, typically requiring fewer than 10 iterations to reach convergence with an error less than 10^{-2}.

While the above is computationally feasible, appropriate questions arise about the impact of maximizing the above approximate log–likelihood versus the true likelihood, $L(f)$. Similar ideas to this approximation were presented in [11]. In Section 5.5, we study the asymptotic properties to demonstrate that if the difference between the $\int_0^1 f(x)\mathrm{d}x$ and $n_q^{-1}\sum_{i=1}^{n_q} f(\tilde{x}_i)$ is sufficiently small, no loss to rate of convergence is incurred.

So far, this section has explained the general framework, but a practical matter persists. We ultimately would like to test for the presence of a pattern, where no pattern is described as $f(x) = \alpha$, $\alpha > 0$. Therefore, in the presence of very little

or no data, a good estimator would regress to a constant but not necessarily 0. We achieve this effect through a small modification of the function space to allow for a non-zero nullspace [124]. In the context of the above estimation procedures, the nullspace is the value of $\hat{f}(x)$ when $\lambda_n = \infty$. The modified function space is then

$$\mathcal{F} = \left\{ (g + \alpha)^2 \, | g \in \mathcal{G}, \alpha \in \mathbb{R} \right\},$$

and the norm, or penalty, $\|g\|$ is unchanged. The methods described above are not affected by the chosen nullspace, the value of $\mathcal{H}$ in (24) is changed to

$$\mathcal{H} = \left\{ h \,\middle|\, h = \alpha + \sum_{i=1}^{n} \sum_{j=1}^{m_i} \beta_{ij} \kappa(\cdot, x_{ij}) + \sum_{i=1}^{n_q} \gamma_i \kappa(\cdot, \tilde{x}_i), \alpha \in \mathbb{R} \right\}.$$

5.3.3 Choice of Kernel Function and Smoothing Parameters

These types of estimation procedures typically work by exploiting the continuity, or higher orders of differentiability, of the underlying intensity function. This property can be described colloquially as if two points on a product are sufficiently close, the chance of a defect is similar. However, the use of this model requires no higher orders of differentiability beyond continuity, i.e. it does not require prefect smoothness.

In an idealized case, the underlying basis functions κ would perfectly represent the underlying function, but this is unachievable in reality. This section discusses the choice and influence of κ. For simplicity, we discuss here only symmetric and stationary kernel functions as these are the most commonly used. These are functions of the form $\kappa(|x - x'|/\theta)$, where the value of $\theta > 0$ represents a lengthscale parameter. If the kernel function decreases from the origin, increasing θ results in an estimated function with fewer ridges and bumps, see Figure 19.

The order of differentiability assumed on the underlying function f is dictated by the kernel function $\kappa(x, x')$. For example, the stationary exponential kernel, $\kappa(x, x') = \exp(-|x - x'|/\theta)$, assumes the patterns are differentiable nowhere. As another example, the Gaussian kernel, $\kappa(x, x') = \exp(-|x - x'|^2/\theta^2)$, assumes that

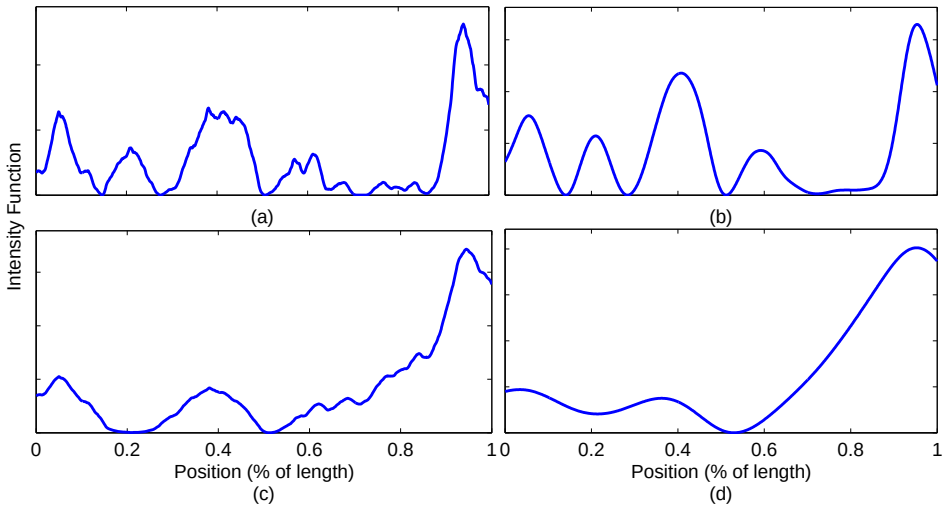

Figure 19: Examples of intensity functions generated by the Matérn kernel with $[\theta, \nu] = [.01, .5]$ (a), $[.01, 2]$ (b), $[.1, .5]$ (c), $[.1, 2]$ (d).

the function has all orders of differentiability everywhere. Here, we borrow from geostatistics (see, e.g. [44]) and advocate for the use of the Matérn kernel function,

$$\kappa(x, x') = \frac{1}{2^{\nu-1}\Gamma(\nu)} \left(2\sqrt{\nu}h/\theta\right)^{\nu} \mathcal{K}_{\nu} \left(2\sqrt{\nu}h/\theta\right),$$

where $\mathcal{K}_{\nu}$ is the modified Bessel function of order ν and $h = |x - x'|$. The use of this function allows a user to independently adjust a smoothness parameter ν, where intensity function is subsequently assumed $\lfloor \nu \rfloor$ times differentiable [53]. This function can also be considered a balance between two extremes, the exponential kernel occurs when $\nu = .5$ and the Gaussian kernel results as $\nu \to \infty$. Figure 19 shows examples of functions generated using different lengthscales and smoothness parameters.

By using a broad kernel class such as the Matérn, typically there is a combination of parameters that are reflective of the underlying function. We propose estimating these parameters from the data via a cross-validation criterion. We employ the log–likelihood function L and judge the parameters based on a *leave–one–out log–score*, given by

$$\sum_{i=1}^{n} \sum_{j=1}^{m_i} \log(\hat{f}_{-i}(x_{ij})) - n \int_0^1 \hat{f}(x) \mathrm{d}x,$$

where $\hat{f}_{-i}(x)$ represents the estimated function from (24) after removing the i^{th} product observation. A larger value of the log-score indicates better prediction, see [46] for more information on the log-score and other criteria to measure prediction accuracy. This criterion takes advantage of the multiple, independent observations of the process. Other criteria for estimating parameters of intensity functions were designed in the absence of this information [30]. A parameter not discussed in this section is λ_n, which can be estimated in the manner above, but better results were found by using a set value that decreases at a rate according to the results in section 5.5.

5.4 Hypothesis Testing

As mentioned in the introduction, one of the major goals of this work is to provide a user with a certificate of the existence of a pattern. The likelihood ratio test, one of the most celebrated methods in statistics, has proved highly effective in a broad array of problems. But the question is how to use it in this circumstance. A straightforward implementation would yield a test statistic of the form

$$T = \sup_{f \in \mathcal{F}} \tilde{L}(f) - \sup_{f \equiv \alpha, \alpha \in \mathbb{R}} \tilde{L}(\alpha),$$

where we are testing whether there is a statistically significant deviation from the null pattern, where the null pattern corresponds to equally likely defects across the product, or $f(x) = \alpha$. If T is significantly large, i.e. above some threshold T_0, we can reject the null hypothesis that $f(x) = \alpha$.

However, as with estimation, the nonparametric setting makes the straightforward approach an unstable procedure. The first term, $\sup_{f \in \mathcal{F}} L(f)$, will get arbitrarily large as f approaches $\sum_{i=1}^{n} \sum_{j=1}^{m_i} \delta(x - x_{ij})$ while the second term $\sup_{f \equiv \alpha, \alpha \in \mathbb{R}} L(\alpha)$ is be bounded by $L(f(x) = m/n)$, where $m = \sum_{i=1}^{n} m_i$. Therefore, no matter what threshold T_0 we set, the value T listed above will always exceed it, resulting in type I errors (false positives) of 100%. As a result, any statistical process control based on the above test statistic would *always* raise an alarm during production. This is

of course undesirable, so once again we employ a penalized approach. [39] discuss a similar method in a generalized setting with nonconcave likelihoods.

To achieve tractable computation, we utilize the approximate likelihood described in the previous section, $\tilde{L}(f) = \sum_{i=1}^{n} \sum_{j=1}^{m_i} \log\{f(x_{ij})\} - nn_q^{-1} \sum_{i=1}^{n_q} f(\tilde{x}_i)$. The proposed test statistic is

$$T = \sup_{f \in \mathcal{F}} \left[\tilde{L}(f) - \lambda_n J(f) \right] - \sup_{f \equiv \alpha, \alpha \in \mathbb{R}} \left[\tilde{L}(f) - \lambda_n J(f) \right].$$

The first term can be evaluated using the estimate from (24). The value

$$\sup_{f \equiv \alpha, \alpha \in \mathbb{R}} \left[\tilde{L}(f) - \lambda J(f) \right]$$

can be explicitly found as using $f(x) = m/n$ from the definition of $\tilde{L}$ and J.

The use of this test statistic for hypothesis testing requires the distribution under the null to be known. While [39] discuss the asymptotic distribution of such an estimator, this approach proved ineffective for our case where sample sizes are limited. To achieve better small sample results (the test we discuss in Section 5.6 has only 5 samples), we opt for a simulation based method approximation to the null where we simulate values of T from homogenous Poisson process with $f(x) = m/n$.

One might question the efficacy of this brand of hypothesis testing. Does the penalization method hurt the traditional likelihood ratio test? The answer lies in the distribution of T, where the use of penalization stabilizes the distribution under the null. Compare this to the case where $\lambda_n = 0$, where we demonstrated earlier in this section that the test statistic is unbounded. This is a direct analog of the traditional bias versus variance tradeoff of most high dimensional statistical problems. Our test statistic is biased toward zero, but the variance is controlled.

5.5 Asymptotic Analysis

While the next section will outline an example of the practical benefits of the proposed methodology, this section will show the proposed method is asymptotically consistent

and discuss the efficiency of the proposed approach. In our industrial problem , sample size restrictions prevent the asymptotic results from being utilized directly, but the consistency of the prediction is critical towards gauging the performance of the proposed framework. An inconsistent procedure would demonstrate that no matter how many products we observe, the resultant conclusions would not improve.

Here, we assume the underlying kernel function is stationary, symmetric and monotonically decreasing. We demonstrate that the estimated function, $\hat{f}$ from (24), approaches the true underlying function, denoted f_0, in terms of the distance $\rho(f, f_0) = \|f^{1/2} - f_0^{1/2}\|_2$, where $\|\|_2$ is the L_2 norm. If both f and f_0 were densities, this would be analogous to the Hellinger distance. The similarities between this distance measure and one used for densities is not coincidental; our approach to demonstrating consistency is analogous to similar proofs for density estimation. Specifically, we rely heavily on the large deviation inequalities of [129].

We demonstrate the result under the following regularity conditions:

(A) The function f_0 is bounded by some known constant $c_1 > 0$.

(B) There exist a constants τ with $\lfloor \tau \rfloor \geq 1$ and $c_2 > 0$ such that the Fourier transform of $\kappa(|h|)$ is less than $c_2(1 + |w|^2)^{-\tau}$ for $w \in \mathbb{R}$.

Condition (A) is verified in terms of our problem by insuring the product does not have any point which *always* has a defect. Figure 17 demonstrates that this is not the case in our example. The constant must be known to allow for the Monte Carlo sampling scheme described later in this section to achieve convergence, but c_1 can be chosen arbitrary large. Condition (B) is more difficult to verify in general, but when using the Matérn kernel function from Section 5.3.3, this condition can be verified with $\tau = \nu + 1/2$ [114]. Therefore, so long as $\nu \geq 1/2$, where $\nu = 1/2$ corresponds to the exponential kernel, our theorem holds. Condition (B) can therefore be interpreted as follows: *the underlying function, $f_0(x)$, exhibits reasonable smoothness.*

The next theorem formally states the core asymptotic result (proof is located in appendix 5.9):

Theorem 5.5.1. *Suppose (A) and (B) are met. Suppose also that* $\max(\lambda_n J(f_0), \lambda_n) \leq c_3\varepsilon$ *for a constant* $0 < c_3 < 1/2c_1$. *There exist strictly positive constants* c_4 *and* c_5 *such that if* $\lambda_n \sim n^{-2\tau/(2\tau+1)}$ *as* $n \to \infty$, *then*

$$P^*\left(\sup_{\rho(f,f_0)<\varepsilon, f\in\mathcal{F}} [L(f) - \lambda_n J(f)] - [L(f_0) + \lambda_n J(f_0)] \geq -c_4 n\varepsilon^2\right) \leq 7\exp(-c_5 n\varepsilon^2),$$

where P^* *is the outer probability measure.*

This result demonstrates conditions for which the probability of an errant estimate becomes exponentially small when using $L(f)$ directly. This differs slightly from the proposed estimate in (24) which is based on the approximate likelihood $\tilde{L}(f)$.

The difference between $\tilde{L}(f)$ and $L(f)$ is dictated by the closeness of $n_q^{-1}\sum_{i=1}^{n_q} g^2(\tilde{x}_i)$ to $\int_0^1 g^2(x)\mathrm{d}x$, where $n_q^{-1}\sum_{i=1}^{n_q} g^2(\tilde{x}_i)$ is often termed a quadrature rule. The choice a quadrature rule is open, but here we offer a specific suggestion where $\tilde{x}_1, \ldots, \tilde{x}_{n_q}$ are drawn from $U([0,1])$, i.e. a Monte Carlo sample. Using Monte Carlo sampling does not require a continuous derivative to converge, therefore with an appropriately large n_q we can reach an approximation even when the function is nondifferentiable. Because of assumption (A), the functions in $\mathcal{F}$ are bounded above and below. Therefore, by Hoeffding's inequality, there exists a constant c_6 such that for all $f \in \mathcal{F}$,

$$P\left(\left|n_q^{-1}\sum_{i=1}^{n_q} f(\tilde{x}_i) - \int f(z)\mathrm{d}z\right| \geq \varepsilon\right) \leq 2\exp(-c_6 n_q^2\varepsilon^2).$$

With this, we can demonstrate the convergence of the estimate $\hat{f}(x)$.

Corollary 5.5.1. *Suppose (A) and (B). Suppose also we draw* $\tilde{x}_1, \ldots, \tilde{x}_{n_q}$ *from* $U([0,1])$ *and we choose* $\lambda_n \sim n^{-2\tau/(2\tau+1)}$ *and* $n_q \sim n$. *Then*

$$\|\hat{f}^{1/2} - f_0^{1/2}\|_2 = \mathcal{O}_p\left(n^{-\tau/(2\tau+1)}\right),$$

where $\hat{f}$ *is defined in (24).*

With this, we demonstrate that $\hat{f}$ will get close to f_0 as measured by the distance metric ρ. When discussing a similar problem for densities, [110] demonstrated that this represents the best possible convergence, in terms of rate, for *any* estimate satisfying the assumptions similar to (A) and (B).

The optimality of the predictor listed above depends on the knowledge of the kernel function. However, in practice the kernel function is not explicitly known and is estimated as in Section 5.3.3. We present the following corollary which explains the result in a more general context:

Corollary 5.5.2. *Suppose (A) and (B). Suppose also we draw* $\tilde{x}_1, \ldots, \tilde{x}_{n_q}$ *from* $U([0,1])$ *and* $n_q \sim n$. *Further suppose that* $\hat{f}$, *defined in (24), is estimated using a kernel function,* $\kappa^*(h)$, *that satisfies* $\kappa^*(h) \leq c_7\kappa(h)$ *and (B) with* $\tau^* \leq \tau$. *If* $\lambda_n \sim n^{-2\tau^*/(2\tau^*+1)}$ *then* $\|\hat{f}^{1/2} - f_0^{1/2}\|_2 = O_p\left(n^{-\tau^*/(2\tau^*+1)}\right)$.

To show this, we only need to use the embedding theorem of [5], which says that under the above conditions f_0 is in the RKHS generated by κ^*, therefore we employ corollary 1 with $\tau = \tau^*$. This demonstrates that even if the kernel function is unknown, the correct behavior of the Fourier transform of the kernel is all that is needed to achieve an optimal rate of convergence. Importantly, the result for $\tau^* > \tau$ is not included above. While not yet shown, the authors suspect this condition could result in an inconsistent estimate. Therefore, if τ^* is chosen to be small, e.g. the exponential kernel function, we sacrifice efficiency for robustness.

5.6 Illustrations

This section will demonstrate the power of the proposed approach with data from the steel rolling mill described earlier. Simulation studies are avoided to allow for comparison of techniques in the context of the current problem.

5.6.1 Function Estimation

We discuss here the estimation of the intensity function with a comparison of the proposed method to a widespread nonparametric technique. As discussed earlier in this work, the popular technique for estimating intensity functions is local kernel smoothing [29], which we will refer to as *local smoothing* in this section to prevent confusion. Like the proposed estimator, local smoothing is a nonparametric approach to finding the intensity function. However, unlike the proposed approach, it contains no penalty for complicated functions, which means the estimate can be extremely complex as the number of observations increases. Another major difference is the basis functions. While the proposed estimate seeks to achieve basis functions with a lengthscale that represents the underlying intensity function, as discussed in Section 5.3.3, the local smoothing philosophy requires shrinking lengthscales to approximate the function. This leads to peculiarities in the estimated function. Furthermore, these problems compound and propagate as we observe more products.

In this section, we compare the two techniques using the data from Figure 17. For the proposed method, we use the Matérn kernel with $\nu = 1/2$. Keeping with the asymptotic results, we choose $\lambda_n = 2n^{-2/3}$ and for both methods the lengthscale parameter, θ, is selected via the cross-validation approach discussed in Section 5.3.3. We compare to the local smoothing technique with Gaussian basis functions. Modification to all described values were attempted, but no different conclusions were reached.

Different estimators for $f(x)$ from the data in Figure 17 can be seen in Figure 20. While both methods have clear peaks where expected, the local smoothing technique results in a more complex estimate. However, there does not appear to be enough data in Figure 17 to possibly estimate such a complex function with accuracy. This conclusion will be further validated by comparing the proposed hypothesis test with a test generated by local smoothed approaches.

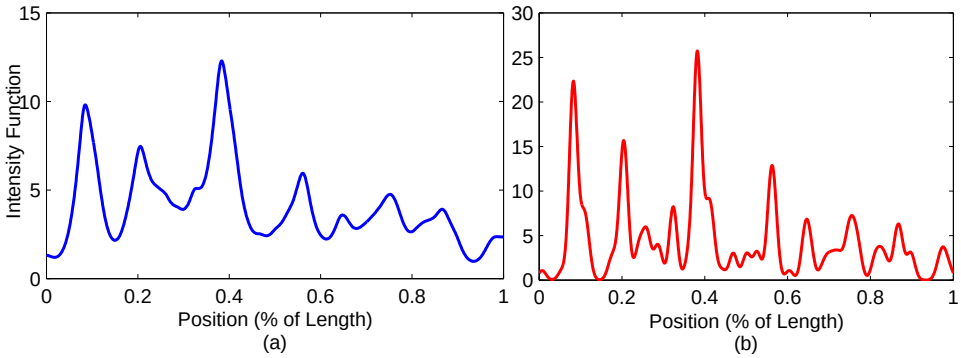

Figure 20: Diagram of predicted intensity function from the data in Figure 17 using the proposed penalized approach (a) and the local smoothing approach (b). Parameters for both estimation procedures were decided by leave–one–out log–score.

5.6.2 Hypothesis Testing

Here we study the hypothesis testing mentioned in Section 5.4. Unlike the estimation problem, there has been limited number of studies on hypothesis testing of the type proposed here (though [32] investigates a two sample problem). Therefore, we will compare to a representative technique that is easily derived from local smoothing, which compares likelihoods, but does not utilize the appropriate supremum. This is philosophically similar to the approach taken in [138]. Denote the local smoothed estimate as $\hat{f}_{LS}$, the local smoothed test statistic is given by

$$T_{LS} = L(\hat{f}_{LS}) - L(f(x) = m/n).$$

For similarity to the proposed approach, the null distribution of T_{LS} is estimated by simulation from a process with $f(x) = m/n$.

We will compare the power of the proposed hypothesis tests by looking at all groups of five consecutive products from Figure 17. By visual inspection and conversations with the provider of the data, this group of products contains a pattern. Therefore, a good statistical test should reject the null. And by using only five bars, we can investigate how the proposed approach would perform in setups like those seen in statistical process control. Here, the *statistical power* is measured by the

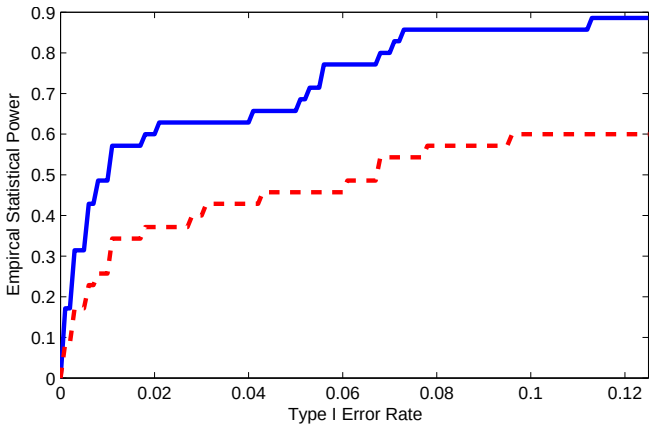

Figure 21: Empirical statistical power of the proposed penalized approach (solid) and the local smoothing approach (dashed) for testing if a pattern exists for blocks of five consecutive of bars from Figure 17.

probability of rejecting the null under a type I error rate α (false alarm). The type I error rate is estimated by simulating from $f(x) = m/n$. Figure 21 shows that by using only five bars, the proposed approach can detect faults with over 30% more power than a similar approach using local smoothing.

This significant improvement over the local smoothing technique can be attributed to variance stabilizing properties of penalized estimation. Extrapolating slightly beyond this individual case study, these results imply that for sparsely observed defects, the penalization framework has significant advantages over local smoothing. However, there is a computational cost to consider; the proposed estimate takes longer to compute. This did not become an impediment in our application. In our largest data set, Figure 17, the estimate with parameter tuning is received in under 30 seconds (250 parameters, $\approx$ 200 defects and 50 quadrature points). With fixed parameters, the estimate is received in .27 seconds.

5.7 Conclusion and Discussion

This work studies the modeling of the spatial pattern of defects across several products. For production systems, the majority of statistical research has focused on cases

in which we observe a set number of statistics per output (i.e. a scalar, vector or matrix of outputs). Monitoring schemes have been designed to tackle multidimensional outputs when the number of the outputs is large, e.g. [68], [137], and [97]. Here, we observe a set of defect locations, where the total number of defects varies from product to product. Currently, it is not clear what relationship, if any, exists between the two cases. One idea to link these frameworks could be done by partitioning the bar as in figures 16 and 17. This creates the mapping from the current data to a vector of 0's and 1's, similar to the approach discussed in [54]. However, transformation destroys some information and curbs the ability to create spatial inference on the continuous product.

The methodology for hypothesis testing appears promising. While not explicitly outlined in this work, the creation of a monitoring scheme to detect a pattern during production could be done by sequentially testing for a pattern in the last k products. This is equivalent to an $\bar{X}$ test, which is known to be less effective at detecting slight patterns. Modified monitoring schemes using techniques such as CUSUM and EWMA would likely provide tremendous benefits, but how to incorporate these ideas into the framework outlined here is unclear. Ideas proposed by [136] in a differing problem might provide insights.

5.8 Optimization Problem for Estimation

There are two major points demonstrated in this section. We show the convexity of the optimization problem, which allows us to use a variety of optimization methods. Then, a standard Newton-Raphson method is described that works well for the cases illustrated in this work.

To begin, we find the first and second derivatives with respect to the parameters $\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$. For simplicity of exposition, we merge all parameters into a single vector $\boldsymbol{\delta}$. By the definition of $\boldsymbol{\delta}$, each element corresponds to either an observed defect, x_{ij},

or a point $\tilde{x}_i$. The elements of $\boldsymbol{\delta}$ are ordered such that elements $1, \ldots, m$ correspond to observed points and elements $m+1, \ldots, n_q$ correspond to the points $\tilde{x}_i$. Let $\boldsymbol{K}$ be an $(m+n_q) \times (m+n_q)$ kernel matrix that corresponds to all points associated with $\boldsymbol{\delta}$. We denote submatrices that correspond to indices $i_1, \ldots, i_2$ and $j_1, \ldots, j_2$ of this matrix by $K_{i_1:i_2, j_1:j_2}$. We denote the current root-intensity function, $g(x)$, at all points associated with $\boldsymbol{\delta}$ as $\boldsymbol{g}(\boldsymbol{\delta}) = [g(x_{1,1}), \ldots, g(x_{n,m_n}), g(\tilde{x}_1), \ldots, g(\tilde{x}_{n_q})]^\mathsf{T}$ and the elementwise inverse as $\boldsymbol{g}^{-1}(\boldsymbol{\delta})$, and it depends on current value of $\boldsymbol{\delta}$.

The vector of partial derivatives of $-L(\boldsymbol{\delta})$ is

$$\boldsymbol{g}_{\delta_0} = -\frac{\partial L}{\partial \boldsymbol{\delta}}\bigg|_{\boldsymbol{\delta}_0} = -2\boldsymbol{K}^\mathsf{T}_{1:m,\cdot}\boldsymbol{g}^{-1}_{1:m}(\boldsymbol{\delta}_0) + 2\frac{n}{n_q}\boldsymbol{K}^\mathsf{T}_{m+1:m+n_q,\bullet}\boldsymbol{g}_{m+1:m+n_q}(\boldsymbol{\delta}_0) + 2\lambda_n \boldsymbol{g}(\boldsymbol{\delta}_0),$$

and the Hessian matrix is

$$\boldsymbol{H}_{\delta_0} = -\frac{\partial^2 L}{\partial \boldsymbol{\delta}\boldsymbol{\delta}^\mathsf{T}}\bigg|_{\boldsymbol{\delta}_0} = \\ 2\boldsymbol{K}^\mathsf{T}_{1:m,\bullet}\,\mathrm{diag}[\boldsymbol{g}^{-2}_{1:m}(\boldsymbol{\delta}_0)]\boldsymbol{K}_{1:m,\bullet} + 2\frac{n}{n_q}\boldsymbol{K}^\mathsf{T}_{m+1:m+n_q,\bullet}\boldsymbol{K}_{m+1:m+n_q,\bullet} + 2\lambda_n\boldsymbol{K}.$$

The form of the Hessian demonstrates the convexity of the problem. Consider a matrix of the form $\boldsymbol{A}^\mathsf{T}\boldsymbol{A}$, then $\boldsymbol{x}^\mathsf{T}\boldsymbol{A}^\mathsf{T}\boldsymbol{A}\boldsymbol{x} = \sum(\boldsymbol{x}^\mathsf{T}\boldsymbol{a}_j)^2 \geq 0$, where $\boldsymbol{a}_j$ are the columns of A. Therefore, the first two matrices in the definition of $\boldsymbol{H}$ are nonnegative definite. The last matrix in the definition of $\boldsymbol{H}$ is positive definite since κ is a positive definite function. The negative of the Hessian is the therefore positive definite, therefore the maximization problem concave over a convex set.

To show the Newton-Raphson converges, we must also show the function is coercive. By the norm equivalence theorem, there exists two positive constants c_1 and c_2 such that

$$c_1\boldsymbol{\delta}^\mathsf{T}\boldsymbol{K}\boldsymbol{\delta} \leq \boldsymbol{\delta}^\mathsf{T}\boldsymbol{K}^\mathsf{T}_{k,\bullet}\boldsymbol{K}_{k,\bullet}\boldsymbol{\delta} \leq c_2\boldsymbol{\delta}^\mathsf{T}\boldsymbol{K}\boldsymbol{\delta}$$

for all $\boldsymbol{\delta}$ and k. Therefore,

$$-L(\boldsymbol{\delta}) \geq -\sum_{i=1}^{n}\sum_{j=1}^{m_i}\log\{c_1\boldsymbol{\delta}^\mathsf{T}\boldsymbol{K}\boldsymbol{\delta}\} + c_2 n\boldsymbol{\delta}^\mathsf{T}\boldsymbol{K}\boldsymbol{\delta} + \lambda_n\boldsymbol{\delta}^\mathsf{T}\boldsymbol{K}\boldsymbol{\delta}.$$

From this, as $\|\boldsymbol{\delta}\|_K := \boldsymbol{\delta}^\top \boldsymbol{K}\boldsymbol{\delta} \to \infty$ we have $-L(\boldsymbol{\delta}) \to \infty$ which demonstrates coercivity. Therefore the Newton-Raphson algorithm converges to a unique maximizer.

The Newton-Raphson technique follows directly, and its incremented values are given by

$$\boldsymbol{\delta}_k = \boldsymbol{\delta}_{k-1} - \boldsymbol{H}^{-1}_{\delta_{k-1}} \boldsymbol{g}_{\delta_{k-1}}.$$

5.9 Proofs of Results

5.9.1 Theorem 1

For the proof, we slightly adjust our notation to make exposition easier. Let Y_i denote the points from a sample, i.e. vector of defects from the ith product. We define the log likelihood according to each observation, $l(f, Y_i) = \sum_{x \in Y_i} \log f(x) - \int f$.

This proof consists of two parts. In the first part, we will verify analogous results to lemmas 3-7 from [129] which establishes important inequalities. The second part will follow the method of [111] to demonstrate the result.

5.9.1.1 Important Inequalities

The variance of a log likelihood ratio, $\log p(W)/p_0(W)$ where $W \sim p_0$, is unbounded in general, so following [129], we look at the truncated version of the likelihoods. Let l_ϕ be defined as

$$l_\phi(f, Y_i) = \sum_{W \in Y_i} \log f_\phi(W) - \int f,$$

where $\phi > 0$ and

$$f_\phi(W) = \begin{cases} \exp(-\phi) f_0(W), & \text{if } f(W) < \exp(-\phi) f_0(W), \\ f(W), & \text{otherwise.} \end{cases}.$$

We will use the following additional notation. Let

$$a_0 = \int_0^1 f_0(z) \mathrm{d}z$$

and

$$Z_f \overset{d}{=} \log(f_\phi(W)/f_0(W)),$$

where W is drawn from the density $f_0(\cdot)/a_0$. Furthermore, let $X_f \overset{d}{=} a_0^{-1} \sum_{j=1}^{M_i} Z_j$, where Z_j are replicates of Z_f.

Lemma 3. From from [129]'s lemma 3, we have that

$$var(X_f) \leq var(Z_f) \leq 4 \exp(\phi) \| f_1^{1/2} - f_2^{1/2} \|_2^2. \tag{25}$$

Lemma 4 Using the fact that for $x \geq 0$, $\log(x+1) \leq x$,

$$\begin{aligned}
\mathbb{E} Z_f =& \int \log \left\{ f_\phi(z)/f_0(z) \right\} f_0(z)/a_0 \mathrm{d}z \\
=& 2 \int \log \left(1 + (|f_\phi(z)/f_0(z)|^{1/2} - 1) \right) f_0(z)/a_0 \mathrm{d}z \\
\leq& 2 \left[-1 + \frac{1}{a_0} \langle f_\phi^{1/2}, f_0^{1/2} \rangle \right] \\
\leq& - \frac{1}{a_0} \| f^{1/2} - f_0^{1/2} \|_2^2 + \frac{1}{a_0} \int f(z) d(z) - 1 \\
& + 2 \exp(-\phi/2) P\{ f(x) < \exp(-\phi) f_0(x) \}. \\
\leq& - \frac{1 - \delta_\phi}{a_0} \| f^{1/2} - f_0^{1/2} \|_2^2 + \frac{1}{a_0} \int f(z) d(z) - 1,
\end{aligned} \tag{26}$$

where the last line is the result of [129]'s lemma 2 and $\delta_\phi > 0$ is a constant that depends on ϕ to be decided. Lastly, we have $\mathbb{E} X_f = \mathbb{E} Z_f$.

Lemmas 5, 6 and 7. Lemma 5 from [129] implies that,

$$\mathbb{E} |Z_f|^k \leq \frac{c_0}{a_0} 2^k k! \| f^{1/2} - f_0^{1/2} \|_2^2.$$

Here, applying Minkowski's inequality,

$$[\mathbb{E} |X_f|^k]^{1/k} \leq a_0^{-1} \mathbb{E}[M_i] [\mathbb{E} |Z_f|^k]^{1/k} = [\mathbb{E} |Z_f|^k]^{1/k}.$$

This allows us to use Bernstein's inequality to derive lemma 6. Lemma 7 follows as a direct result.

5.9.1.2 Demonstration of Result

We first bound the probability in the statement of the theorem with

$$P^*\left(\sup_{\rho(f,f_0)\geq\varepsilon, f\in\mathcal{F}} n^{-1}\sum_{i=1}^{n} l(f,Y_i)-l(f_0,Y_i)-lambda_n(J(f)-J(f_0))\geq\ -c_4\varepsilon^2\right)$$
$$\leq P^*\left(\sup_{\rho(f,f_0)\geq\varepsilon, f\in\mathcal{F}} n^{-1}\sum_{i=1}^{n} a_0^{-1}[l_\phi(f,Y_i)-l_\phi(f_0,Y_i)]-\lambda_n a_0^{-1}(J(f)-J(f_0))\geq -c_4 a_0^{-1}\varepsilon^2\right),$$

which follows from the definition of l_ϕ. In the interest of brevity, we will leave the last section of this proof as a citation to Theorem 1 from [111]. To do this, we must first verify his condition A.

Condition A from [111] depends on the function spaces $L(k)=\{l^{1/2}(f,\cdot): f\in\mathcal{F}, J(f)\leq k\}$. First we study the spaces $\mathcal{F}(k)=\{f^{1/2}: f\in\mathcal{F}, J(f)\leq k\}$. We need to show that there exist constants a_1 and a_2 such that

$$\sup_{k\geq 1}\psi(\epsilon,k)=a_2 n^{1/2},$$

where $\psi(\epsilon,k)=\int_K^{K^{1/2}} H^{1/2}(u,\mathcal{L}(k))/K$ with $K=(a_1\varepsilon^2+\lambda_n(k-1))$. Here, $H(u,\mathcal{F}(k))$ is the bracketing metric entropy of $\mathcal{F}$ under distance measure ρ. Using condition (A) in our work, [127] show that $\mathcal{G}$ can be embedded in the Solbolv space with equivalent norms. Using norm equivalence Theorem (see e.g. Theorem 5.2 of [14] or example 1, case 1 of [111]), the metric entropy is bounded by $H(u,\mathcal{F}(k))\leq c(k/u)^{1/\tau}$. Because of the analogous lemma 3 demonstrated in the previous section of our proof, we have the relation $H(\epsilon,\mathcal{L})\leq H(4\exp(\phi)\epsilon,\mathcal{F})$. Therefore assumption A of [111] is satisfied with $\psi(\varepsilon,k)=\varepsilon^{-(2\tau+1)/2\tau}(a_1+c_3(k-1))^{-(2\tau+1)/4\tau}$ since

$$a_0^{-1}\lambda_n\max(J(f_0),1)\leq c_3\varepsilon^2.$$

Therefore, [111]'s condition A holds.

The following results are from the relationship $a_0^{-1}[l_\phi(f,Y_i)-l_\phi(f_0,Y_i)]\stackrel{d}{=}X_f+a_0^{-1}\int f_0-f\mathrm{d}z$ and the inequalities presented earlier in this subsection. We will now partition the function space and derive bounds on the mean and variance of these

sets. To bound the probability, it is convenient to define a two dimensional sequence of function spaces for $i, j \geq 1$,

$$A_{ij} = \left\{ f \in \mathcal{F} : \rho(f, f_0) \in [2^{i-1}\varepsilon, 2^i\varepsilon), J(f) \in [2^{j-1}J(f_0), 2^j J(f_0)) \right\},$$

and for $i \geq 1$,

$$A_{i0} = \left\{ f \in \mathcal{F} : \rho(f, f_0) \in [2^{i-1}\varepsilon, 2^i\varepsilon), J(f) \leq J(f_0) \right\}.$$

Choose ϕ such that $a_0^{-1}(1 - \delta_\phi - c_4) = a_1 > 0$. From the relation shown in (26), for any $i, j \geq 1$,

$$\inf_{A_{ij}} E\left(a_0^{-1}[l_\phi(f, Y_i) - l_\phi(f_0, Y_i) - \lambda_n(J(f) - J(f_0))]\right) \geq$$
$$a_1(2^{i-1}\varepsilon)^2 + a_0^{-1}\lambda_n(2^{j-1} - 1)J(f_0) \equiv M_{i,j}. \quad (27)$$

By assumption $a_0^{-1}\lambda_n \max(J(f_0), 1) \leq c_3\varepsilon^2$, and therefore for $i \geq 1$,

$$\inf_{A_{i0}} E\left(a_0^{-1}[l_\phi(f, Y_i) - l_\phi(f_0, Y_i) - \lambda_n(J(f) - J(f_0))]\right) \geq$$
$$a_1\left[(2^{i-1}\varepsilon)^2 + c_3\varepsilon^2\right] \equiv M_{i,0}. \quad (28)$$

Furthermore, using the relation shown in (25), we have

$$\sup_{A_{i0}} var\left(a_0^{-1}[l_\phi(f, Y_i) - l_\phi(f_0, Y_i)]\right) \leq 4\exp(\phi)\left[(2^i\varepsilon)^2 + \frac{2}{a_1}\lambda_n(2^{j-1} - 1)J(f_0)\right].$$

From the above, we bound the outer probability by

$$P^*\left(\sup_{\rho(f,f_0)\geq\varepsilon, f\in\mathcal{F}} n^{-1}\sum_{i=1}^n a_0^{-1}[l_\phi(f, Y_i) - l_\phi(f_0, Y_i)] - a_0^{-1}\lambda_n(J(f) - J(f_0)) \geq -a_0^{-1}c_4\varepsilon^2\right)$$
$$\leq \sum_{i,j=1}^\infty P^*\left(\sup_{\rho(f,f_0)\geq\varepsilon, f\in A_{ij}} n^{-1}\sum_{i=1}^n a_0^{-1}[l_\phi(f, Y_i) - l_\phi(f_0, Y_i)] \geq M_{ij}\right)$$
$$+ \sum_{i=1}^\infty P^*\left(\sup_{\rho(f,f_0)\geq\varepsilon, f\in A_{i0}} n^{-1}\sum_{i=1}^n a_0^{-1}[l_\phi(f, Y_i) - l_\phi(f_0, Y_i)] \geq M_{i0}\right).$$

We now cite [111], paragraph 4 onward, because the inequalities are exactly the same. The only difference is replacing his invocation of [129]'s lemma 7 with the statement shown in the first part of this proof.

5.9.2 Corollary 1

Let $A_n = \sup_{\rho(f,f_0)\geq\varepsilon_n} L(f) - \tilde{L}(f)$, we have that

$$P\left(\rho(f,\hat{f}_n) \geq \varepsilon_n\right) \leq P^*\left(\sup_{\rho(f,f_0)\geq\varepsilon_n, f\in\mathcal{F}} L(f) - L(f_0) - \lambda_n(J(f) - J(f_0)) \geq 2A_n\right)$$
$$\leq P^*\left(\sup_{\rho(f,f_0)<\varepsilon_n, f\in\mathcal{F}} L(f) - L(f_0) - \lambda_n(J(f) - J(f_0)) \geq -c_4\varepsilon_n^2\right) P(-A_n \geq -c_4\varepsilon_n^2)$$
$$+ P(-A_n < -c_4\varepsilon_n^2)$$

As demonstrated in section 5, $P(A_n > t) \leq \exp(-c_6 n t^2)$. Under the assumptions of Theorem 1, let $c_7 \leq \min(c_4, c_6)$, then

$$P\left(\rho(\hat{f}_n, f_0) \geq \varepsilon_n\right) \leq c_8 \exp(-2c_7 n\varepsilon_n^2),$$

for some constant $c_8 > 0$, implying that $\rho(\hat{f}_n, f_0) = \mathcal{O}_p(\varepsilon_n)$. Let ε_n be the smallest ε satisfying assumption A from [111] discussed in the previous proof. Hence $\rho(\hat{f}_n, f_0) = \mathcal{O}_p(\lambda_n^{1/2})$, and the result follows.

Chapter VI

REFERENCES

Bibliography

[1] Achard, P. and De Schutter, E., "Complex parameter landscape for a complex neuron model," *PLoS Computational Biology*, vol. 2, no. 7, p. e94, 2006.

[2] Ankenman, B., Nelson, B. L., and Staum, J., "Stochastic kriging for simulation metamodeling," *Operations Research*, vol. 58, no. 2, pp. 371–382, 2010.

[3] Antoniadis, A., "A penalty method for nonparametric estimation of the intensity function of a counting process," *Annals of the Institute of Statistical Mathematics*, vol. 41, no. 4, pp. 781–807, 1989.

[4] Arendt, P. D., Apley, D. W., and Chen, W., "Quantification of model uncertainty: calibration, model discrepancy, and identifiability," *Journal of Mechanical Design*, vol. 134, no. 10, p. 100908, 2012.

[5] Aronszajn, N., "Theory of reproducing kernels," *Trans. Amer. Math. Soc*, vol. 68, no. 3, pp. 337–404, 1950.

[6] Banerjee, S., Gelfand, A., Finley, A., and Sang, H., "Gaussian predictive process models for large spatial data sets," *Journal of the Royal Statistical Society: Series B*, vol. 70, no. 4, pp. 825–848, 2008.

[7] Barthelmann, V., Novak, E., and Ritter, K., "High dimensional polynomial interpolation on sparse grids," *Advances in Computational Mathematics*, vol. 12, no. 4, pp. 273–288, 2000.

[8] Barton, R. R., "Simulation metamodels," in *Simulation Conference Proceedings, 1998. Winter*, vol. 1, pp. 167–174, IEEE, 1998.

[9] Bayarri, M. J., Berger, J. O., Paulo, R., Sacks, J., Cafeo, J. A., Cavendish, J., Lin, C.-H., and Tu, J., "A framework for validation of computer models," *Technometrics*, vol. 49, no. 2, 2007.

[10] Bear, M. F., Connors, B. W., and Paradiso, M. A., *Neuroscience*, vol. 2. Lippincott Williams & Wilkins, 2007.

[11] Berman, M. and Turner, T. R., "Approximating point process likelihoods with glim," *Applied Statistics*, vol. 41, pp. 31–38, 1992.

[12] Bernardo, M. C., Buck, R., Liu, L., Nazaret, W. A., Sacks, J., and Welch, W. J., "Integrated circuit design optimization using a sequential strategy," *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol. 11, no. 3, pp. 361–372, 1992.

[13] Bigot, J., Gadat, S., Klein, T., and Marteau, C., "Intensity estimation of non-homogeneous poisson processes from shifted trajectories," *Electronic Journal of Statistics*, vol. 7, pp. 881–931, 2013.

[14] Birman, M. S. and Solomyak, M. Z., "Piecewise-polynomial approximations of functions of the classes w_p^α," *Matematicheskii Sbornik*, vol. 115, no. 3, pp. 331–355, 1967.

[15] Bondarenko, V. E., Szigeti, G. P., Bett, G. C., Kim, S.-J., and Rasmusson, R. L., "Computer model of action potential of mouse ventricular myocytes," *American Journal of Physiology-Heart and Circulatory Physiology*, vol. 287, no. 3, pp. H1378–H1403, 2004.

[16] Box, G. E. and Hunter, W. G., "A useful method for model-building," *Technometrics*, vol. 4, no. 3, pp. 301–318, 1962.

[17] Britton, O. J., Bueno-Orovio, A., Van Ammel, K., Lu, H. R., Towart, R., Gallacher, D. J., and Rodriguez, B., "Experimentally calibrated population of models predicts and explains intersubject variability in cardiac cellular electrophysiology," *Proceedings of the National Academy of Sciences*, vol. 110, no. 23, pp. E2098–E2105, 2013.

[18] Brumback, B. A. and Rice, J. A., "Smoothing spline models for the analysis of nested and crossed samples of curves," *Journal of the American Statistical Association*, vol. 93, no. 443, pp. 961–976, 1998.

[19] Bueno-Orovio, A., Sánchez, C., Pueyo, E., and Rodriguez, B., "Na/k pump regulation of cardiac repolarization: insights from a systems biology approach," *Pflügers Archiv-European Journal of Physiology*, vol. 466, no. 2, pp. 183–193, 2014.

[20] Cavalier, L. and Koo, J.-Y., "Poisson intensity estimation for tomographic data using a wavelet shrinkage approach," *Information Theory, IEEE Transactions on*, vol. 48, no. 10, pp. 2794–2802, 2002.

[21] Clancy, C. E. and Rudy, Y., "Linking a genetic defect to its cellular phenotype in a cardiac arrhythmia," *Nature*, vol. 400, no. 6744, pp. 566–569, 1999.

[22] Cotter, S., Roberts, G., Stuart, A., White, D., and others, "Mcmc methods for functions: modifying old algorithms to make them faster," *Statistical Science*, vol. 28, no. 3, pp. 424–446, 2013.

[23] Cowling, A., Hall, P., and Phillips, M. J., "Bootstrap confidence regions for the intensity of a poisson point process," *Journal of the American Statistical Association*, vol. 91, no. 436, pp. 1516–1524, 1996.

[24] Cressie, N. and Johannesson, G., "Fixed rank kriging for very large spatial data sets," *Journal of the Royal Statistical Society: Series B*, vol. 70, no. 1, pp. 209–226, 2008.

[25] Csilléry, K., Blum, M. G., Gaggiotti, O. E., and François, O., "Approximate bayesian computation (abc) in practice," *Trends in ecology & evolution*, vol. 25, no. 7, pp. 410–418, 2010.

[26] Currin, C., Mitchell, T., Morris, M., and Ylvisaker, D., "Bayesian prediction of deterministic functions, with applications to the design and analysis of computer experiments," *Journal of the American Statistical Association*, vol. 86, no. 416, pp. 953–963, 1991.

[27] De Iorio, M., Mueller, P., Rosner, G. L., and MacEachern, S. N., "An ANOVA model for dependent random measures," *Journal of the American Statistical Association*, vol. 99, no. 465, pp. 205–215, 2004.

[28] Diggle, P. J. and Ribeiro, P. J., *Model-based geostatistics*, vol. 13. Springer New York, 2007.

[29] Diggle, P., "A kernel method for smoothing point process data," *Applied Statistics*, vol. 34, pp. 138–147, 1985.

[30] Diggle, P. and Marron, J. S., "Equivalence of smoothing parameter selectors in density and intensity estimation," *Journal of the American Statistical Association*, vol. 83, no. 403, pp. 793–800, 1988.

[31] Diggle, P. J., Gates, D. J., and Stibbard, A., "A nonparametric estimator for pairwise-interaction point processes," *Biometrika*, vol. 74, no. 4, pp. 763–770, 1987.

[32] Diggle, P. J., Mateu, J., and Clough, H. E., "A comparison between parametric and non-parametric approaches to the analysis of replicated spatial point patterns," *Advances in Applied Probability*, vol. 32, no. 2, pp. 331–343, 2000.

[33] Ding, X., Puterman, M. L., and Bisi, A., "The censored newsvendor and the optimal acquisition of information," *Operations Research*, vol. 50, no. 3, pp. 517–527, 2002.

[34] Du, D., Yang, H., Norring, S., and Bennett, E., "In-silico modeling of glycosylation modulation dynamics in herg ion channels and cardiac electrical signals.," *IEEE journal of biomedical and health informatics*, vol. 18, no. 1, pp. 205–214, 2014.

[35] Dunson, D. B., Pillai, N., and Park, J. H., "Bayesian density regression," *Journal of the Royal Statistical Society: Series B*, vol. 69, no. 2, pp. 163–183, 2007.

[36] Ednie, A. R., Horton, K.-K., Wu, J., and Bennett, E. S., "Expression of the sialyltransferase, st3gal4, impacts cardiac voltage-gated sodium channel activity, refractory period and ventricular conduction," *Journal of molecular and cellular cardiology*, vol. 59, pp. 117–127, 2013.

[37] Ednie, A. and Bennett, E., "Modulation of voltage-gated ion channels by sialylation.," *Comprehensive Physiology*, vol. 2, no. 2, pp. 1269–1301, 2012.

[38] Fan, J., Yao, Q., and Tong, H., "Estimation of conditional densities and sensitivity measures in nonlinear dynamical systems," *Biometrika*, vol. 83, no. 1, pp. 189–206, 1996.

[39] Fan, J. and Peng, H., "Nonconcave penalized likelihood with a diverging number of parameters," *The Annals of Statistics*, vol. 32, no. 3, pp. 928–961, 2004.

[40] Forman, R. G., Kearney, V. E., and Engle, R. M., "Numerical analysis of crack propagation in cyclic-loaded structures," *Journal of Basic Engineering*, vol. 89, no. 3, pp. 459–463, 1967.

[41] Franke, R., "Scattered data interpolation: tests of some methods," *Mathematics of Computation*, vol. 38, no. 157, pp. 181–200, 1982.

[42] Furrer, R., Genton, M. G., and Nychka, D., "Covariance tapering for interpolation of large spatial datasets," *Journal of Computational and Graphical Statistics*, vol. 15, no. 3, 2006.

[43] Gatrell, A. C., Bailey, T. C., Diggle, P. J., and Rowlingson, B. S., "Spatial point pattern analysis and its application in geog raphical epidemiology," *Transactions of the Institute of British Geographers*, vol. 21, pp. 256–274, 1996.

[44] Gelfand, A., Diggle, P., Fuentes, M., and Guttorp, P., *Handbook of Spatial Statistics*. CRC Press, 2010.

[45] Gelman, A., Carlin, J. B., Stern, H. S., Dunson, D. B., Vehtari, A., and Rubin, D. B., *Bayesian data analysis*. CRC press, 2013.

[46] Gneiting, T. and Raftery, A. E., "Strictly proper scoring rules, prediction, and estimation," *Journal of the American Statistical Association*, vol. 102, no. 477, pp. 359–378, 2007.

[47] Gramacy, R. B. and Lee, H. K., "Bayesian treed gaussian process models with an application to computer modeling," *Journal of the American Statistical Association*, vol. 103, no. 483, 2008.

[48] Grant, A. O., "Cardiac ion channels," *Circulation: Arrhythmia and Electrophysiology*, vol. 2, no. 2, pp. 185–194, 2009.

[49] Grashow, R., Brookings, T., and Marder, E., "Reliable neuromodulation from circuits with variable underlying structure," *Proceedings of the National Academy of Sciences*, vol. 106, no. 28, pp. 11742–11746, 2009.

[50] Grujicic, M., He, T., Marvi, H., Cheeseman, B., and Yen, C., "A comparative investigation of the use of laminate-level meso-scale and fracture-mechanics-enriched meso-scale composite-material models in ballistic-resistance analyses," *Journal of materials science*, vol. 45, no. 12, pp. 3136–3150, 2010.

[51] Guan, Y., "On consistent nonparametric intensity estimation for inhomogeneous spatial point processes," *Journal of the American Statistical Association*, vol. 103, no. 483, pp. 1238–1247, 2008.

[52] Haaland, B. and Qian, P. Z., "Accurate emulators for large-scale computer experiments," *The Annals of Statistics*, vol. 39, no. 6, pp. 2974–3002, 2011.

[53] Handcock, M. and Stein, M., "A Bayesian analysis of kriging," *Technometrics*, vol. 35, no. 4, pp. 403–410, 1993.

[54] Hansen, M. H., Nair, V. N., and Friedman, D. J., "Monitoring wafer map data from integrated circuit fabrication processes for spatially clustered defects," *Technometrics*, vol. 39, no. 3, pp. 241–253, 1997.

[55] Henderson, D. A., Boys, R. J., Krishnan, K. J., Lawless, C., and Wilkinson, D. J., "Bayesian emulation and calibration of a stochastic computer model of mitochondrial DNA deletions in substantia nigra neurons," *Journal of the American Statistical Association*, vol. 104, no. 485, pp. 76–87, 2009.

[56] Higdon, D., Gattiker, J., Williams, B., and Rightley, M., "Computer model calibration using high-dimensional output," *Journal of the American Statistical Association*, vol. 103, no. 482, 2008.

[57] Hille, B., *Ion channels of excitable membranes*, vol. 507. Sinauer Sunderland, MA, 2001.

[58] Hodgkin, A. L. and Huxley, A. F., "A quantitative description of membrane current and its application to conduction and excitation in nerve," *The Journal of physiology*, vol. 117, no. 4, p. 500, 1952.

[59] Hudson, C. M. and Scardina, J. T., "Effect of stress ratio on fatigue-crack growth in 7075-t6 aluminum-alloy sheet," *Engineering fracture mechanics*, vol. 1, no. 3, pp. 429–446, 1969.

[60] Huxley, A., "From overshoot to voltage clamp," *Trends in neurosciences*, vol. 25, no. 11, pp. 553–558, 2002.

[61] Jäger, P., Steinmann, P., and Kuhl, E., "On local tracking algorithms for the simulation of three-dimensional discontinuities," *Computational Mechanics*, vol. 42, no. 3, pp. 395–406, 2008.

[62] Joseph, V. R. and Hung, Y., "Orthogonal-maximin Latin hypercube designs," *Statistica Sinica*, vol. 18, no. 1, pp. 171–186, 2008.

[63] Joseph, V. R. and Yan, H., "Engineering-driven statistical adjustment and calibration," *Technometrics*, vol. to appear, 2014.

[64] Karr, A. F., "Maximum likelihood estimation in the multiplicative intensity model via sieves," *The Annals of Statistics*, vol. 15, no. 2, pp. 473–490, 1987.

[65] Karr, A. F., *Point Processes Their Statistical Inference 2e*, vol. 7. CRC PressI Llc, 1991.

[66] Kass, R. E., Carlin, B. P., Gelman, A., and Neal, R. M., "Markov chain monte carlo in practice: A roundtable discussion," *The American Statistician*, vol. 52, no. 2, pp. 93–100, 1998.

[67] Kennedy, M. C. and O'Hagan, A., "Bayesian calibration of computer models," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 63, no. 3, pp. 425–464, 2001.

[68] Kim, K., Mahmoud, M. A., and Woodall, W. H., "On the monitoring of linear profiles," *Journal of Quality Technology*, vol. 35, no. 3, pp. 317–328, 2003.

[69] Kleijnen, J. P. C., *Design and analysis of simulation experiments*. Springer Publishing Company, Incorporated, 2007.

[70] Koenker, R., *Quantile regression*. Cambridge university press, 2005.

[71] Kolaczyk, E. D., "Wavelet shrinkage estimation of certain poisson intensity signals using corrected thresholds," *Statistica Sinica*, vol. 9, no. 1, pp. 119–135, 1999.

[72] Kurtz, T. G., "Limit theorems for sequences of jump markov processes approximating ordinary differential processes," *Journal of Applied Probability*, vol. 8, no. 2, pp. 344–356, 1971.

[73] Li, J., Shi, J., and Chang, T.-S., "On-line seam detection in rolling processes using snake projection and discrete wavelet transform," *Journal of manufacturing science and engineering*, vol. 129, no. 5, pp. 926–933, 2007.

[74] Li, Y., Liu, Y., and Zhu, J., "Quantile regression in reproducing kernel hilbert spaces," *Journal of the American Statistical Association*, vol. 102, no. 477, pp. 255–268, 2007.

[75] Li, Y.-D. and Lee, K. Y., "Fracture analysis on the arc-shaped interface in a layered cylindrical piezoelectric sensor polarized along its axis," *Engineering Fracture Mechanics*, vol. 76, no. 13, pp. 2065–2073, 2009.

[76] Lin, Y. and Yang, J., "On statistical moments of fatigue crack propagation," *Engineering Fracture Mechanics*, vol. 18, no. 2, pp. 243–256, 1983.

[77] Matheron, G., "Principles of geostatistics," *Economic geology*, vol. 58, no. 8, pp. 1246–1266, 1963.

[78] Matoušek, J., "On the l_2-discrepancy for anchored boxes," *Journal of Complexity*, vol. 14, no. 4, pp. 527–556, 1998.

[79] McKay, M. D., Beckman, R. J., and Conover, W. J., "Comparison of three methods for selecting values of input variables in the analysis of output from a computer code," *Technometrics*, vol. 21, no. 2, pp. 239–245, 1979.

[80] Miller Jr, R. G., *Survival analysis*. Wiley-Interscience, 2011.

[81] Mohler, G. O., Short, M. B., Brantingham, P. J., Schoenberg, F. P., and Tita, G. E., "Self-exciting point process modeling of crime," *Journal of the American Statistical Association*, vol. 106, no. 493, pp. 100–108, 2011.

[82] Møller, J. and Waagepetersen, R. P., *Statistical inference and simulation for spatial point processes*, vol. 100. CRC Press, 2004.

[83] Montpetit, M. L., Stocker, P. J., Schwetz, T. A., Harper, J. M., Norring, S. A., Schaffer, L., North, S. J., Jang-Lee, J., Gilmartin, T., Head, S. R., and others, "Regulated and aberrant glycosylation modulate cardiac electrical signaling," *Proceedings of the National Academy of Sciences*, vol. 106, no. 38, pp. 16517–16522, 2009.

[84] Morris, M. D. and Mitchell, T. J., "Exploratory designs for computational experiments," *Journal of Statistical Planning and Inference*, vol. 43, no. 3, pp. 381–402, 1995.

[85] Morris, M. D., Mitchell, T. J., and Ylvisaker, D., "Bayesian design and analysis of computer experiments: use of derivatives in surface prediction," *Technometrics*, vol. 35, no. 3, pp. 243–255, 1993.

[86] Neal, R., "Mcmc using hamiltonian dynamics," in *Handbook of Markov Chain Monte Carlo* (Brooks, S., Gelman, A., Jones, G., and Meng, X.-L., eds.), vol. 2, 2011.

[87] Neher, E., Sakmann, B., and Steinbach, J. H., "The extracellular patch clamp: a method for resolving currents through individual open channels in biological membranes," *Pflügers Archiv*, vol. 375, no. 2, pp. 219–228, 1978.

[88] Nobile, F., Tempone, R., and Webster, C. G., "A sparse grid stochastic collocation method for partial differential equations with random input data," *SIAM Journal on Numerical Analysis*, vol. 46, no. 5, pp. 2309–2345, 2008.

[89] Noble, D., Garny, A., and Noble, P. J., "How the hodgkin–huxley equations inspired the cardiac physiome project," *The Journal of physiology*, vol. 590, no. 11, pp. 2613–2628, 2012.

[90] O'Hagan, A., "Bayes–Hermite quadrature," *Journal of Statistical Planning and Inference*, vol. 29, no. 3, pp. 245–260, 1991.

[91] O'Sullivan, F., "Nonparametric estimation in the cox model," *The Annals of Statistics*, vol. 21, pp. 124–145, 1993.

[92] Owen, A., "Controlling correlations in Latin hypercube samples," *Journal of the American Statistical Association*, vol. 89, no. 428, pp. 1517–1522, 1994.

[93] Picheny, V., Ginsbourger, D., Richet, Y., and Caplin, G., "Quantile-based optimization of noisy computer experiments with tunable precision," *Technometrics*, vol. 55, no. 1, pp. 2–13, 2013.

[94] Plumlee, M., "Fast prediction of deterministic functions using sparse grid experimental designs," *Journal of the American Statistical Association*, vol. 109, no. 508, pp. 1581–1591, 2014.

[95] Plumlee, M. and Tuo, R., "Building accurate emulators for stochastic simulations via quantile kriging," *Technometrics*, vol. 56, no. 4, pp. 466–473, 2014.

[96] Pratola, M. T., Sain, S. R., Bingham, D., Wiltberger, M., and Rigler, E. J., "Fast sequential computer model calibration of large nonstationary spatial-temporal processes," *Technometrics*, vol. 55, no. 2, pp. 232–242, 2013.

[97] Qiu, P., Zou, C., and Wang, Z., "Nonparametric profile monitoring by mixed effects modeling," *Technometrics*, vol. 52, no. 3, pp. 265–277, 2010.

[98] Ramlau-Hansen, H., "Smoothing counting process intensities by means of kernel functions," *The Annals of Statistics*, vol. 11, pp. 453–466, 1983.

[99] Ramsay, J. O., Hooker, G., Campbell, D., and Cao, J., "Parameter estimation for differential equations: a generalized smoothing approach," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 69, no. 5, pp. 741–796, 2007.

[100] Ray, A. and Tangirala, S., "Stochastic modeling of fatigue crack dynamics for on-line failure prognostics," *Control Systems Technology, IEEE Transactions on*, vol. 4, no. 4, pp. 443–451, 1996.

[101] Reynaud-Bouret, P. and Rivoirard, V., "Near optimal thresholding estimation of a poisson intensity on the real line," *Electronic journal of statistics*, vol. 4, pp. 172–238, 2010.

[102] Rigby, R. and Stasinopoulos, D., "Generalized additive models for location, scale and shape," *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, vol. 54, no. 3, pp. 507–554, 2005.

[103] Ritter, K., Wasilkowski, G., and Woźniakowski, H., "Multivariate integration and approximation for random fields satisfying Sacks-Ylvisaker conditions," *The Annals of Applied Probability*, vol. 5, pp. 518–540, 1995.

[104] Rosenblatt, M., "Remarks on some nonparametric estimates of a density function," *The Annals of Mathematical Statistics*, vol. 27, pp. 832–837, 1956.

[105] Sacks, J., Welch, W., Mitchell, T., and Wynn, H., "Design and analysis of computer experiments," *Statistical Science*, vol. 4, no. 4, pp. 409–423, 1989.

[106] Santner, T., Williams, B., and Notz, W., *The design and analysis of computer experiments*. Springer Verlag, 2003.

[107] Sarkar, A. X., Christini, D. J., and Sobie, E. A., "Exploiting mathematical models to illuminate electrophysiological variability between individuals," *The Journal of physiology*, vol. 590, no. 11, pp. 2555–2567, 2012.

[108] Schölkopf, B., Herbrich, R., and Smola, A. J., "A generalized representer theorem," in *Computational learning theory*, pp. 416–426, Springer, 2001.

[109] Senthilselvan, A., "Penalized likelihood estimation of hazard and intensity functions," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 49, pp. 170–174, 1987.

[110] Shen, X., "On methods of sieves and penalization," *The Annals of Statistics*, vol. 25, no. 6, pp. 2555–2591, 1997.

[111] Shen, X., "On the method of penalization," *Statistica Sinica*, vol. 8, pp. 337–358, 1998.

[112] Smolyak, S. A., "Quadrature and interpolation formulas for tensor products of certain classes of functions," *Soviet Math Dokl.*, vol. 4, pp. 240–243, 1963.

[113] Sobczyk, K., "Modelling of random fatigue crack growth," *Engineering fracture mechanics*, vol. 24, no. 4, pp. 609–623, 1986.

[114] Stein, M., *Interpolation of Spatial Data: some theory for kriging*. Springer Verlag, 1999.

[115] Stephens, R. I. and Fuchs, H. O., *Metal fatigue in engineering*. Wiley New York, 2001.

[116] Stocker, P. J. and Bennett, E. S., "Differential sialylation modulates voltage-gated na+ channel gating throughout the developing myocardium," *The Journal of general physiology*, vol. 127, no. 3, pp. 253–265, 2006.

[117] Stone, C. J., "Optimal global rates of convergence for nonparametric regression," *The Annals of Statistics*, vol. 10, no. 4, pp. 1040–1053, 1982.

[118] Sze, D. Y., "Or practice: A queueing model for telephone operator staffing," *Operations Research*, vol. 32, no. 2, pp. 229–249, 1984.

[119] Tang, B., "Orthogonal array-based Latin hypercubes," *Journal of the American Statistical Association*, vol. 88, no. 424, pp. 1392–1397, 1993.

[120] Temlyakov, V., "Approximate recovery of periodic functions of several variables," *Mathematics of the USSR-Sbornik*, vol. 56, no. 1, p. 249, 1987.

[121] Thorarinsdottir, T. L., Gneiting, T., and Gissibl, N., "Using proper divergence functions to evaluate climate models," *Preprint*, 2013.

[122] Tian, L., Zucker, D., and Wei, L., "On the cox model with time-varying regression coefficients," *Journal of the American Statistical Association*, vol. 100, no. 469, pp. 172–183, 2005.

[123] van der Vaart, A. and van Zanten, H., "Information rates of nonparametric gaussian process methods," *The Journal of Machine Learning Research*, vol. 12, pp. 2095–2119, 2011.

[124] Wahba, G., *Spline models for observational data*, vol. 59. Society for Industrial Mathematics, 1990.

[125] Wang, Y. and Wahba, G., "Smoothing spline models for the analysis of nested and crossed samples of curves: Comment," *Journal of the American Statistical Association*, vol. 93, no. 443, pp. 976–980, 1998.

[126] Wasilkowski, G. and Woźniakowski, H., "Explicit cost bounds of algorithms for multivariate tensor product problems," *Journal of Complexity*, vol. 11, no. 1, pp. 1–56, 1995.

[127] Wendland, H., *Scattered Data Approximation*. Cambridge Monographs on Applied and Computational Mathematics, 2005.

[128] Willett, R. M. and Nowak, R. D., "Multiscale poisson intensity and density estimation," *Information Theory, IEEE Transactions on*, vol. 53, no. 9, pp. 3171–3187, 2007.

[129] Wong, W. H. and Shen, X., "Probability inequalities for likelihood ratios and convergence rates of sieve mles," *The Annals of Statistics*, vol. 23, pp. 339–362, 1995.

[130] Woźniakowski, H., "Average case complexity of linear multivariate problems i. theory," *Journal of Complexity*, vol. 8, no. 4, pp. 337–372, 1992.

[131] Xiu, D., *Numerical Methods for Stochastic Computations: a spectral method approach*. Princeton Univ Pr, 2010.

[132] Xiu, D. and Hesthaven, J., "High-order collocation methods for differential equations with random inputs," *SIAM Journal on Scientific Computing*, vol. 27, no. 3, p. 1118, 2006.

[133] Yang, F., Ankenman, B. E., and Nelson, B. L., "Estimating cycle time percentile curves for manufacturing systems via simulation," *INFORMS Journal on Computing*, vol. 20, no. 4, pp. 628–643, 2008.

[134] Yang, J., Salivar, G., and Annis, C., "Statistical modeling of fatigue-crack growth in a nickel-base superalloy," *Engineering Fracture Mechanics*, vol. 18, no. 2, pp. 257–270, 1983.

[135] Ye, K. Q., "Orthogonal column Latin hypercubes and their application in computer experiments," *Journal of the American Statistical Association*, vol. 93, no. 444, pp. 1430–1439, 1998.

[136] Zhou, Q., Zou, C., Wang, Z., and Jiang, W., "Likelihood-based ewma charts for monitoring poisson count data with time-varying sample sizes," *Journal of the American Statistical Association*, vol. 107, no. 499, pp. 1049–1062, 2012.

[137] Zou, C. and Qiu, P., "Multivariate statistical process control using lasso," *Journal of the American Statistical Association*, vol. 104, no. 488, pp. 1586–1596, 2009.

[138] Zou, C., Tsung, F., and Wang, Z., "Monitoring profiles based on nonparametric regression methods," *Technometrics*, vol. 50, no. 4, pp. 512–526, 2008.

[139] Zucker, D. M. and Karr, A. F., "Nonparametric survival analysis with time-dependent covariate effects: a penalized partial likelihood approach," *The Annals of Statistics*, vol. 18, pp. 329–353, 1990.