2017

Steady-State Co-Kriging Models

Sahar Hemmati

Steady-State Co-Kriging Models

Sahar Hemmati

Thesis submitted
to the College of Engineering and Mineral Resources
at West Virginia University

in partial fulfillment of the requirements for the degree of

Master of Science in
Industrial Engineering

Feng Yang, Ph.D., Chair
Majid Jaridi, Ph.D.
Xi Chen, Ph.D.

Department of Industrial and Management Systems Engineering

Morgantown, West Virginia
July 2017

Keywords: Co-Kriging, Stochastic Kriging, Gaussian process,
Steady-State Simulation, Intrinsic Variability

Abstract

Steady-State Co-Kriging Models

Sahar Hemmati

In deterministic computer experiments, a computer code can often be run at different levels of complexity/fidelity and a hierarchy of levels of code can be obtained. The higher the fidelity and hence the computational cost, the more accurate output data can be obtained. Methods based on the co-kriging methodology Cressie (2015) for predicting the output of a high-fidelity computer code by combining data generated to varying levels of fidelity have become popular over the last two decades. For instance, Kennedy and O'Hagan (2000) first propose to build a metamodel for multi-level computer codes by using an auto-regressive model structure. Forrester et al. (2007) provide details on estimation of the model parameters and further investigate the use of co-kriging for multi-fidelity optimization based on the efficient global optimization algorithm Jones et al. (1998). Qian and Wu (2008) propose a Bayesian hierarchical modeling approach for combining low-accuracy and high-accuracy experiments. More recently, Gratiet and Cannamela (2015) propose sequential design strategies using fast cross-validation techniques for multi-fidelity computer codes.

This research intends to extend the co-kriging metamodeling methodology to study steady-state simulation experiments. First, the mathematical structure of co-kriging is extended to take into account heterogeneous simulation output variances. Next, efficient steady-state simulation experimental designs are investigated for co-kriging to achieve a high prediction accuracy for estimation of steady-state parameters. Specifically, designs consisting of replicated longer simulation runs at a few design points and replicated shorter simulation runs at a larger set of design points will be considered. Also, design with no replicated simulation runs at long simulation is studied, along with different methods for calculating the output variance in absence of replicated outputs.

Stochastic co-kriging (SCK) method is applied to an M/M/1, as well as an M/M/5 queueing system. In both examples, the prediction performance of the SCK model is promising. It is also shown that the SCK method provides better response surfaces compared to the SK method.

"The moving finger writes,
and having written moves on.
Nor all thy piety nor all thy wit,
can cancel half a line of it. "

Omar Khayyam

Acknowledgements

First, I would like to thank my dear parents, for all the support and love that they have had for me through every single stage of my personal and academic life. They have always made me believe in myself and fly high. I send my gratitude to Professor Majid Jaridi, who has been beyond a professor, my "Patient Stone", making me feel the relief of having a trustworthy and caring person when I am thousands of miles away from my family. I also thank Dr Feng Yang, my advisor and Dr Xi Chen, my committee member, for being really helpful thorough my master studies at West Virginia University and being academically supportive of me through this research.

To my beloved family

Contents

List of Figures

List of Tables

Chapter 1

INTRODUCTION

Computer simulations are highly utilized in technology-oriented era that we live in. Complex scientific systems can be modeled into a computer code with use of simulation models. This will make the study of the system easier and more flexible. Moreover, high costs of implementing real experiments can be eliminated. Even more interesting, there are situations that resources for conducting actual experiment are very limited or highly valuable, such as those involving human subjects. In such cases, there is either no data or very small pool of data that might not be enough for analysis. In all mentioned scenarios, simulation models can make the life easier for us.

However, simulation models themselves, are sometimes hard to deal with, in the sense that the results that are retrieved must be carefully analyzed. After all, they pertain to models of the complex systems, not the real systems, so human error terms in designing the simulation model and heterogeneous simulation output error need to be considered. Plus, simulation outputs are indeed stochastic, not deterministic, so this leads to further work for output analysis.

Speaking of simulation models, they are not always the simplest, easy to get results and straight forward computer codes. Even if we ignore the amount of effort spent on building the model and predicting the parameters, simulation models can get very complex and costly to execute. One of the most important factors that can be relevant to complexity of simulation model, is the accuracy of outputs. The more detailed and well-constructed the model, the better the results, also the longer time it takes for the model to generate results. On the other hand, rough models are less costly to construct and faster to execute, but the results are not as

accurate as detailed simulation models, but of course they provide us with useful information.

Methods based on the co-kriging methodology (Cressie, 2015) for predicting the output of a high-fidelity computer code by combining data generated to varying levels of fidelity have flourished over the last two decades. For instance, Kennedy and O'Hagan (2000) first propose to build a metamodel for multi-level computer codes by using an auto-regressive model structure. Forrester et al. (2007) provide details on estimation of the model parameters and further investigate the use of co-kriging for multi-fidelity optimization based on the efficient global optimization algorithm (Jones et al., 1998). Qian and Wu (2008) propose a Bayesian hierarchical modeling approach for combining low-accuracy and high-accuracy experiments. More recently, Gratiet and Cannamela (2015) propose sequential design strategies using fast cross-validation techniques for multi-fidelity computer codes.

In the context of stochastic simulation, steady-state simulations are often employed for studying long-run system behavior, and they play a significant role in system design and risk assessment. Long-run performance of stochastic systems such as telecommunication networks is often evaluated by steady-state mean and quantiles of the system's response times Jeong et al. (2005). Therefore, estimation of steady-state parameters of complex stochastic systems is of great interest to simulation researchers and practitioners.

There exists a plethora of work on point or interval estimation of mean performance measure implied by a steady-state simulation. Various data collection and analysis methods have been proposed to overcome the two challenges arising from output analysis of a steady-state simulation, namely, the initial bias in the sample mean as a point estimator caused by the initial conditions and the difficulty in estimating the variance of the sample mean due to correlations in the sequence of outputs from within a single replication. Existing variance estimation methods include those based on independent replications (IR), batch means (BM), overlapping batch means (OBM), uncorrelated sampling, regenerative cycles, spectral analysis, autoregressive representation and standardized time series, etc.; see Pawlikowski (1990) for a survey on various methods proposed for steady-state queueing simulations by early 1990's. More recently, Argon et al. (2013) propose the replicated batch means approach (RBM), known as a compromise method between IR and BM (Alexopoulos and Goldsman, 2004).

Assuming a simulation budget constraint given in terms of simulation clock time or the number of discretely-indexed observations, a decision must be made before the simulations are run as to the number of independent, identically initialized and terminated replications to make, and the runlength of each. It is well known that running a "long" simulation (i.e., taking the runlength large) will result in a sample mean that is "close" to the true mean performance; correspondingly, we will refer to a "long" simulation replication as a high-fidelity one and a "short" simulation replication as a low-fidelity one. The question of "whether a single long replication is preferable to several shorter ones" has been studied before; for example, see Kelton (1986), Whitt (1991), Alexopoulos and Goldsman (2004) and Grassmann (2016).

Relatively little attention has been given to metamodeling approaches for approximating a steady-state performance measure response surface across a design space of interest, with exceptions of Yang et al. (2008), Bekki et al. (2014) and Chen and Kim (2014), to name a few. In particular, an important yet underdeveloped topic is whether and how one can construct an adequate metamodel for approximating a mean response surface under a given simulation budget constraint, by utilizing steady-state simulation runs performed to controlled levels of fidelity at selected design points.

1.1 Statement of the Problem

In this research,the data sets are assumed to be constructed from steady-state simulation outputs. It is clear that these results are stochastic, meaning that it would be a naive action to dismiss inherent variability of these outputs. The more accurate the simulation model, the less intrinsic variability of outputs. This is the most basic expectation from high cost simulation model of interest in the research under work.

In studying steady-state simulation experiments, it is expensive to have detailed and accurate simulations run for long time. On the other hand, approximate simulations are cheaper to run, but the results derived form those, are not precise. One approach to this problem, is to build a model which combines the results from both simulations, expensive complex simulation with cheaper rough simulations.

In reality, we can have L levels of fidelity for simulations $\mathbf{D}_1$, $\mathbf{D}_2$,..., $\mathbf{D}_L$ however we are going to base our work on two levels of fidelity, only.

1.2 Research Objectives

In this work the main goal is to extend the co-kriging metamodeling methodology to study steady-state simulation experiments. The mathematical structure of co-kriging is extended to encompass heterogeneous simulation output variability. However, achieving intrinsic variance values is not as straight forward as one might think. For the high fidelity data set that is constructed from detailed, high cost simulation model, it might not be affordable to have more than one replication. Further output analysis is required to estimate the values for intrinsic variability at each design point.

1.3 Research Approach

Response surfaces of queueing systems can get very complicated, that's why a Gaussian Process model is a good fit to the target surfaces because of its flexibility and ability to provide valid statistical inference.

The outline for the rest of this research is as follows. Chapter 2 is allocated to review of the existing literature. Chapter 3 describes the methodology used in this work, and presents a brief explanation of experimental design used. In Chapter 4, the modeling efficiency of co-kriging is examined, via empirical studies. In Chapter 5, finally, the conclusions and recommendations for further study are provided.

Chapter 2

LITERATURE REVIEW

In simulation-based experiments, it is critical to achieve a balance between simulation accuracy and cost. The higher the budget, the more accurate results can be achieved. As it was previously mentioned, a combination of cheap simulation experiments with higher number of design points and replications, and more expensive, higher accuracy simulation code can be used to better fit a model to existing data. Kennedy and O'Hagan (2001) states that the approximation simulations can be used in order to estimate results for more complex simulations, where running at more design points is not affordable. They apply this method to simulation of oil reservoir.

Kennedy and O'Hagan (2001) uses Bayesian Analysis,based on an auto-regressive model. They conclude that if at a design point, the results of a sophisticated code is available, no more information can be learned from approximated simulation. Further, they consider that high-fidelity response surface is a multiple of low-fidelity response plus a Gaussian process which is referred to as difference model. They continue the construction of model based on difference model.

In later studies, Qian et al. (2008), use detailed and approximate simulation models to build surrogate models. Since detailed simulations are accompanied with computational complexity, approximated simulations are used to alleviate the problem. First, they build a surrogate model based on approximate simulation. Gaussian Processes are appropriate choice in this context, because of several statistical properties that they have. Then, they modify it to fit the detailed simulations. The next step is to build the final model based on two previous models. Qian et al. (2008) further continues to come up with the best design using both

approximate and detailed simulation. This research intends to maximize the heat transfer of a microprocessor with linear cellular material.

Forrester et al. (2007) deals with data from different levels of fidelity in order to build surrogate models. Gaussian Processes again are the key in their work as well, since they can be easily adapted to represent models built based on correlation. For the design of low-fidelity simulation model, their work is premised on Morris et al. (1993)'s work, where equally spaced design points are chosen. The design points are chosen as such the minimum distance between points is maximized. The same rule is used for selection of subset for high-fidelity data set, with difference that the space is discrete, in contrary to the first selection space that was continuous.

The previous studies in co-kriging are all concerned about deterministic computer experiments, with no heteroscedastic simulation variances taken into account. In the stochastic simulation setting, Ankenman et al. (2010) propose to add the intrinsic error term to conventional kriging model, such that the error terms are identically and independently distributed across the replications at each design point, but there might be correlation between the error terms from one design point to the other. This assumption, changes the covariance matrix structure used for fitting the kriging model.

For steady-state simulation, one challenging task is to estimate the variance of simulation outputs from a single simulation replication, given that replicating the simulation is prohibitively expensive. Also, another crtitical decision is about the data points and run length that will provide the steady state outputs. Ni and Henderson (2015) says that the systems with higher utilization (larger ρ), need longer run length to provide accurate estimations of steady state measures. The later challenge mentioned is referred to as design of experiment. It is the question of which points should be chosen to build the low-fidelity data set? How long should the simulation code run for? How to choose a subset of low-fidelity samples to construct the high fidelity data set? And how long more to simulate the system at these points to get statistically better and more accurate responses?

With the total budget to be spent on the experiments in hand, the experimenter can choose the design points to run the simulation code at. In this research evenly-spaced design is used. This design proposed by Morris and Mitchell (1995) deals with maximizing the minimum distance between design points. Selection

of low fidelity design points, is in a continuous space. However, for high fidelity design points, since they are a subset of low fidelity points, the selection space is discrete, yet the idea is the same.

For estimation of variance, there are many methods of which two main categories were utilized in this work; non-overlapping batch mean (NBM) and overlapping batch mean (OBM) methods. These methods that are proposed by Alexopoulos et al. (2007b), break the simulation run into batches of the same size (non-overlapping or overlapping) and then measure the mean value of interest on each batch and finally give an estimate of variability through the whole run.In both of the aforementioned methods, one critical decision is the batch size that should be used. Song (1996) proposes that sample size should be such that the the estimated correlation ρ for the samples gets placed in the interval $(-p.\sigma(\rho), p.\sigma(\rho))$, where p is a constant parameter between 2 and 4 and σ represents the standard deviation. In another research, with the similar objective, Song and Schmeiser (1995) suggest that optimal sample size for minimizing mean square error, depends merely on sample size n and the ration λ_1/λ_0 where $\lambda_1 = \sum_{h=-\infty}^{\infty} |h|\rho_h$, $\lambda_0 = \sum_{h=-\infty}^{\infty} \rho_h$ and h represents the covariance lag.

In this research the author is interested in using two different levels of fidelity for simulation model outputs, along with taking into account the intrinsic variability of simulation outputs. This research in some manner, is a combination of Kennedy and O'Hagan (2000) and Ankenman et al. (2010)'s work. Steady state simulation co-kriging will be a version of original co-kriging model, with replicated simulation outputs at design points. This method is different from those mentioned in following ways.

1. Accounting for the heterogeneous variability of simulation codes, makes their use for modeling the real systems more realistic.
2. Since the simulation outputs are not considered to be deterministic anymore, which means intrinsic variance is involved, a stochastic kriging model is built for the low-fidelity data set and the difference model resulting from difference of low-fidelity and high-fidelity data sets.
3. This research also deals with the cases that direct calculation of intrinsic variability is not possible. Since high-fidelity simulations are computationally complicated and expensive, in the case that replication of simulation model is not affordable, with utilizing output analysis methods such as Alexopoulos

et al. (2007b) and Goldsman and Nelson (2006), an estimation of variance is used to continue with the steady state co-kriging.

Chapter 3

METHODOLOGY

As it was discussed in previous chapters, when data from different levels of accuracy are available, co-kriging method proves to be helpful. However, this method is not capable of handling output intrinsic variability due to stochastic essence of simulation results. In this chapter, co-kriging method is adapted to better fit the multi-fidelity data sets with replicated simulation runs. This model can be generalized to any data set with intrinsic variability.

Let $\mathbf{D}_1$ and $\mathbf{D}_2$ represent, respectively, the design-point sets in $\mathcal{X} \subset \mathbb{R}^d$ for running the low- and high-fidelity simulation runs. More specifically, let $\mathbf{D}_1 = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{k_1}\}$ and $\mathbf{D}_2 = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{k_2}\}$, such that $\mathbf{D}_1 = \mathbf{D}_2 \cup \{\mathbf{x}_{k_2+1}, \mathbf{x}_{k_2+1}, \ldots, \mathbf{x}_{k_1}\}$.

At design point $\mathbf{x}_i$ in $\mathbf{D}_1$ (for $i = 1, 2, \ldots, k_1$), we perform $n_i^{\{1\}}$ low-fidelity simulation replications and generate independent and identically distributed (i.i.d.) simulation outputs $\{\mathcal{Y}_j^{\{1\}}(\mathbf{x}_i)\}_{j=1}^{n_i^{\{1\}}}$. Specifically, the jth low-fidelity simulation replication has a simulation runlength of $s_i^{\{1\}}$ (in terms of run time or number of more basic simulation outputs) which produces the low-fidelity simulation output $\mathcal{Y}_j^{\{1\}}(\mathbf{x}_i)$, for $j = 1, 2, \ldots, n_i^{\{1\}}$.

On the other hand, at design point $\mathbf{x}_i$ in $\mathbf{D}_2$ (for $i = 1, 2, \ldots, k_2$), we perform $n_i^{\{2\}}$ high-fidelity simulation replications and generate i.i.d. simulation outputs $\{\mathcal{Y}_j^{\{2\}}(\mathbf{x}_i)\}_{j=1}^{n_i^{\{2\}}}$. That is, the jth high-fidelity simulation replication has a runlength of $s_i^{\{2\}}$ which produces the high-fidelity simulation output $\mathcal{Y}_j^{\{2\}}(\mathbf{x}_i)$. Furthermore, we assume $s_i^{\{2\}} \gg s_i^{\{1\}}$, for $i = 1, 2, \ldots, k_2$. Hence, $\mathbf{D}_2$ denotes the set of design

points where more simulation efforts are expended. However, the number of replications $n_i^{\{2\}}$ is not necessarily larger than $n_i^{\{1\}}$, for $i = 1, 2, \ldots, k_2$, as we typically want to conduct less replications of the computationally intensive simulation runs. Note that, if $\mathcal{Y}_j^{\{2\}}(\mathbf{x}_i)$ values are known, no more helpful information can be gained from $\mathcal{Y}_j^{\{1\}}(\mathbf{x}_i)$. In other words, in presence of high accuracy simulation outputs, there is no need for low accuracy simulations at those design points.

3.1 Co-Kriging for Steady-State Simulation

In this section, the co-kriging metamodeling methodology is extended to study steady-state simulation experiments. The mathematical structure of co-kriging is expanded to encompass heterogeneous simulation output variances, which are referred to as intrinsic variability. Intrinsic variability is due to the nature of simulation experiments themselves, sine they produce stochastic outputs. The low-accuracy performance measure estimator $\bar{\mathcal{Y}}^{\{1\}}(\mathbf{x}_i)$ at design point $\mathbf{x}_i \in \mathbf{D}_1$ can be modeled as

$$\begin{aligned} \bar{\mathcal{Y}}^{\{1\}}(\mathbf{x}_i) &= \frac{1}{n_i^{\{1\}}} \sum_{j=1}^{n_i^{\{1\}}} \mathcal{Y}_j^{\{1\}}(\mathbf{x}_i) \\ &= \mathsf{Y}^{\{1\}}(\mathbf{x}_i) + \bar{\zeta}^{\{1\}}(\mathbf{x}_i) \quad i = 1, 2, \ldots, k_1, \end{aligned} \tag{3.1}$$

where $\mathsf{Y}^{\{1\}}(\mathbf{x}_i)$ denotes the unknown true mean of $\bar{\mathcal{Y}}^{\{1\}}(\mathbf{x}_i)$, and $\bar{\zeta}^{\{1\}}(\mathbf{x}_i) = \sum_{j=1}^{n_i^{\{1\}}} \zeta_j^{\{1\}}(\mathbf{x}_i)/n_i^{\{1\}}$ denotes the simulation error in the estimator $\bar{\mathcal{Y}}^{\{1\}}(\mathbf{x}_i)$. Notice that the $\zeta_j^{\{1\}}(\mathbf{x}_i)$'s represent the i.i.d. simulation errors with zero mean and variance $\mathsf{V}^{\{1\}}(\mathbf{x}_i)$. Hence, $\mathrm{Var}(\bar{\zeta}^{\{1\}}(\mathbf{x}_i)) = \mathsf{V}^{\{1\}}(\mathbf{x}_i)/n_i^{\{1\}}$ and it decreases with the number of replications applied at $\mathbf{x}_i$. We note that $\mathsf{V}^{\{1\}}(\mathbf{x}_i)$ measures the variance of the simulation output from each low-fidelity simulation replication, and it decreases with the simulation runlength $s_i^{\{1\}}$. If replications are available at $\mathbf{x}_i$ (i.e., $n_i^{\{1\}} > 1$), then $\mathsf{V}^{\{1\}}(\mathbf{x}_i)$ can be estimated by the sample variance $\widehat{\mathsf{V}}^{\{1\}}(\mathbf{x}_i)$ obtained at $\mathbf{x}_i \in \mathbf{D}_1$,

$$\widehat{\mathsf{V}}^{\{1\}}(\mathbf{x}_i) = \frac{1}{n_i^{\{1\}} - 1} \sum_{j=1}^{n_i^{\{1\}}} (\mathcal{Y}_j^{\{1\}}(\mathbf{x}_i) - \bar{\mathcal{Y}}^{\{1\}}(\mathbf{x}_i))^2, \quad i = 1, 2, \ldots, k_1.$$

We provide some further details on $\mathsf{Y}^{\{1\}}(\mathbf{x}_i)$ which can be described as follows:

$$\mathsf{Y}^{\{1\}}(\mathbf{x}_i) = \mathbf{f}_1(\mathbf{x}_i)^\top \boldsymbol{\beta}_1 + \mathsf{M}_1(\mathbf{x}_i),$$

where $\boldsymbol{\beta}_1$ is a $p_1 \times 1$ vector of parameters and $\mathbf{f}_1(\cdot)$ is a vector of known regression functions of compatible dimensions. As treated in the design and analysis of deterministic computer experiments literature Santner et al. (2003), we assume that $\mathsf{M}_1(\cdot)$ is a mean-zero stationary Gaussian random field. There exists a spatial correlation function $\mathcal{R}_1(\cdot;\boldsymbol{\theta}_1)$ that measures the correlation of the values of $\mathsf{M}_1(\mathbf{x}_i)$ and $\mathsf{M}_1(\mathbf{x}_\ell)$. This correlation is determined by the distance between $\mathbf{x}_i$ and $\mathbf{x}_\ell$ measured along each of the d dimensions, and the $d \times 1$ parameter vector $\boldsymbol{\theta}_1 = (\theta_{11}, \theta_{12}, \ldots, \theta_{1d})^\top$ controls how quickly the spatial correlation diminishes as the two points become farther apart in each direction. Commonly used correlation functions include the Gaussian correlation function, Matérn correlation functions, and the exponential correlation function (see Chapter 4 of Rasmussen and Williams (2006)); we choose to use the popular Gaussian correlation function $\mathcal{R}_1(\mathbf{x}_i, \mathbf{x}_\ell; \boldsymbol{\theta}_1) = \exp\left(-\sum_{r=1}^{d} \theta_{1r}(x_{ir} - x_{\ell r})^2\right)$ in this paper. Given a correlation function, the implied covariance function is given by

$$\mathrm{Cov}(\mathsf{M}_1(\mathbf{x}_i), \mathsf{M}_1(\mathbf{x}_\ell)) = \tau_1^2 \mathcal{R}_1(\mathbf{x}_i, \mathbf{x}_\ell; \boldsymbol{\theta}_1)\ , \tag{3.2}$$

where τ_1^2 denotes the variance of $\mathsf{M}_1(\mathbf{x})$ for all $\mathbf{x} \in \mathcal{X}$.

On the other hand, we model the high-accuracy performance measure estimator $\bar{\mathcal{Y}}^{\{2\}}(\mathbf{x}_i)$ obtained at $\mathbf{x}_i \in \mathbf{D}_2$ as follows

$$\begin{aligned} \bar{\mathcal{Y}}^{\{2\}}(\mathbf{x}_i) &= \frac{1}{n_i^{\{2\}}} \sum_{j=1}^{n_i^{\{2\}}} \mathcal{Y}_j^{\{2\}}(\mathbf{x}_i) \\ &= \mathsf{Y}^{\{2\}}(\mathbf{x}_i) + \bar{\zeta}^{\{2\}}(\mathbf{x}_i), \\ &= \rho \mathsf{Y}^{\{1\}}(\mathbf{x}_i) + \delta(\mathbf{x}_i) + \bar{\zeta}^{\{2\}}(\mathbf{x}_i), \quad i = 1, 2, \ldots, k_2, \end{aligned} \tag{3.3}$$

where $\mathsf{Y}^{\{2\}}(\mathbf{x}_i)$ represents the true mean of $\bar{\mathcal{Y}}^{\{2\}}(\mathbf{x}_i)$, $\delta(\mathbf{x}_i)$ denotes the difference between $\mathsf{Y}^{\{2\}}(\mathbf{x}_i)$ and $\rho \mathsf{Y}^{\{1\}}(\mathbf{x}_i)$ on which we will elaborate later. Notice that the $\zeta_j^{\{2\}}(\mathbf{x}_i)$'s denote the i.i.d. simulation errors with zero mean and variance $\mathsf{V}^{\{2\}}(\mathbf{x}_i)$, and $\bar{\zeta}^{\{2\}}(\mathbf{x}_i) := \sum_{j=1}^{n_i^{\{2\}}} \zeta_j^{\{2\}}(\mathbf{x}_i)/n_i^{\{2\}}$ denotes the average simulation error across the $n_i^{\{2\}}$ simulation replications at $\mathbf{x}_i$. Notice that $\mathrm{Var}(\bar{\zeta}^{\{2\}}(\mathbf{x}_i)) = \mathsf{V}^{\{2\}}(\mathbf{x}_i)/n_i^{\{2\}}$ and it decreases with the number of replications $n_i^{\{2\}}$ applied at $\mathbf{x}_i$. Here $\mathsf{V}^{\{2\}}(\mathbf{x}_i)$

represents the variance of the simulation output generated from each high-fidelity simulation replication, and it decreases with the simulation runlength $s_i^{\{2\}}$. If replications are available (i.e., $n_i^{\{2\}} > 1$), then $\mathsf{V}^{\{2\}}(\mathbf{x}_i)$ can be estimated by the sample variance $\widehat{\mathsf{V}}^{\{2\}}(\mathbf{x}_i)$ obtained at $\mathbf{x}_i$ in a similar fashion as given in (3.1). We provide more details on estimation of $\mathsf{V}^{\{2\}}(\mathbf{x}_i)$ from a single high-fidelity simulation replication in Subsection 3.1.2.

We note that the model given in (3.3) relies on the following Markov property about true mean performance values implied by two-fidelity levels of simulation runs as introduced by Kennedy and O'Hagan (2000):

$$\text{Cov}\left(\mathsf{Y}^{\{2\}}(\mathbf{x}), \mathsf{Y}^{\{1\}}(\widetilde{\mathbf{x}}) | \mathsf{Y}^{\{1\}}(\mathbf{x})\right) = 0, \tag{3.4}$$

for all $\mathbf{x} \neq \widetilde{\mathbf{x}}$. This property essentially states that if the true mean performance value implied by a *low-fidelity* simulation run at $\mathbf{x}$ is known, then we can learn no more about the true mean performance value implied from a *high-fidelity* simulation run at $\mathbf{x}$ from knowing any mean performance value of a *low-fidelity* simulation run at $\widetilde{\mathbf{x}}$ for $\widetilde{\mathbf{x}} \neq \mathbf{x}$.

We further model the difference term $\delta(\mathbf{x}_i)$ specified in (3.3) as

$$\delta(\mathbf{x}_i) = \mathbf{f}_2(\mathbf{x}_i)^\top \boldsymbol{\beta}_2 + \mathsf{M}_2(\mathbf{x}_i), \tag{3.5}$$

where $\boldsymbol{\beta}_2$ is a vector of unknown parameters, $\mathbf{f}_2(\cdot)$ is a vector of known regression functions of compatible dimensions and $\mathsf{M}_2(\cdot)$ is a stationary Gaussian process with mean zero, and covariance function $\text{Cov}\left(\mathsf{M}_2(\mathbf{x}_i), \mathsf{M}_2(\mathbf{x}_\ell)\right) = \tau_\delta^2 \mathcal{R}_\delta(\mathbf{x}_i, \mathbf{x}_\ell; \boldsymbol{\theta}_\delta)$. Notice that the discussion given for the spatial correlation function and hyperparameters for $\mathsf{M}_1(\cdot)$ applies to the spatial correlation function $\mathcal{R}_\delta(\cdot, \cdot; \boldsymbol{\theta}_\delta)$ and the hyperparameters τ_δ and $\boldsymbol{\theta}_\delta$ for $\mathsf{M}_2(\mathbf{x}_i)$ here .

The true quantity of interest in our context, $\mathsf{Y}(\mathbf{x})$, can be a steady-state distribution parameter such as the steady-state mean of a random quantity of interest at $\mathbf{x}$. In spite that neither of the low- and high-fidelity point estimators, $\bar{\mathcal{Y}}^{\{1\}}(\mathbf{x}_i)$ and $\bar{\mathcal{Y}}^{\{2\}}(\mathbf{x}_i)$, is unbiased for $\mathsf{Y}(\mathbf{x})$ (or equivalently, $\mathsf{Y}^{\{1\}}(\mathbf{x}_i) \neq \mathsf{Y}(\mathbf{x}_i)$ and $\mathsf{Y}^{\{2\}}(\mathbf{x}_i) \neq \mathsf{Y}(\mathbf{x}_i)$), the simulation runlength ($s_i^{\{1\}}$ or $s_i^{\{2\}}$) applied at a design point determines the bias and variance of the point estimator obtained. In particular, $|\text{Bias}[\bar{\mathcal{Y}}^{\{2\}}(\mathbf{x}_i)]| \leq |\text{Bias}[\bar{\mathcal{Y}}^{\{1\}}(\mathbf{x}_i)]|$ and $\mathsf{V}^{\{2\}}(\mathbf{x}_i) \leq \mathsf{V}^{\{1\}}(\mathbf{x}_i)$, for $\mathbf{x}_i \in \mathbf{D}_2$.

Assuming that all hyperparameters are given, we now perform the stochastic co-kriging prediction of the expected high-fidelity response at a prediction point $\mathbf{x}_0$. Notice that standard results indicate that $\left(\mathsf{Y}^{\{2\}}(\mathbf{x}_0), \bar{\mathcal{Y}}^\top\right)^\top$ follow a multivariate normal distribution Kennedy and O'Hagan (2000), where $\bar{\mathcal{Y}} = \left(\bar{\mathcal{Y}}^{\{1\}}, \bar{\mathcal{Y}}^{\{2\}}\right)^\top$ and $\bar{\mathcal{Y}}^{\{i\}} = \left(\bar{\mathcal{Y}}^{\{i\}}(\mathbf{x}_1), \bar{\mathcal{Y}}^{\{i\}}(\mathbf{x}_2), \ldots, \bar{\mathcal{Y}}^{\{i\}}(\mathbf{x}_{k_i})\right)^\top$ for $i = 1, 2$. In particular, the conditional distribution of $\mathsf{Y}^{\{2\}}(\mathbf{x}_0)$ given $\bar{\mathcal{Y}}$ is also normal with the mean function given by

$$\widehat{\mathsf{Y}}^{\{2\}}(\mathbf{x}_0) = \mathbf{f}(\mathbf{x}_0)^\top \widehat{\boldsymbol{\beta}} + \mathbf{c}(\mathbf{x}_0)^\top \boldsymbol{\Sigma}^{-1}(\bar{\mathcal{Y}} - \mathbf{F}\widehat{\boldsymbol{\beta}}) \tag{3.6}$$

where $\mathbf{f}(\mathbf{x}_0)^\top = (\rho \mathbf{f}_1(\mathbf{x}_0)^\top, \mathbf{f}_2(\mathbf{x}_0)^\top)$, and

$$\mathbf{F} = \begin{bmatrix} \mathbf{f}_1(\mathbf{x}_1)^\top & 0 \\ \vdots & \vdots \\ \mathbf{f}_1(\mathbf{x}_{k_1})^\top & 0 \\ \rho \mathbf{f}_1(\mathbf{x}_1)^\top & \mathbf{f}_2(\mathbf{x}_1)^\top \\ \vdots & \vdots \\ \rho \mathbf{f}_1(\mathbf{x}_{k_2})^\top & \mathbf{f}_2(\mathbf{x}_{k_2})^\top \end{bmatrix}, \tag{3.7}$$

$$\widehat{\boldsymbol{\beta}} = \left(\widehat{\boldsymbol{\beta}}_1^\top, \widehat{\boldsymbol{\beta}}_2^\top\right)^\top = \left(\mathbf{F}^\top \boldsymbol{\Sigma}^{-1} \mathbf{F}\right)^{-1} \mathbf{F}^\top \boldsymbol{\Sigma}^{-1} \bar{\mathcal{Y}}, \tag{3.8}$$

and $\boldsymbol{\Sigma} = \boldsymbol{\Sigma}_{\mathsf{M}} + \boldsymbol{\Sigma}_\varepsilon$, and $\mathbf{c}(\mathbf{x}_0)$ denotes the following $(k_1 + k_2) \times 1$ covariance vector

$$\mathbf{c}(\mathbf{x}_0) = \left(\rho \tau_1^2 \mathcal{R}_1(\mathbf{D}_1, \mathbf{x}_0; \boldsymbol{\theta}_1)^\top \qquad \rho^2 \tau_1^2 \mathcal{R}_1(\mathbf{D}_2, \mathbf{x}_0; \boldsymbol{\theta}_1)^\top + \tau_\delta^2 \mathcal{R}_\delta(\mathbf{D}_2, \mathbf{x}_0; \boldsymbol{\theta}_\delta)^\top\right)^\top,$$

where $\mathcal{R}_1(\mathbf{D}_i, \mathbf{x}_0; \boldsymbol{\theta}_1)$ denotes the $k_i \times 1$ vector of spatial correlations between $\mathsf{Y}^{\{1\}}(\mathbf{x}_0)$ and $\mathsf{Y}^{\{1\}}(\mathbf{x}_\ell)$, for $\ell = 1, 2, \ldots, k_i$, $i = 1, 2$; $\mathcal{R}_\delta(\mathbf{D}_2, \mathbf{x}_0; \boldsymbol{\theta}_\delta)$ denotes the $k_2 \times 1$ vector of spatial correlations between $\mathsf{Y}^{\{2\}}(\mathbf{x}_0)$ and $\mathsf{Y}^{\{2\}}(\mathbf{x}_\ell)$ for $\ell = 1, 2, \ldots, k_2$. Furthermore, $\boldsymbol{\Sigma} = \boldsymbol{\Sigma}_{\mathsf{M}} + \boldsymbol{\Sigma}_\varepsilon$, and

$$\begin{aligned} \boldsymbol{\Sigma}_{\mathsf{M}} &= \begin{pmatrix} \Sigma_{\mathsf{M}}^{11} & \Sigma_{\mathsf{M}}^{12} \\ (\Sigma_{\mathsf{M}}^{12})^\top & \Sigma_{\mathsf{M}}^{22} \end{pmatrix} \\ &= \begin{pmatrix} \tau_1^2 \mathcal{R}_1(\mathbf{D}_1, \mathbf{D}_1; \boldsymbol{\theta}_1) & \rho \tau_1^2 \mathcal{R}_1(\mathbf{D}_1, \mathbf{D}_2; \boldsymbol{\theta}_1) \\ \rho \tau_1^2 \mathcal{R}_1(\mathbf{D}_1, \mathbf{D}_2; \boldsymbol{\theta}_1)^\top & \rho^2 \tau_1^2 \mathcal{R}_1(\mathbf{D}_2, \mathbf{D}_2; \boldsymbol{\theta}_1) + \tau_\delta^2 \mathcal{R}_\delta(\mathbf{D}_2, \mathbf{D}_2; \boldsymbol{\theta}_\delta) \end{pmatrix} \end{aligned}$$

Notice that the notation $R_1(\mathbf{D}_1, \mathbf{D}_2; \boldsymbol{\theta}_1)$ denotes the matrix of correlations between the values of $\mathsf{Y}^{\{1\}}(\cdot)$ at design points in $\mathbf{D}_1$ and $\mathbf{D}_2$, with its (i, j)th entry

given by $R_1(\mathbf{x}_i, \mathbf{x}_j; \boldsymbol{\theta}_1)$ for all $\mathbf{x}_i \in \mathbf{D}_1$ and $\mathbf{x}_j \in \mathbf{D}_2$. The other notation such as $R_1(\mathbf{D}_1, \mathbf{D}_1; \boldsymbol{\theta}_1)$ and $R_\delta(\mathbf{D}_2, \mathbf{D}_2; \boldsymbol{\theta}_\delta)$ is defined in a similar fashion.

The intrinsic variance-covariance matrix of $\bar{\mathcal{Y}}$ is

$$\boldsymbol{\Sigma}_\varepsilon = \begin{pmatrix} \Sigma_\varepsilon^{11} & \Sigma_\varepsilon^{12} \\ (\Sigma_\varepsilon^{12})^\top & \Sigma_\varepsilon^{22} \end{pmatrix},$$

where Σ_ε^{ii} is the $k_i \times k_i$ variance-covariance matrix of $\bar{\mathcal{Y}}^{\{i\}}$ for $i = 1, 2$; and Σ_ε^{12} is the $k_1 \times k_2$ covariance matrix of $\bar{\mathcal{Y}}^{\{1\}}$ and $\bar{\mathcal{Y}}^{\{2\}}$. Specifically,

$$\begin{aligned} \Sigma_\varepsilon^{11} &= \operatorname{diag}\left(\operatorname{Var}(\bar{\zeta}^{\{1\}}(\mathbf{x}_1)), \ldots, \operatorname{Var}(\bar{\zeta}^{\{1\}}(\mathbf{x}_{k_1}))\right) \\ &= \operatorname{diag}\left(\mathsf{V}^{\{1\}}(\mathbf{x}_1)/n_1^{\{1\}}, \ldots, \operatorname{Var}(\mathsf{V}^{\{1\}}(\mathbf{x}_{k_1})/n_{k_1}^{\{1\}})\right), \end{aligned}$$

$$\begin{aligned} \Sigma_\varepsilon^{22} &= \operatorname{diag}\left(\operatorname{Var}(\bar{\zeta}^{\{2\}}(\mathbf{x}_1)), \ldots, \operatorname{Var}(\bar{\zeta}^{\{2\}}(\mathbf{x}_{k_2}))\right) \\ &= \operatorname{diag}\left(\mathsf{V}^{\{2\}}(\mathbf{x}_1)/n_1^{\{2\}}, \ldots, \operatorname{Var}(\mathsf{V}^{\{2\}}(\mathbf{x}_{k_2})/n_{k_2}^{\{2\}})\right), \end{aligned}$$

$$\Sigma_\varepsilon^{12} = \begin{pmatrix} \operatorname{diag}(\operatorname{Cov}(\bar{\zeta}^{\{1\}}(\mathbf{x}_1), \bar{\zeta}^{\{2\}}(\mathbf{x}_1)), \ldots, \operatorname{Cov}(\bar{\zeta}^{\{1\}}(\mathbf{x}_{k_2}), \bar{\zeta}^{\{2\}}(\mathbf{x}_{k_2}))) \\ \mathbf{0}_{(k_1-k_2)\times k_2} \end{pmatrix},$$

where $\mathbf{0}_{(k_1-k_2)\times k_2}$ represents a $(k_1-k_2)\times k_2$ matrix of zeros, and for $i = 1, 2, \ldots, k_2$,

$$\operatorname{Cov}(\bar{\zeta}^{\{1\}}(\mathbf{x}_i), \bar{\zeta}^{\{2\}}(\mathbf{x}_i)) = \operatorname{Cov}(\zeta_j^{\{1\}}(\mathbf{x}_i), \zeta_j^{\{2\}}(\mathbf{x}_i))/\max\{n_i^{\{1\}}, n_i^{\{2\}}\}.$$

The conditional prediction variance follows as

$$\operatorname{Var}(\widehat{\mathsf{Y}}^{\{2\}}(\mathbf{x}_0)) = \tau_\delta^2 + \rho^2\tau_1^2 - \mathbf{c}(\mathbf{x}_0)^\top \boldsymbol{\Sigma}^{-1}\mathbf{c}(\mathbf{x}_0) + \eta(\mathbf{x}_0)^\top \left(\mathbf{F}^\top \boldsymbol{\Sigma}^{-1}\mathbf{F}\right)^{-1} \eta(\mathbf{x}_0), \quad (3.9)$$

where $\eta(\mathbf{x}_0) = \mathbf{f}(\mathbf{x}_0) - \mathbf{c}(\mathbf{x}_0)^\top \boldsymbol{\Sigma}^{-1}\mathbf{F}$.

We note that the predictor given in (3.6) can be used as a cheap approximation to the mean function value implied by high-fidelity simulation runs at a given prediction point $\mathbf{x}_0 \in \mathcal{X}$. Provided that high-fidelity simulations have been performed at enough design points, (3.6) should be more accurate than that given based on low-fidelity simulation runs. The conditional prediction variance (3.9) can be used to measure the prediction uncertainty associated with (3.6).

3.1.1 Estimating the Model Hyperparameters

Given that the parameter vector $\boldsymbol{\beta}$ is estimated by $\widehat{\boldsymbol{\beta}}$ given in (3.8), below we consider the estimation of model hyperparameters. As a result of the choice of design-point locations in $\mathbf{D}_1$ and those in $\mathbf{D}_2 \subset \mathbf{D}_1$ and the Markov property, we can estimate the parameters $(\tau_1^2, \boldsymbol{\theta}_1^\top)$ separately from $(\rho, \tau_\delta^2, \boldsymbol{\theta}_\delta^\top)$ following a similar argument as given by Kennedy and O'Hagan (2000).

Conditional on $(\tau_1^2, \boldsymbol{\theta}_1^\top)$, the distribution of $\bar{\mathcal{Y}}^{\{1\}}$ is normal and the log-likelihood of $\bar{\mathcal{Y}}^{\{1\}}$ can be written as

$$\begin{aligned} \ln \mathcal{L}(\tau_1^2, \boldsymbol{\theta}_1^\top) = -(&\frac{k_1}{2}\ln(2\pi) + \frac{1}{2}\ln(\det(\boldsymbol{\Sigma}^{11})) \\ &+ \frac{1}{2}(\bar{\mathcal{Y}}^{\{1\}} - \mathbf{F}_1\widehat{\boldsymbol{\beta}}_1)^\top(\boldsymbol{\Sigma}^{11})^{-1}(\bar{\mathcal{Y}}^{\{1\}} - \mathbf{F}_1\widehat{\boldsymbol{\beta}}_1)), \end{aligned} \tag{3.10}$$

where

$$\mathbf{F}_1 = \begin{bmatrix} \mathbf{f}_1(\mathbf{x}_1)^\top \\ \vdots \\ \mathbf{f}_1(\mathbf{x}_{k_1})^\top \end{bmatrix},$$

$\boldsymbol{\Sigma}^{11} = \tau_1^2\mathcal{R}_1(\mathbf{D}_1, \mathbf{D}_1; \boldsymbol{\theta}_1) + \Sigma_\varepsilon^{11}$. First, we need to estimate Σ_ε^{11} and replace it by its estimator in (3.10). Then estimates of τ_1^2 and $\boldsymbol{\theta}_1$ can be obtained by suitable optimization routines such as those available in Matlab.

We write the vector of differences between the two point estimators built on low- and high-fidelity simulation runs as

$$\widetilde{\delta} = \bar{\mathcal{Y}}^{\{2\}} - \rho\bar{\mathcal{Y}}^{\{1\}} = \left(\widetilde{\delta}(\mathbf{x}_1), \widetilde{\delta}(\mathbf{x}_2), \ldots, \widetilde{\delta}(\mathbf{x}_{k_2})\right)^\top.$$

It follows from (3.1) and (3.3) and the description given in section 3.1 that

$$\widetilde{\delta}(\mathbf{x}_i) = \bar{\mathcal{Y}}^{\{2\}}(\mathbf{x}_i) - \rho\bar{\mathcal{Y}}^{\{1\}}(\mathbf{x}_i) = \delta(\mathbf{x}_i) + \bar{\zeta}^{\{2\}}(\mathbf{x}_i) - \rho\bar{\zeta}^{\{1\}}(\mathbf{x}_i), \quad \text{for } \mathbf{x}_i \in \mathbf{D}_2.$$

Conditional on $(\rho, \tau_\delta^2, \boldsymbol{\theta}_\delta^\top)$, the distribution of $\widetilde{\delta}$ is normal and the log-likelihood of $\widetilde{\delta}$ can be written as

$$\begin{aligned} \ln \mathcal{L}(\rho, \tau_\delta^2, \theta_\delta^\top) = -(&\frac{k_2}{2}\ln(2\pi) + \frac{1}{2}\ln(\det(\boldsymbol{\Sigma}^{22})) \\ &+ \frac{1}{2}(\widetilde{\delta} - \mathbf{F}_\delta\widehat{\boldsymbol{\beta}}_2)^\top(\boldsymbol{\Sigma}^{22})^{-1}(\widetilde{\delta} - \mathbf{F}_\delta\widehat{\boldsymbol{\beta}}_2)), \end{aligned} \tag{3.11}$$

where

$$\mathbf{F}_\delta = \begin{bmatrix} \mathbf{f}_2(\mathbf{x}_1)^\top \\ \vdots \\ \mathbf{f}_2(\mathbf{x}_{k_2})^\top \end{bmatrix},$$

$\boldsymbol{\Sigma}^{22} = \tau_\delta^2 \mathcal{R}_\delta(\mathbf{D}_2, \mathbf{D}_2; \boldsymbol{\theta}_\delta) + \boldsymbol{\Sigma}_\varepsilon^{\widetilde{\delta}}$, with $\boldsymbol{\Sigma}_\varepsilon^{\widetilde{\delta}} = \text{diag}\left(\text{Var}(\bar{\zeta}^{\{2\}}(\mathbf{x}_1) - \rho\bar{\zeta}^{\{1\}}(\mathbf{x}_1)), \ldots, \text{Var}(\bar{\zeta}^{\{2\}}(\mathbf{x}_{k_2}) - \rho\bar{\zeta}^{\{1\}}(\mathbf{x}_{k_2}))\right)$, and for $\mathbf{x}_i \in D_2$,

$$\begin{aligned}
\text{Var}(\bar{\zeta}^{\{2\}}(\mathbf{x}_i) - \rho\bar{\zeta}^{\{1\}}(\mathbf{x}_i)) &= \text{Var}(\bar{\zeta}^{\{2\}}(\mathbf{x}_i)) + \rho^2\text{Var}(\bar{\zeta}^{\{1\}}(\mathbf{x}_i)) \\
&\quad - 2\rho\text{Cov}(\bar{\zeta}^{\{2\}}(\mathbf{x}_i), \bar{\zeta}^{\{1\}}(\mathbf{x}_i)) \\
&= \boldsymbol{\Sigma}_\varepsilon^{22}(i,i) + \rho^2\boldsymbol{\Sigma}_\varepsilon^{11}(i,i) - 2\rho\boldsymbol{\Sigma}_\varepsilon^{12}(i,i) \\
&= \mathsf{V}^{\{2\}}(\mathbf{x}_i)/n_i^{\{2\}} + \rho^2\mathsf{V}^{\{1\}}(\mathbf{x}_i)/n_i^{\{1\}} \\
&\quad - 2\rho\text{Cov}(\zeta_j^{\{1\}}(\mathbf{x}_i), \zeta_j^{\{2\}}(\mathbf{x}_i))/\max\{n_i^{\{1\}}, n_i^{\{2\}}\}.
\end{aligned}$$

First, we need to estimate $\boldsymbol{\Sigma}_\varepsilon^{\widetilde{\delta}}$ in $\boldsymbol{\Sigma}^{22}$ and replace it by its estimator in (3.11). Estimates of ρ, τ_δ^2 and $\boldsymbol{\theta}_2$ can be obtained by suitable optimization routines subsequently.

3.1.2 Covariance Estmation for Steady-State Simulation

In this section we review a small selection of methods for steady-state variance estimation. These methods facilitate the application of SCK using outputs from within a single high-fidelity simulation replication. We will concentrate on discrete-time processes (continuous-time processes can be handled in a similar manner). Recall that at each low-fidelity design point $\mathbf{x}_i$ in $\mathbf{D}_1$, we run $n_i^{\{1\}}$ independent simulation replications and generate i.i.d. outputs $\{\mathcal{Y}_j^{\{1\}}(\mathbf{x}_i)\}_{j=1}^{n_i^{\{1\}}}$. The output $\mathcal{Y}_j^{\{1\}}(\mathbf{x}_i)$ generated on the jth simulation replication is considered as the sample mean of $s_i^{\{1\}}$ basic outputs, i.e., $\mathcal{Y}_1^{\{1\}}(\mathbf{x}_i) = \sum_{t=1}^{s_i^{\{1\}}} Y_t(\mathbf{x}_i)/s_i^{\{1\}}$. On the other hand, at each high-fidelity design point $\mathbf{x}_i \in \mathbf{D}_2$, a single simulation replication is performed with a runlength much longer than that of a low-fidelity simulation replication, i.e., $s_i^{\{2\}} >> s_i^{\{1\}}$, and produces a single point estimate $\mathcal{Y}_1^{\{2\}}(\mathbf{x}_i) = \sum_{t=1}^{s_i^{\{2\}}} Y_t(\mathbf{x}_i)/s_i^{\{2\}}$, the sample mean of $s_i^{\{2\}}$ basic outputs at $\mathbf{x}_i$.

Given a single long simulation replication at each high-fidelity design point $\mathbf{x}_i \in \mathbf{D}_2$, we next consider estimating the variance of the sample mean, $\mathsf{V}^{\{2\}}(\mathbf{x}_i)$, via some selected correlation-based methods; see details from, for example, Alexopoulos and Goldsman (2004), Goldsman and Nelson (2006), Alexopoulos et al.

(2007b) and Alexopoulos et al. (2007a). For ease of exposition, we omit the design point $\mathbf{x}_i$ from our notation and further denote $\mathsf{V}^{\{2\}}(\mathbf{x}_i)$ by V.

Nonoverlapping Batch Mean Variance Estimator (NBM)

Suppose that each high fidelity run has a runlength of $s^{\{2\}} = mb$, and the simulation outputs, $Y_1, Y_2, \ldots, Y_{s^{\{2\}}}$, can be divided into b contiguous, nonoverlapping batches of outputs, each of batch size m. That is, the ith batch is consisted of observations $Y_{(i-1)m+1}, Y_{(i-1)m+2}, \ldots, Y_{im}$, for $i = 1, 2, \ldots, b$. The NBM estimator for V is given by

$$\widehat{\mathsf{V}}^{\{2\}}_{\mathrm{NBM}} = \frac{m}{(b-1)s^{\{2\}}} \sum_{i=1}^{b} \left(\bar{Y}_{i,m} - \bar{Y}_{s^{\{2\}}}\right)^2,$$

where $\bar{Y}_{i,m} = m^{-1} \sum_{\ell=1}^{m} Y_{(i-1)m+\ell}$ for $i = 1, 2, \ldots, b$; and $\bar{Y}_{s^{\{2\}}} = \sum_{i=1}^{s^{\{2\}}} Y_i / s^{\{2\}}$.

The next two variance estimators are constructed from the following standardized time series (STS) based on the ith nonoverlapping batch of size m,

$$T_{i,m}(t) = \frac{\lfloor mt \rfloor \left(\bar{Y}_{i,m} - \bar{Y}_{i,\lfloor mt \rfloor}\right)}{\sqrt{\mathsf{V} m}}, \quad \text{for } t \in [0, 1],$$

where $\lfloor \cdot \rfloor$ denotes the floor function and $\bar{Y}_{i,j} = j^{-1} \sum_{\ell=1}^{j} Y_{(i-1)m+\ell}$ denotes the jth cumulative sample mean for $j = 1, 2, \ldots, m$ from the ith batch, $i = 1, 2, \ldots, b$.

Nonoverlapping Batched Area Variance Estimator (NA)

We denote $A_i(f; m)$ as the weighted area estimator computed under the STS from the ith nonoverlapping batch,

$$A_i(f; m) = \left[m^{-1} \sum_{\ell=1}^{m} f\left(\frac{\ell}{m}\right) \mathsf{V}^{\frac{1}{2}} T_{i,m}\left(\frac{\ell}{m}\right) \right]^2, \quad \text{for } i = 1, 2, \ldots, b.$$

where $f(\cdot)$ is a weighting function and we adopt $f(t) = \sqrt{840}(3t^2 - 3t + 0.5)$ for $t \in [0, 1]$ in Section 4.1 for numerical evaluations; see other weighting functions

from, for instance, Goldsman and Nelson (2006). The NA estimator follows as

$$\widehat{\mathsf{V}}_{\mathrm{NA}}^{\{2\}} = \frac{1}{bs^{\{2\}}} \sum_{i=1}^{b} A_i(f;m).$$

Nonoverlapping Batched Weighted Cramér-von Mises Estimator (NCvM)

We denote $C_i(g;m)$ as the weighted area under the STS from the ith nonoverlapping batch,

$$C_i(g;m) = m^{-1} \sum_{\ell=1}^{m} g\left(\frac{\ell}{m}\right) \mathsf{V}T_{i,m}^2\left(\frac{\ell}{m}\right), \quad \text{for } i = 1,2,\ldots,b.$$

where $g(\cdot)$ is a weighting function and we adopt $g(t) = -24 + 150t - 150t^2$ for $t \in [0,1]$ in Section 4.1 for numerical evaluations; other weight functions can be found from, for example, Goldsman and Nelson (2006). The NCvM estimator is given by

$$\widehat{\mathsf{V}}_{\mathrm{NCvM}}^{\{2\}} = \frac{1}{bs^{\{2\}}} \sum_{i=1}^{b} C_i(g;m).$$

Overlapping Batch Mean Variance Estimator (OBM)

Suppose that we divide $Y_1, Y_2, \ldots, Y_{n_i^{\{2\}}}$ into $s^{\{2\}} - m + 1$ overlapping batches, each of size m. That is, the observations $Y_1, Y_2, \ldots, Y_m$ comprise the 1st batch, and $Y_2, Y_3, \ldots, Y_{m+1}$ form the 2nd batch, and so on. In general, $Y_i, Y_{i+1}, \ldots, Y_{i+m-1}$ form the ith batch, for $i = 1, 2, \ldots, n_i^{\{2\}} - m + 1$. The OBM estimator can be given by

$$\widehat{\mathsf{V}}_{\mathrm{OBM}}^{\{2\}} = \frac{s^{\{2\}} m}{(s^{\{2\}} - m + 1)(s^{\{2\}} - m)s^{\{2\}}} \sum_{i=1}^{s^{\{2\}}-m+1} \left(\bar{Y}_{i,m}^{O} - \bar{Y}_{s^{\{2\}}}\right)^2,$$

where $\bar{Y}_{i,m}^{O} = m^{-1} \sum_{\ell=0}^{m-1} Y_{i+\ell}$ for $i = 1, 2, \ldots, s^{\{2\}} - m + 1$, and $\bar{Y}_{s^{\{2\}}} = \sum_{i=1}^{s^{\{2\}}} Y_i / s^{\{2\}}$.

The next two variance estimators are constructed from the following STS based on the ith overlapping batch of size m,

$$T_{i,m}^{O}(t) = \frac{\lfloor mt \rfloor \left(\bar{Y}_{i,m}^{O} - \bar{Y}_{i,\lfloor mt \rfloor}^{O} \right)}{\sqrt{\mathsf{V} m}}, \quad \text{for } t \in [0,1],$$

for $i = 1, 2, \ldots, s^{\{2\}} - m + 1$, where $\bar{Y}_{i,j}^{O} = j^{-1} \sum_{\ell=0}^{j-1} Y_{i+\ell}$ for $i = 1, 2, \ldots, s^{\{2\}} - m + 1$ and $j = 1, 2, \ldots, m$.

Overlapping Batched Area Variance Estimator (OA)

We denote $A_i^O(f; m)$ as the weighted area estimator computed under the STS from the ith overlapping batch,

$$A_i^O(f; m) = \left[m^{-1} \sum_{\ell=1}^{m} f\left(\frac{\ell}{m}\right) \mathsf{V}^{\frac{1}{2}} T_{i,m}^{O}\left(\frac{\ell}{m}\right) \right]^2, \quad \text{for } i = 1, 2, \ldots, s^{\{2\}} - m + 1,$$

where the weighting function f is the same as described for the NA estimator. The OA estimator then follows as

$$\widehat{\mathsf{V}}_{\mathrm{OA}}^{\{2\}} = \frac{1}{(s^{\{2\}} - m + 1) s^{\{2\}}} \sum_{i=1}^{s^{\{2\}} - m + 1} A_i^O(f; m).$$

Nonoverlapping Batched Weighted Cramér-von Mises Estimator (OCvM)

We denote $C_i^O(g; m)$ as the weighted area under the STS from the ith overlapping batch,

$$C_i^O(g; m) = m^{-1} \sum_{\ell=1}^{m} g\left(\frac{\ell}{m}\right) \left[\mathsf{V}^{\frac{1}{2}} T_{i,m}^{O}\left(\frac{\ell}{m}\right) \right]^2, \quad \text{for } i = 1, 2, \ldots, s^{\{2\}} - m + 1.$$

where the weighting function g is the same as described for the NCvM estimator. The OCvM estimator then follows as

$$\widehat{\mathsf{V}}_{\mathrm{OCvM}}^{\{2\}} = \frac{1}{(s^{\{2\}} - m + 1) s^{\{2\}}} \sum_{i=1}^{s^{\{2\}} - m + 1} C_i^O(g; m).$$

3.2 Design of Experiment

One of the most important questions to be answered in this research problem, is which points should be chosen to build the low-fidelity data set? How long should the simulate code run for? There is also a more critical inquiry regarding the data set. How to choose a subset of low-fidelity samples to construct the high fidelity data set? and how long more to simulate the system at these points to get statistically better and more accurate responses?

The simplest way to proceed is to divide the available budget among evenly spread sample points. In the empirical example in chapter 4, M/M/1 queueing system will be studied. The desired range for utilization of servers is between 0.5 and 0.91. Since the service rate (λ) and number of servers remain constant over the time, the independent variable under study is service rate (μ) to the system. The purpose is to see how average waiting time in the system for customers change, for different values of service rate.

Assume that number of design points in low fidelity and high fidelity data set is k_1 and k_2, respectively. Number of replications through all the design points for each level of data sets is equal and considered to be n_1 and n_2 for low fidelity and high fidelity data set respectively. Moreover, run length of the rough and detailed simulation codes is represented with T_1 and T_2 . So, the total time budget spent on this experiment is $T = \sum_{i=1}^{2}(\sum_{j=1}^{k_i}(n_i k_i T_i))$. With the total budget to be spent on the experiments in hand, the experimenter can choose the design points to run the simulation code at. In this research evenly-spaced design is used. This design deals with maximizing the minimum distance between design points. Selection of low fidelity design points, is in a continuous space. However, for high fidelity design points, since they are a subset of low fidelity points, the selection space is discrete, yet the idea is the same.

Chapter 4

EMPIRICAL STUDIES

In this chapter, we will evaluate the performance of our proposed stochastic co-kriging model in comparison to stochastic kriging model for M/M/1 and M/M/S queueing systems. Two different approaches were taken in this analysis. First, high fidelity simulation runs were designed to have replications at each design point. Then, since our assumption is that high fidelity simulations are costly, the analysis were exerted on single replication of high fidelity simulations.

4.1 An M/M/1 Queueing System

Consider simulating an M/M/1 queue with arrival rate 1 per time unit and service rate x per time unit with $x \in \mathcal{X} = [1.1, 2]$. It is well known from queueing theory that the mean steady-state waiting time in the queue is $\mathsf{Y}(x) = 1/\big(x(x-1)\big)$ Whitt (1989), which is the function we intend to estimate. For each simulation experiment, a set of k equispaced design points are chosen from $\mathcal{X}$, with $x_1 = 1.1$ and $x_k = 2$. Each simulation replication (run) is *initialized either in empty state or steady state*, and the runlength T is specified by the number of simulated customers. We note that the value of T here determines the steady-state simulation fidelity level. The simulation output on a given replication is the sample-path average waiting time of the T customers simulated.

4.1.1 Experiment Setup

Two-fidelity-level simulation experiment. We consider two sets of two-fidelity-level simulation experiment, which share the common low-fidelity simulation runs and only differ in the high-fidelity simulation runs. The low-fidelity design-point set $\mathbf{D}_1$ consists of a grid of 25 equidistant design points in $[1.1, 2]$ with $x_1 = 1.1$ and $x_{25} = 2$. At each design point in $\mathbf{D}_1$, $n^{\{1\}} = 10$ simulation replications are applied and the runlength of each replication is 5000. The high-fidelity design-point set $\mathbf{D}_2 \subset \mathbf{D}_1$ consists of 4 design points, i.e., $\mathbf{D}_2 = \{x_1, x_9, x_{17}, x_{25}\}$. We consider the following two types of high-fidelity simulation runs:

1. **High-fidelity simulation with multiple replications:** At each design point in $\mathbf{D}_2$, $n^{\{2\}} = 4$ replications are applied with each replication having a runlength of $275,000$. Therefore, the simulation budget expended at each high fidelity design point is 1.1×10^6.

2. **High-fidelity simulation with a single replication :** At each design point in $\mathbf{D}_2$, a single simulation replication is applied with a runlength of 1.1×10^6.

Despite the difference in the two sets of high-fidelity simulation runs, we note that the resulting total simulation budget for the above two sets of two-fidelity-level simulation experiment stays the same which is 5.65×10^6.

Single-fidelity-level simulation experiment. We consider conducting a single-fidelity-level simulation experiment at the same set of design points as those in $\mathbf{D}_1$. Specifically, at each design point in $\mathbf{D}_1$, $n = 10$ replications are applied with each replication having a runlength of $22,600$. The total simulation budget is the same as that of the two sets of two-fidelity-level simulation experiment.

We consider the following metamodeling methods and compare their predictive performance: (1) stochastic kriging applied with the single-fidelity-level simulation experiment (**SK-1L**), (2) stochastic co-kriging applied with the two-fidelity simulation experiment in which high-fidelity simulations are replicated (**SCK-mH**), and (3) stochastic co-kriging applied with the two-fidelity simulation experiment in which a single high-fidelity simulation replication is used (**SCK-sH**). For implementing SCK-sH, we use the methods reviewed in Subsection 3.1.2 for

estimating the variance of a point estimate using the individual waiting times generated from within a single long replication. Notice that for the sake of brevity, we omit the results obtained by SCK with OCvM and OA applied. An important decision in this context is to determine the batch size m to use for variance estimation. For discussions of appropriate batch sizes to use, see Nelson (2011), Song and Schmeiser (1995) and Song (1996), to name a few. In our implementation, we set the ratio of the runlength to the batch size $b = s^{\{2\}}/m$ to 20, 50, and 110 which corresponds to $m = 55,000$, $22,000$, $10,000$, respectively.

A grid of $K = 193$ equispaced check-points are chosen from $\mathcal{X}$ to evaluate predictive performance of stochastic co-kriging (SCK) and stochastic kriging (SK). The aforementioned two-fidelity-level and single-fidelity-level experiments are respectively executed for 100 independent macro-replications, and the predictive performance measure, the empirical root mean squared errors (ERMSE), is calculated as follows,

$$\text{ERMSE}_\ell = \sqrt{\frac{1}{K}\sum_{i=1}^{K}\left(\widehat{\mathsf{Y}}_\ell(x_i) - \mathsf{Y}(x_i)\right)^2}, \quad \ell = 1, 2, \ldots, 100, \tag{4.1}$$

where $\widehat{\mathsf{Y}}_\ell(\cdot)$ represents the prediction given by SCK or SK on the ℓth macro-replication.

4.1.2 Summary of Results for M/M/1

The ERMSEs obtained by SCK and SK from 100 macro-replications are summarized in Table 4.1. The value in each cell of Table 4.1 is the average ERMSE obtained across the 100 macro-replications; and the value in parentheses is the corresponding standard error. We observe that regardless of the initializing condition, SCK-mH outperforms SCK-sH and SK-1L and SK-1L performs the worst. The performances achieved by SCK-sH with different batch means methods applied is close to one another, and the ERMSEs obtained are relatively stable as the batch size increases from 10,000 to 55,000. In terms of experimental design for the two-level-fidelity simulation experiment, we observe that while keeping the low-fidelity simulation runs fixed, SCK seems to work better with a few moderately long simulation replications as compared to a single long simulation replication;

and a lack of replications at high-fidelity design points may lead to loss of predictive accuracy achieved by SCK. Lastly, initializing a simulation run at steady state does not seem to make a significant impact on the performance achieved by SCK as opposed to initializing in empty state.

TABLE 4.1: Results for the M/M/1 queueing example.

Initialization	SK-1L	SCK-mH	SCK-sH + batch mean methods			
			batch size	NBM	NA	NCvM	OBM
Empty state	0.48 (0.01)	0.39 (0.01)	10,000	0.41 (0.02)	0.41 (0.02)	0.41 (0.02)	0.40 (0.02)
			22,000	0.41 (0.02)	0.41 (0.02)	0.41 (0.02)	0.40 (0.02)
			55,000	0.41 (0.02)	0.41 (0.02)	0.41 (0.02)	0.404 (0.02)
			batch size	NBM	NA	NCvM	OBM
Steady state	0.49 (0.01)	0.38 (0.01)	10,000	0.40 (0.01)	0.41 (0.01)	0.41 (0.01)	0.40 (0.01)
			22,000	0.40 (0.01)	0.41 (0.01)	0.41 (0.01)	0.40 (0.01)
			55,000	0.40 (0.01)	0.41 (0.01)	0.41 (0.01)	0.40 (0.01)

4.2 An M/M/5 Queueing System

Consider the following M/M/5 infinite capacity queue, with a single waiting line. Customers arrive according to a Poisson process with a variable arrival rate x per time unit in $[2.5, 4.5]$ and the the service times are assumed to be i.i.d. exponential with constant service rate $\mu = 1$ per time unit. The goal is to estimate the mean steady-state waiting time for a customer in the queue $\mathsf{Y}(x)$ as a function of the arrival rate x. Queueing theory gives us the following expression of steady state mean waiting time for an $M/M/s$ queue (in our case $s = 5$):

$$\mathsf{Y}(x) = \frac{1}{x}\left[\frac{x}{\mu} + \left(\frac{\left(\frac{x}{\mu}\right)^{s+1}}{(s-1)!\left(s-\frac{x}{\mu}\right)^2}\right) P_0(x)\right] \tag{4.2}$$

where

$$P_0(x) = \left(\sum_{i=0}^{s-1} \frac{x^i}{i!\mu^i} + \frac{x^s}{s!\mu^s}\frac{s\mu}{s\mu - x}\right)^{-1} \tag{4.3}$$

To make sure that the queue is stable, we consider $x \in [2.5, 4.5]$ such that $\rho = x/(5\mu) \in [0.5, 0.9]$ in this example. Each of the replications starts in the empty state. We consider the following experimental design of the problem.

4.2.1 Experiment Setup

Two-fidelity-level simulation experiment. The low fidelity design-point set D_1 consists of a grid of 25 equidistant design points in $[2.5, 4.5]$ with $x_1 = 2.5$ and $x_{25} = 4.5$. At each design point in D_1, $n^{\{1\}} = 10$ replications are applied with each replication having a run length of 5000 time units. The high fidelity design-point set $D_2 \subset D_1$ and $D_2 = \{x_1, x_9, x_{17}, x_{25}\}$.

1. **High-fidelity simulation with multiple replications:** At each design point in D_2, $n^{\{2\}} = 5$ replications are applied with each replication having a run length of 187500 time units. Therefore, the total simulation budget is 5×10^6 time units.

2. **High-fidelity simulation with a single replication :** At each design point in D_2, $n^{\{2\}} = 1$ replication is applied with each replication having a run length of 937500 time units.

Despite the difference in the two sets of high-fidelity simulation runs, we note that the resulting total simulation budget for the above two sets of two-fidelity-level simulation experiment stays the same which is 5×10^6.

Single-fidelity-level simulation experiment. The design set D consists of a grid of 25 equidistant design points in [2.5, 4.5] with $x_1 = 2.5$ and $x_{25} = 4.5$. At each design point in D, $n = 10$ replications are applied with each replication having a run length of 2×10^4time units. The total simulation budget stays the same as in the two fidelity level simulation.

As it was mentioned in 4.1.1, we consider three metamodeling methods and compare their predictive performances: (1) stochastic kriging applied with the single-fidelity-level simulation experiment (**SK-1L**), (2) stochastic co-kriging applied with the two-fidelity simulation experiment in which high-fidelity simulations are replicated (**SCK-mH**), and (3) stochastic co-kriging applied with the two-fidelity simulation experiment where only one high-fidelity simulation replication is used (**SCK-sH**).

4.2.2 Summary of Results for M/M/5

The aforementioned experiment is repeated for 30 independent macro-replications, and the corresponding performance measure, the empirical root mean squared errors (ERMSE), is calculated as follows,

$$\text{ERMSE}_\ell = \sqrt{\frac{1}{K}\sum_{i=1}^{K}\left(\widehat{\mathsf{Y}}_\ell(x_i) - \mathsf{Y}(x_i)\right)^2}, \quad \ell = 1, 2, \ldots, 30, \tag{4.4}$$

where $\widehat{\mathsf{Y}}_\ell(\cdot)$ represents the prediction given by SCK or SK on the ℓth macro-replication; K represents the number of prediction points and $K = 193$. We will compare the performance of SCK and SK by mean and variance of their respective ERMSEs, for 30 macro replications when there are no replications in Table 4.2. For comparison of SCK with replicated simulations to SK, we will use the data from 100 macro replications.

TABLE 4.2: Results for the M/M/5 queueing example.

Initialization	SK-1L	SCK-mH	SCK-sH + batch mean methods				
			b	NBM	NA	NCvM	OBM
Empty state	0.0328 (0.0001)	0.0322 (0.0001)	20	0.0328 (0.0001)	0.0330 (0.0001)	0.0327 (0.0001)	0.0327 (0.0001)
			50	0.0326 (0.0001)	0.0332 (0.0001)	0.0333 (0.0001)	0.0327 (0.0001)

The predictive performance of SK-1L and SCK-sH is summarized by mean of ERMSEs given in Table 4.2. In this table, b is defined such that $s^{\{2\}} = mb$ where $s^{\{2\}}$ shows each high fidelity run runlength and m is batch size.Our findings from this work show that SCK-sH method works better than SK-1L method for fitting response surfaces, in the cases that there are no replicated high fidelity data, given that the variance is estimated using NBM or OBM method. However, using NA and NCvM variance estimation methods, SCK-sH predictive accuracy has decreased. Speciifically, NA variance estimation method has increased the mean ERMSE to values greater than that of SK-1L. Using NCvM variance estimation method, however, SCK-sH has worse results than SK-1L only when number of batches is 20. In this example, since the simulation run length is based on time units, not number of simulated completed services, different number of arrivals has been recorded as a result of simulation, meaning that different number of waiting times at the end of each macro-replication is recorded. So, instead of using fixed batch size for variance estimation, fixed number of batches is used. As

it is discussed here, careful selection of number of batches to condcuct variance estimation is critical to performance of SCK-sH.

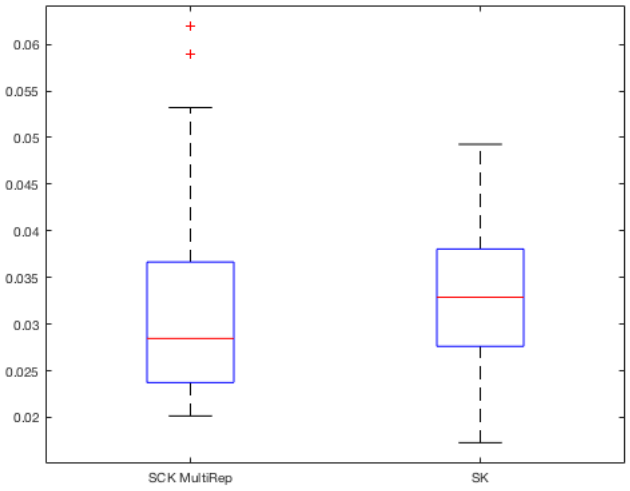

FIGURE 4.1: ERMSEs obtained by SK-1L and SCK-mH over 100 macro-replications

To be able to fairly compare the predictive accuracy of SCK-mH and SK-1L, 100 independent macro-replications of M/M/5 simulation are used. The results for 100 independent macro-replications show that ERMSE for SK-1L has a mean of 0.0335 and variance of 0.0001, where as the ERMSE for SCK-mH has a mean of 0.0325 and a variance of 0.0001. This result, which can be seen in figure 4.1, indicate that SCK-mH method, has a better predictive capability compared to SK-1L.

Chapter 5

CONCLUSIONS

In summary, we have presented the stochastic co-kriging methodology (SCK) for approximating an steady-state mean response surface based on outputs from both long and short simulation replications performed at selected design points. We have provided details on how to construct an SCK metamodel, perform parameter estimation, and make prediction via SCK. From a deign of simulation experiments perspective, metamodels reduce the computational cost of exploring large regions of the design space by replacing replicated long simulations required to obtain accurate steady-state mean parameter estimates. However, it is well known that a substantial computational effort is involved in performing long steady-state simulations to build metamodels. Using SCK proposed in this paper, with the same computational effort expended, it is possible to improve the accuracy of the metamodels obtained from the relatively short simulation replications, by supplementing the outputs from these simulations with outputs from long simulation replications performed at only a few design points. Therefore, it is possible to explore a design space with enhanced metamodels that are more accurate than metamodels based entirely on short simulation replications but less computationally expensive than metamodels based exclusively on long simulation replications.

We have shown the promise of using SCK for approximating a mean response surface using simulation runs performed to two levels of fidelity, in presence of replicated data at high fidelity design points. This method can be extendable to multiple levels and there exist many other types of wisdom that may be incorporated into simulation experimental designs for SCK, such as approximation

results from queueing theory (e.g., Whitt (1989) and Whitt (2006)). Future research topics include investigating design-point sets for performing simulations with different levels of fidelity and seeking suitable simulation budget allocation rules when a fixed computational budget is given.

REFERENCES

Alexopoulos, C., Argon, N. T., Goldsman, D., Steiger, N. M., Tokol, G., and Wilson, J. R. (2007a). Efficient computation of overlapping variance estiamtors for simulation. *INFORMS Journal on Computing*, 29:314–327.

Alexopoulos, C., Argon, N. T., Goldsman, D., Tokol, G., and Wilson, J. R. (2007b). Overlapping variance estimators for simulation. *Operations Research*, 55(6):1090–1103.

Alexopoulos, C. and Goldsman, D. (2004). To batch or not to batch? *ACM Transactions on Modeling and Computer Simulation*, 14:76–114.

Ankenman, B. E., Nelson, B. L., and Staum, J. (2010). Stochastic kriging for simulation metamodeling. *Operations Research*, 58:371–382.

Argon, N. T., Andradóttir, S., Alexopoulos, C., and Goldsman, D. (2013). Steady-state simulation with replication-dependent initial transients: analysis and examples. *INFORMS Journal on Computing*, 25:177–191.

Bekki, J. M., Chen, X., and Batur, D. (2014). Steady-state quantile parameter estimation: an empirical comparison of stochastic kriging and quantile regression. In Tolk, A., Diallo, S. D., Ryzhov, I. O., Yilmaz, L., Buckley, S., and Miller, J. A., editors, *Proceedings of the 2014 Winter Simulation Conference*, pages 3880–3891, Piscataway, New Jersey. Institute of Electrical and Electronics Engineers, Inc.

Chen, X. and Kim, K.-K. (2014). Stochastic kriging with biased sample estimates. *ACM Transactions on Modeling and Computer Simulation*, 24:8/1–8/23.

Cressie, N. A. C. (2015). *Statistics for Spatial Data*. John Wiley & Sons, New Jersey, revised edition.

Forrester, A. I. J., Sóbester, A., and Keane, A. J. (2007). Multi-fidelity optimization via surrogate modelling. In Tew, J. D., Manivannan, S., Sadowski, D. A., and Seila, A. F., editors, *Proceedings of the Royal Society A*, pages 3251–3269.

Goldsman, D. and Nelson, B. L. (2006). Correlation-based methods for output analysis. In Henderson, S. G. and Nelson, B. L., editors, *Simulation*, Handbooks in Operations Research and Management Science Vol. 13, pages 455–475. Elsevier.

Grassmann, W. (2016). *Multiple runs in the simulation of stochastic systems can improve the estimates of equilibrium expectations*, pages 35–48. Springer International Publishing, Cham.

Gratiet, L. L. and Cannamela, C. (2015). Cokriging-based sequential design strategies using fast cross-validation techniques for multi-fidelity computer codes. *Technometrics*, 57:418–427.

Jeong, H.-D. J., Lee, J.-S. R., McNickle, D., and Pawlikowski, K. (2005). Distributed steady-state simulation of telecommunication networks with self-similar teletraffic. *Simulation Modeling Practice and Theory*, 13:233–256.

Jones, D. R., Schonlau, M., and Welch, W. J. (1998). Efficient global optimization of expensive black-box functions. *Journal of Global Optimization*, 13:455–492.

Kelton, W. D. (1986). Replication splitting and variance for simulating discrete-parameter stochastic processes. *Operations Research Letters*, 4:275–279.

Kennedy, M. C. and O'Hagan, A. (2000). Predicting the output from a complex computer code when fast approximations are available. *Biometrika*, 87:1–13.

Kennedy, M. C. and O'Hagan, A. (2001). Bayesian calibration of computer models. *Journal of Royal Statistical Society*, 63:425–464.

Morris, M. and Mitchell, T. (1995). Exploratory designs for computer experiments. *Journal of Statistical Planning and Inference*, 43:381–402.

Morris, M. D., Mitchell, T. J., and Ylvisaker, D. (1993). Bayesian design and analysis of computer experiments: Use of derivatives in surface prediction. *Technometrics*, 35:243–255.

Nelson, B. L. (2011). Thirty years of 'batch size effects'. In Jain, S., Creasey, R. R., Himmelspach, J., White, K. P., and Fu, M., editors, *Proceedings of the*

2011 Winter Simulation Conference, pages 393–400, Piscataway, New Jersey. Institute of Electrical and Electronics Engineers, Inc.

Ni, E. C. and Henderson, S. G. (2015). How hard are steady-state queueing simulations? *ACM Transactions on Modeling and Computer Simulation (TOMACS)*, 25(4):27.

Pawlikowski, K. (1990). Steady-state simulation of queueing processes: a survey of problems and solutions. *ACM Computing Surveys*, 22:123–170.

Qian, P. Z. G. and Wu, C. F. J. (2008). Bayesian hierarchical modeling for integrating low-accuracy and high-accuracy experiments. *Technometrics*, 50:192–204.

Qian, Z. G. P., Wu, H. Q., and Wu, C. F. J. (2008). Gaussian process models for computer experiments with qualitative and quantitative factors. *Technometrics*, 50:192–204.

Rasmussen, C. E. and Williams, C. K. I. (2006). *Gaussian Processes for Machine Learning*. MIT, Cambridge.

Santner, T. J., Williams, B. J., and Notz, W. I. (2003). *The Design and Analysis of Computer Experiments*. Springer, New York.

Song, W. T. (1996). On the estimation of optimal batch sizes in the analysis of simulation output. *European Journal of Operational Research*, 88(2):304–319.

Song, W. T. and Schmeiser, B. W. (1995). Optimal mean-squared-error batch sizes. *Management Science*, 41:110–123.

Whitt, W. (1989). Planning queueing simulations. *Management Science*, 35:1341–1366.

Whitt, W. (1991). The efficiency of one long run versus independent replications in steady-state simulation. *Management Science*, 37:645–666.

Whitt, W. (2006). Analysis for design. In Henderson, S. G. and Nelson, B. L., editors, *Simulation*, Handbooks in Operations Research and Management Science Vol. 13, pages 381–413. Elsevier.

Yang, F., Ankenman, B. E., and Nelson, B. L. (2008). Estimating cycle time percentile curves for manufacutring systems via simulation. *INFORMS Journal on Computing*, 20:628—-643.