Proceedings of the 2018 Winter Simulation Conference
M. Rabe, A. A. Juan, N. Mustafee, A. Skoogh, S. Jain, and B. Johansson, eds.

UNBIASED METAMODELING VIA LIKELIHOOD RATIOS

Jing Dong

Graduate School of Business
Columbia University
New York, NY 10027, USA

M. Ben Feng

Deparment of Statistics and Actuarial Science
University of Waterloo
Waterloo, Ontario, CANADA

Barry L. Nelson

Department of Industrial Engineering and Management Sciences
Northwestern University
Evanston, IL 60208, USA

ABSTRACT

Metamodeling has been a topic of longstanding interest in stochastic simulation because of the usefulness of metamodels for optimization, sensitivity, and real- or near-real-time decision making. Experiment design is the foundation of classical metamodeling: an effective experiment design uncovers the spatial relationships among the design/decision variables and the simulation response; therefore, more design points, providing better coverage of space, is almost always better. However, metamodeling based on likelihood ratios (LRs) turns the design question on its head: *each* design point provides an unbiased prediction of the response at *any* other location in space, but perhaps with such inflated variance as to be counterproductive. Thus, the question becomes more which design points to employ for prediction and less where to place them. In this paper we take the first comprehensive look at LR metamodeling, categorizing both the various types of LR metamodels and the contexts in which they might be employed.

1 INTRODUCTION

Simulation metamodeling—representing some aspect of the performance of a system that is described by a stochastic simulation via a functional model—has been of interest since at least the 1960's; see Kleijnen (1974), Kleijnen (1975) for one of the first comprehensive treatments. Early works focused on the *mean* response and *linear regression* metamodels, with an emphasis on experiment designs that exploited the advantages of simulation over a physical experiment; see for instance Schruben and Margolin (1978). There has been substantial progress since then for different responses and different metamodel forms.

The value of metamodeling is that it draws statistical strength from simulations run at a number of distinct *design points* to make better predictions at settings not yet simulated, or even at the design points themselves. Once created, a metamodel can typically be evaluated with little computational effort, while simulations at new settings take time. Further, the fitted metamodel can provide insight into system behavior—e.g., the coefficients of a linear regression may be interpreted as rates of change with respect to the design variables—or even be used for system optimization. Experiment design for fitting linear regression metamodels, and more recently inference based on Gaussian process metamodels, are well-studied topics in the simulation literature and beyond (Barton and Meckesheimer 2006).

Metamodeling inherently involves a bias-variance tradeoff: bias because the underlying functional model, even if "fitted" optimally, is not of the same form as the true, unknown response surface; and variance because the more flexible the base metamodel is the more sensitive it is to the random simulation

outputs employed to tune it. This is the case whether considering the basis functions in a linear regression or the correlation kernels in a Gaussian-process regression. Typically it is assumed (hoped) that the base metamodel is of the "correct form" or is flexible enough that the bias is not significant; therefore, the focus is on experiment design and parameter estimation to achieve low variance. Unfortunately, a low-bias metamodel may not always be achievable.

In this paper we consider a type of metamodeling that, when applicable, always leads to unbiased, or at the very least consistent, metamodels, but for which control of prediction variance may be difficult even with many design points. The approach is, loosely speaking, based on "importance sampling," but more precisely exploits *likelihood ratios (LRs)*; it is applicable when the design or decision variables are parameters of the simulation input distributions. "Inputs" are the fully specified probability distributions that drive the simulation, such as service times in queues and underlying asset values in finance. The idea is not new e.g., Beckman and McKay (1987), but there are a number of variations and different problem contexts, as well as myths, and these have not been considered comprehensively prior to this paper.

The organization of the paper is as follow: Section 2 links LR and classical metamodeling. Section 3 describes the possible benefits of employing LR metamodels, and the different contexts within which they might be useful. In Section 4 we carefully organize the various LR metamodels that have appeared in the literature, and in Section 5 we summarize an empirical evaluation.

2 FOUNDATION

We consider a simulation output $Y(\theta)$ whose distribution is a function of a $d \times 1$ vector of design or decision variables $\theta \in \Theta \subseteq \Re^d$. Our interest is in metamodeling some property of $Y(\theta)$ as a function of θ, such as $\mu(\theta) = \mathrm{E}[Y(\theta)]$ or $p(\theta) = \Pr\{Y(\theta) \le y\}$; we will focus on the mean $\mu(\cdot)$ to be concrete.

Suppose we have already performed simulation experiments at design points $\theta_1, \theta_2, \ldots, \theta_J$, observing stochastic simulation outputs $Y(\theta_1), Y(\theta_2), \ldots, Y(\theta_J)$, and we wish to predict the value of $\mu(\theta_0)$, where θ_0 may or may not be one of the design points. The predictors from a number of metamodeling approaches can be represented as

$$\widehat{\mu}(\theta_0) = \sum_{j=1}^{J} \widehat{w}_j(\theta_0, \theta_j) \cdot Y(\theta_j), \tag{1}$$

a weighted average of the observed responses at the design points; this includes both linear and Gaussian-process regression. The $\widehat{w}_j(\theta_0, \theta_j)$'s are estimated weights that typically depend on both the design point and the prediction point, and might depend on *all* of the design points. Our LR metamodels can also be expressed in this way, which we do to facilitate comparisons. For simplicity of presentation in this section we consider only a single simulation replication at each design point; later when we introduce replications, the weights may also depend on the replications.

Metamodeling methods differ in the philosophy that leads to the weights: For linear regression, a strong functional relationship between the response and the design variables is conjectured, such as $\mu(\theta) = \beta_0 + \beta^\top \theta$, where the β parameters are tuned using noisy data, leading to the estimated weights. Stated differently, the β's are learned by observing how the simulated system responds to different θ settings, so diverse experiment designs that tease out the individual effect of each element of θ are important. If $\mu(\theta) = \beta_0 + \beta^\top \theta$ is actually the correct form of the relationship then this sort of approach is hard to beat.

For Gaussian-process regression, the unknown response function $\mu(\cdot)$ is conjectured to be a realization of a Gaussian random field. This implies that $[\mu(\theta_0), \mu(\theta_1), \ldots, \mu(\theta_J)]$ have a multivariate normal distribution with spatial correlation structure that is a function of θ (Ankenman et al. 2010). The metamodel is the conditional distribution of $\mu(\theta_0)$ given noisy observations of the other responses, noisy observations that are also used to tune the spatial correlation relationship.

In most metamodeling methods it is important that the observed responses $Y(\theta_1), Y(\theta_2), \ldots, Y(\theta_J)$ estimate the response $\mu(\cdot)$ at distinct, diverse design points in Θ so that the structure of the relationship can be estimated. Bias arises because the underlying functional/spatial model is often incorrect.

The weights for LR metamodeling are derived from a different philosophy: weight the outputs $Y(\theta_1), Y(\theta_2), \ldots, Y(\theta_J)$ so that they each represent responses at the prediction point θ_0. This is only feasible when the θ's are parameters of the stochastic inputs to the simulation. We present the simplest case here for exposition.

Suppose we can represent the simulation output as $Y(\theta) = g(\mathbf{X})$ where $\mathbf{X} \sim f(\mathbf{x}|\theta)$ and f is the known joint distribution of the $m \times 1$ input $\mathbf{X}$. Define the likelihood ratio as $\ell_j(\mathbf{x}) = f(\mathbf{x}|\theta_0)/f(\mathbf{x}|\theta_j)$; throughout this paper we will assume that the support of $f(\cdot|\theta)$ is not a function of θ. Now consider observing $Y_j = Y(\theta_j) = g(\mathbf{X}_j)$, where $\mathbf{X}_j \sim f(\mathbf{x}|\theta_j), j = 1, 2, \ldots, J$ are independently generated at each design point. Then one possible LR metamodel is

$$\widehat{\mu}^{LR}(\theta_0) = \frac{1}{J}\sum_{j=1}^{J} \ell_j(\mathbf{X}_j) Y_j = \frac{1}{J}\sum_{j=1}^{J} \ell_j(\mathbf{X}_j) g(\mathbf{X}_j). \tag{2}$$

The weight in representation (1) is $\widehat{w}_j(\theta_0, \theta_j) = \ell_j(X_j)/J$. Standard results show that (2) is unbiased for $\mu(\theta_0)$, as in fact is each term $\ell_j(\mathbf{X}_j)Y_j$. Notice that diversity of the experiment design is no longer directly relevant, as how the output at θ_j is related to the output at θ_0 is in a sense already known via $\ell_j(\mathbf{X}_j)$. However, it is possible that including some $\ell_j(\mathbf{X}_j)Y_j$ terms in the average will actually increase the variance of the prediction, so a more refined weighting, including weight 0 for some design points, may be desirable. This type of metamodel, and its many variations, are what we investigate. See Chapter 9 of Owen (2013) for an introduction to the key ideas from the perspective of importance sampling for variance reduction; other key references include Hesterberg (1995), Owen and Zhou (2000) and Veach and Guibas (1995).

3 BENEFITS AND CONTEXTS

Clearly the range of applicabilty of LR metamodeling is more limited than other methods because of the restriction on θ. Why then, *might* it be useful?

- Predictions are *unbiased*, and no form of the metamodel has to be guessed or justified. Further, predictions are consistent as the number of replications at a fixed, finite set of design points increases, rather than as the number of design points increases. Consistency in classical metamodeling usually requires infill, meaning that Θ is packed more and more densely with design points asymptotically.
- A metamodel can be formed from a *single* design point, θ_1. No other method of which we are aware has this potential.
- The weights may be *the same* for any performance measure that is a function of Y, e.g., $\mathrm{E}[Y^2]$ or $I(Y \le y)$.

Unfortunately, none of these potential benefits guarantee a low-variance prediction.

We have also identified three distinct contexts within which LR metamodels might be used: Global θ_0, Moving θ_0, and Target θ_0. The effectiveness of LR methods and which variation we might use can depend on the context.

3.1 Global θ_0

This is the context of classical metamodeling: The design points $\theta_1, \theta_2, \ldots, \theta_J$ are selected so that their simulation results facilitate good predictions for *any* $\theta_0 \in \Theta$. For LR metamodeling, the question is both what design points to choose, and which of them to actually employ for a given θ_0. Due to the potential for variance inflation, these are not easy questions, as we illustrate in our empirical results.

3.2 Moving θ_0

Here the design points $\theta_1, \theta_2, \theta_3, \ldots$ are revealed sequentially, but they move stochastically throughout Θ. At stage k of experimentation we have the simulation outputs from design points $\theta_1, \theta_2, \ldots, \theta_{k-1}$ (and

possibly θ_k) and we want to predict some property of $Y(\theta_k)$ using all or some of the accumulated results. This is the context of Feng and Staum (2017) in which θ_k represents some financial market conditions in force on day k. They show that under certain conditions the LR metamodel predictions converge as the number of repeated experiments increases even though the number of simulation replications per experiment remains constant. We do not explore this context any further here.

3.3 Target θ_0

Here the design points $\theta_1, \theta_2, \theta_3, \ldots$ are also revealed sequentially, but they approach an unknown, fixed value θ_0. At stage k of experimentation, we have the simulation outputs from design points $\theta_1, \theta_2, \ldots, \theta_k$ and we want to predict some property of $Y(\theta_k)$ using some or all of the accumulated results. This might occur if θ_0 is being estimated from real-world data and each day brings additional data; or if θ_0 is a solution to a simulation optimization problem (e.g., mean maximizing) and we are employing some search method to find it. To the best of our knowledge, this context has not been considered previously. We address this context in our empirical study and show that LR metamodeling has substantial potential.

4 LR METAMODELS AND PROPERTIES

In this section we survey many of the variations of LR metamodels. The goal is to introduce the key ideas and properties of these estimators; other combinations of these ideas are possible. For ease of notation, let $\mathbf{X}_{ij}$ denote the ith independent sample from the jth distribution, indexed by θ_j; that is $\mathbf{X}_{ij} \overset{i.i.d.}{\sim} f(\cdot|\theta_j)$. We also let $Y_{ij} = Y_i(\theta_j) = g(\mathbf{X}_{ij})$.

4.1 LR Metamodels Formed from One Design Point

We first introduce three types of LR metamodels that are formed from a single design point. We summarize known results on the variance of these estimators as a measure of their efficiency. For clarity, we let θ_1 be the single design point from which we have obtained n_1 replications, and θ_0 be the prediction point that has not been simulated.

Baseline Likelihood Ratio (BLR) Metamodel:

$$\widehat{\mu}_1^{BLR}(\theta_0; n_1) = \frac{1}{n_1} \sum_{i=1}^{n_1} \ell_1(\mathbf{X}_{i1}) g(\mathbf{X}_{i1}). \tag{3}$$

For this estimator we have $\mathbb{E}\left[\widehat{\mu}_1^{BLR}(\theta_0; n_1)\right] = \mu(\theta_0)$ and $\mathbb{V}\mathrm{ar}\left[\widehat{\mu}_1^{BLR}(\theta_0; n_1)\right] = \sigma^2_{1,BLR}/n_1$ where

$$\begin{aligned} \sigma^2_{1,BLR} &= \mathbb{V}\mathrm{ar}\left[\ell_1(\mathbf{X}_{i1}) g(\mathbf{X}_{i1})\right] = \mathbb{E}\left[\ell_1(\mathbf{X}_{i1})^2 g(\mathbf{X}_{i1})^2\right] - \mu(\theta_0)^2 \\ &= \int_\Omega \frac{\left(g(\mathbf{x}) f(\mathbf{x}|\theta_0) - \mu(\theta_0) f(\mathbf{x}|\theta_1)\right)^2}{f(\mathbf{x}|\theta_1)} \, d\mathbf{x}. \end{aligned} \tag{4}$$

We observe from (4) that we have an estimator with small variance if $(g(\mathbf{x}) f(\mathbf{x}|\theta_0) - \mu(\theta_0) f(\mathbf{x}|\theta_1))^2$ is small. On the other hand, if $f(\mathbf{x}|\theta_1)$ is very small in regions where $(f(\mathbf{x}|\theta_0) g(\mathbf{x}) - \mu f(\mathbf{x}|\theta_1))^2$ lacks this proportionality then we can have a very large variance. Indeed, in some cases the variance can be infinite (Owen 2013). The next two estimators mitigate the potential of large variance by accepting some bias. Both exploit the fact that under mild conditions, $\mathbb{E}_{\theta_1}[\ell_1(\mathbf{X}_{i1})] = 1$.

Self-normalizing Likelihood Ratio (SLR) Metamodel: The SLR estimator renormalizes the BLR estimator so that the weights constructed using the likelihood ratios sum up to 1. Specifically,

$$\widehat{\mu}_1^{SLR}(\theta_0; n_1) = \sum_{i=1}^{n_1} \left[\frac{\ell_1(\mathbf{X}_{i1})}{\sum_{p=1}^{n_1} \ell_1(\mathbf{X}_{p1})} \right] g(\mathbf{X}_{i1}). \tag{5}$$

Using the Delta method, we can show that,

$$\sqrt{n_1}\left(\widehat{\mu}_1^{SLR}(\theta_0;n_1)-\mu(\theta_0)\right)\Rightarrow \mathrm{N}\left(0,\sigma^2_{1,SLR}\right)$$

as $n_1\to\infty$, where

$$\begin{aligned}\sigma^2_{1,SLR} &= \frac{\mathbb{E}\left[\left(\ell_1(\mathbf{X}_{i1})g(\mathbf{X}_{i1})-\mu(\theta_0)\ell_1(\mathbf{X}_{i1})\right)^2\right]}{\left(\mathbb{E}\left[\ell_1(\mathbf{X}_{i1})\right]\right)^2}=\mathbb{E}\left[(g(\mathbf{X}_{i1})-\mu(\theta_0))^2\ell_1(\mathbf{X}_{i1})^2\right]\\ &= \int_\Omega \frac{\left(g(\mathbf{x})f(\mathbf{x}|\theta_0)-\mu(\theta_0)f(\mathbf{x}|\theta_0)\right)^2}{f(\mathbf{x}|\theta_1)}\,d\mathbf{x}. \end{aligned} \tag{6}$$

This establishes consistency, but not unbiasedness. Comparing (6) to (4), the numerator in (6) is independent of θ_1, while in (4) it is not. We notice that

$$\mathbb{E}\left[(g(\mathbf{X}_{i1})-\mu(\theta_0))^2\ell_1(\mathbf{X}_{i1})^2\right]\le \mathbb{E}\left[(g(\mathbf{X}_{i1})-\mu(\theta_0))^4\right]^{1/2}\mathbb{E}\left[\ell_1(\mathbf{X}_{i1})^4\right]^{1/2}.$$

Thus, when the design space is bounded and under suitable moment conditions the possibility of infinite variance is eliminated.

Remark 1 A variation of LR metamodeling that we will not cover in detail is the so-called rejection method (Owen 2013): The idea is to resample from $\mathbf{X}_{11},\mathbf{X}_{21},\ldots,\mathbf{X}_{n_1,1}$ so that it is representative of samples from $f(\cdot|\theta_0)$ by using rejection based on the revised likelihoods. However, we can show that rejection is equivalent to SLR asymptotically, making SLR preferable.

Regression/Control Variate Likelihood Ratio (RLR) Metamodel: While SLR renormalizes the observed likelihood ratios to have sum 1, the RLR metamodel uses the difference between the average likelihood ratio and 1 to correct the BLR metamodel in a way that reduces variance. A general treatment of regression-based control variates can be found in Nelson (1990). Consider an estimator of the form

$$\frac{1}{n_1}\sum_{i=1}^{n_1}\left[\ell_1(\mathbf{X}_{i1})g(\mathbf{X}_{i1})-\beta_1(\ell_1(\mathbf{X}_{i1})-1)\right].$$

The β_1 that minimizes the variance of this estimator is $\beta_1=\mathbb{C}\mathrm{ov}\left[\ell_1(\mathbf{X}_{i1})g(\mathbf{X}_{i1}),\ell_1(\mathbf{X}_{i1})\right]/\mathbb{V}\mathrm{ar}\left[\ell_1(\mathbf{X}_{i1})\right]$, which make the variance $\sigma^2_{1,RLR}/n_1$, where

$$\begin{aligned}\sigma^2_{1,RLR} &= \mathbb{V}\mathrm{ar}\left[\ell_1(\mathbf{X}_{i1})g(\mathbf{X}_{i1})\right]-\frac{\mathbb{C}\mathrm{ov}\left[\ell_1(\mathbf{X}_{i1})g(\mathbf{X}_{i1}),\ell_1(\mathbf{X}_{i1})\right]^2}{\mathbb{V}\mathrm{ar}\left[\ell_1(\mathbf{X}_{i1})\right]}\\ &= \sigma^2_{1,BLR}\left(1-\mathbb{C}\mathrm{orr}\left[\ell_1(\mathbf{X}_{i1})g(\mathbf{X}_{i1}),\ell_1(\mathbf{X}_{i1})\right]^2\right).\end{aligned}$$

Thus, the variance is no greater, and typically less than, BLR. In fact, the correlation will be large—and thus the variance small—when $\ell_1(\mathbf{x})\propto\ell_1(\mathbf{x})g(\mathbf{x})$, which is different from BLR for which we will have small variance if $\ell_1(\mathbf{x})\propto 1/g(\mathbf{x})$. Thus, RLR might be effective when BLR is not and vice versa.

To implement RLR we have to estimate β_1. Let $\bar{\ell}_1=\sum_{i=1}^{n_1}\ell_1(\mathbf{X}_{i1})/n_1$. The least-squares regression estimator is

$$\widehat{\beta}_1=\frac{\sum_{i=1}^{n_1}(\ell_1(\mathbf{X}_{i1})-\bar{\ell}_1)g(\mathbf{X}_{i1})\ell_1(\mathbf{X}_{i1})}{\sum_{i=1}^{n_1}(\ell_1(\mathbf{X}_{i1})-\bar{\ell}_1)^2}.$$

The RLR estimator is thus

$$\widehat{\mu}_1^{RLR}(\theta_0;n_1)=\frac{1}{n_1}\sum_{i=1}^{n_1}\left[\ell_1(\mathbf{X}_{i1})g(\mathbf{X}_{i1})-\widehat{\beta}_1(\ell_1(\mathbf{X}_{i1})-1)\right]. \tag{7}$$

Employing $\widehat{\beta}_1$ inflates the variance by a factor of approximately $(n_1-2)/(n_1-3)$, which is negligible when n_1 is not too small (Nelson 1990).

4.2 LR Metamodels Formed from Multiple Design Points

The LR metamodels in Section 4.1 are built using simulation outputs from a single design point θ_1. These will rarely be sufficient in practice, and rather are building blocks for LR metamodels based on multiple design points $\theta_1, \theta_2, \ldots, \theta_J$. Since single-design-point LR metamodels are unbiased (or nearly so), simple averaging of J of them is possible. However, more sophisticated weighted averages can perform substantially better (lead to smaller variances). We consider these variations in this section.

4.2.1 Linear Combinations of Single-Design-Point LR Metamodels

A direct way to form LR metamodels using multiple design points is to combine the LR metamodels presented in Section 4.1. Specifically, consider LR metamodels of the following form:

$$\widehat{\mu}^{\text{est}}(\theta_0) = \sum_{j=1}^{J} \widehat{w}_j(\theta_0, \theta_j)\widehat{\mu}_j^{\text{est}}(\theta_0), \tag{8}$$

where $\widehat{\mu}_j^{\text{est}}(\theta_0)$ are the single-design-point LR metamodels discussed in Section 4.1; i.e., "est" may be "BLR," "SLR," or "RLR." The weights $\widehat{w}(\theta_0, \theta_j)$ satisfy $\widehat{w}(\theta_0, \theta_j) \geq 0$ and $\sum_{j=1}^{n} \widehat{w}(\theta_0, \theta_j) = 1$. We allow $\widehat{w}(\theta_0, \theta_j)$ to depend on both the prediction point and all of the design points.

Optimally Weighted: Let $\sigma_{j,\text{est}}^2 = \mathbb{V}\text{ar}\left[\widehat{\mu}_j^{\text{est}}(\theta_0; 1)\right]$. The variance-minimizing weights solve

$$\begin{array}{ll} \min & \mathbb{V}\text{ar}\left[\sum_{j=1}^{J} \widehat{w}_j(\theta_0, \theta_j)\widehat{\mu}_j^{\text{est}}(\theta_0; n_j)\right] = \sum_{j=1}^{J} \widehat{w}_j^2(\theta_0, \theta_j)\dfrac{\sigma_{j,\text{est}}^2}{n_j} \\ \text{s.t.} & \sum_{i=1}^{J} \widehat{w}_j(\theta_0, \theta_j) = 1 \\ & \widehat{w}_j(\theta_0, \theta_j) \geq 0, \ j = 1, 2, \ldots, J. \end{array}$$

The optimal solution is

$$\widehat{w}_j^{\text{est}-OW}(\theta_0, \theta_j) = \frac{n_j \sigma_{j,\text{est}}^{-2}}{\sum_{q=1}^{J} n_q \sigma_{q,\text{est}}^{-2}}.$$

This gives us

$$\widehat{\mu}^{\text{est}-OW}(\theta_0) = \sum_{j=1}^{J} \left(\frac{n_j \sigma_{j,\text{est}}^{-2}}{\sum_{q=1}^{J} n_q \sigma_{q,\text{est}}^{-2}} \right) \widehat{\mu}_j^{\text{est}}(\theta_0; n_j). \tag{9}$$

Notice that the larger the number of replications or the smaller the variance at design point j, the larger the weight assigned to it.

Despite the theoretical optimality of the weights $\widehat{w}_j^{\text{est}-OW}(\theta_0, \theta_j)$, they depend on the LR variances $\sigma_{j,\text{est}}^2 = \mathbb{V}\text{ar}\left[\widehat{\mu}_j^{\text{est}}(\theta_0)\right]$. In practical implementations, these quantities are unknown and therefore need to be estimated from simulation outputs. Unfortunately, our experience is that the estimation errors can signficantly inflate the variance relative to the true optimal weights. Moreover, the impact of these estimation errors increases as more design points are included. *A feature of LR metamodels is that adding design points is not always productive.*

To overcome the variance estimation error one may use simpler weights that do not require any distributional information about the design point θ_j or the prediction point θ_0. For instance, we can put equal weight on each single-design-point LR metamodel. This gives us the Equally Weighted LR Metamodel

$$\widehat{\mu}^{\text{est}-EW}(\theta_0) = \frac{1}{J} \sum_{j=1}^{J} \widehat{\mu}_j^{\text{est}}(\theta_0; n_j). \tag{10}$$

A slight enhancement is to make the weights proportional to the sample size, that is $\widehat{w}_j^{PW}(\theta_0,\theta_j) = n_j/n$, where $n = \sum_{j=1}^{J} n_j$. This yields the Proportionally Weighted LR Metamodel

$$\widehat{\mu}^{\text{est}-PW}(\theta_0) = \sum_{j=1}^{J} \left(\frac{n_j}{n}\right) \widehat{\mu}_j^{\text{est}}(\theta_0; n_j). \tag{11}$$

When the sample sizes are chosen so that all of the $\sigma^2_{j,\text{est}}/n_j$'s are close in value, then the equally weighted metamodel will work well.

4.2.2 Linear Combinations of Replications at Different Design Points

Another way to build LR metamodels from multiple design points is to use more granular weights $\widehat{w}(\theta_0,\theta_j,\mathbf{X}_{ij})$ to combine the individual replications at different design points. In this section we consider LR metamodels of the form

$$\widehat{\mu}^{\text{est}}(\theta_0) = \sum_{j=1}^{J} \frac{1}{n_j} \sum_{i=1}^{n_j} \widehat{w}(\theta_0,\theta_j,\mathbf{X}_{ij}) \ell_j(\mathbf{X}_{ij}) Y_{ij}. \tag{12}$$

In addition to the prediction point θ_0 and *all* design points $\theta_1,\theta_2,\ldots,\theta_J$, the weights $\widehat{w}(\theta_0,\theta_j,\mathbf{X}_{ij})$ may also depend on *all* of the simulation inputs and outputs $(\mathbf{X}_{ij},Y_{ij})$ for $i = 1,2,\ldots,n_j$ and $j = 1,2,\ldots,J$. As studied in Veach and Guibas (1995) in the context of multiple importance sampling, the combined LR metamodel $\widehat{\mu}^{\text{est}}(\theta_0)$ is unbiased if $\sum_{j=1}^{J} \widehat{w}(\theta_0,\theta_j,\mathbf{X}_{ij}) = 1$ for all $\mathbf{x}$ in the common support.

Let $\alpha_j = n_j/n$ where $n = \sum_{j=1}^{J} n_j$; our results below are specific to this choice of α_j. Given the J design points $\theta_1,\theta_2,\ldots,\theta_J$, define the corresponding mixture distribution and mixture likelihood ratio as

$$f(\mathbf{x}|\theta_\alpha) = \sum_{j=1}^{J} \alpha_j f(\mathbf{x}|\theta_j) \quad \text{and} \quad \ell_\alpha(\mathbf{x}) = \frac{f(\mathbf{x}|\theta_0)}{f(\mathbf{x}|\theta_\alpha)}. \tag{13}$$

Given simulation outputs $Y_{ij}, i = 1,2,\ldots,n_j$, where the inputs are sampled from distribution $f(\mathbf{x}|\theta_j)$, for $j = 1,2,\ldots,J$, the main idea is to view the entire collection of simulation outputs as a sample from this mixture distribution, where the number of observations from each distribution is forced to be proportional to the mixture probability. Similar to Section 4.1, we next introduce three variants of mixture LR estimators.

BLR-M Metamodel Consider the weights

$$\widehat{w}^{BLR-M}(\theta_0,\theta_j,\mathbf{X}_{ij}) = \frac{\alpha_j f(\mathbf{X}_{ij}|\theta_j)}{f(\mathbf{X}_{ij}|\theta_\alpha)}, \quad i = 1,2,\ldots,n_j;\ j = 1,2,\ldots,J.$$

Then we have

$$\widehat{\mu}^{BLR-M}(\theta_0) = \sum_{j=1}^{J} \frac{1}{n_j} \sum_{i=1}^{n_j} \frac{\alpha_j f(\mathbf{X}_{ij}|\theta_j)}{f(\mathbf{X}_{ij}|\theta_\alpha)} \ell_j(\mathbf{X}_{ij}) Y_{ij} = \frac{1}{n} \sum_{j=1}^{J} \sum_{i=1}^{n_j} \ell_\alpha(\mathbf{X}_{ij}) Y_{ij}. \tag{14}$$

For each simulation input $\mathbf{X}_{ij}$ at design point θ_j, the weights $\widehat{w}^{BLR-M}(\theta_0,\theta_j,\mathbf{X}_{ij})$ depend on the prediction point, *all* of the design points, and the particular simulation output Y_{ij}.

The BLR-M metamodel coincides with the "balance heuristic" studied in Veach and Guibas (1995) in the context of multiple importance sampling. Veach and Guibas (1995) showed that the BLR-M metamodel cannot be much worse than any other LR metamodel of the form (12). This robustness protects us from infinite variance if we also include simulations from the prediction point itself (Owen and Zhou 2000).

Theorem 1 (Paraphrase of Theorem 9.2 in Veach and Guibas (1995)) Let $\widehat{\mu}^{\text{est}}(\theta_0)$ be any unbiased estimator of the form (12), then

$$\mathbb{V}\text{ar}\left[\widehat{\mu}^{BLR-M}(\theta_0)\right] \leq \mathbb{V}\text{ar}\left[\widehat{\mu}^{\text{est}}(\theta_0)\right] + \left(\frac{1}{\min_j\{n_j\}} - \frac{1}{\sum_{j=1}^J n_j}\right)\mu(\theta_0)^2.$$

SLR-M Metamodel Motivated by the SLR metamodel in Section 4.1, consider the self-normalizing weights

$$\widehat{w}^{SLR-M}(\theta_0,\theta_j,\mathbf{X}_{ij}) = \frac{n_j f(\mathbf{X}_{ij}|\theta_j)/f(\mathbf{X}_{ij}|\theta_\alpha)}{\sum_{q=1}^J\sum_{p=1}^{n_q}\ell_\alpha(\mathbf{X}_{pq})}, \quad i=1,2,\ldots,n_j; j=1,2,\ldots,J.$$

Then we have

$$\widehat{\mu}^{SLR-M}(\theta_0) = \sum_{j=1}^J\sum_{i=1}^{n_j}\left[\frac{\ell_\alpha(\mathbf{X}_{ij})}{\sum_{q=1}^J\sum_{p=1}^{n_q}\ell_\alpha(\mathbf{X}_{pq})}\right]Y_{ij}. \tag{15}$$

For each simulation input $\mathbf{X}_{ij}$ at design point θ_j, the weights $\widehat{w}^{SLR-M}(\theta_0,\theta_j,\mathbf{X}_{ij})$ depend on the prediction point, *all* of the design points, and *all* of the simulation outputs from *all* of the design points. Analogous to the SLR metamodel, the estimator is biased but consistent. To the best of our knowledge $\widehat{\mu}^{SLR-M}(\theta_0)$ has not been studied in the literature, but it is an obvious competitor.

RLR-M Metamodel Motivated by the RLR metamodel in Section 4.1, we consider the estimator

$$\begin{aligned}\widehat{\mu}^{RLR-M}(\theta_0) &= \frac{1}{n}\sum_{j=1}^J\sum_{i=1}^{n_j}\left[\ell_\alpha(\mathbf{X}_{ij})Y_{ij} - \beta_\alpha(\ell_\alpha(\mathbf{X}_{ij})-1)\right] \\ &= \widehat{\mu}^{BLR-M}(\theta_0) - \beta_\alpha\left(\bar{\ell}_\alpha - 1\right),\end{aligned} \tag{16}$$

where $\bar{\ell}_\alpha = \sum_{j=1}^J\sum_{i=1}^{n_j}\ell_\alpha(\mathbf{X}_{ij})/n$. We notice that

$$\mathbb{E}\left[\bar{\ell}_\alpha\right] = \frac{1}{n}\sum_{j=1}^J\sum_{i=1}^{n_j}\int_\Omega\frac{f(\mathbf{x}|\theta_0)}{f(\mathbf{x}|\theta_\alpha)}f(\mathbf{x}|\theta_j)d\mathbf{x} = \int_\Omega f(\mathbf{x}|\theta_0)d\mathbf{x} = 1.$$

The variance-minimizing β_α is

$$\beta_\alpha = \frac{\mathbb{C}\text{ov}\left[\bar{\ell}_\alpha,\widehat{\mu}^{BLR-M}(\theta_0)\right]}{\mathbb{V}\text{ar}\left[\bar{\ell}_\alpha\right]} = \frac{\sum_{j=1}^J\alpha_j\mathbb{C}\text{ov}\left[\ell_\alpha(\mathbf{X}_{1j}),\ell_\alpha(\mathbf{X}_{1j})Y_{1j}\right]}{\sum_{j=1}^J\alpha_j\mathbb{V}\text{ar}\left[\ell_\alpha(\mathbf{X}_{1j})\right]},$$

which can be estimated via regression, yielding

$$\widehat{\beta}_\alpha = \frac{\sum_{j=1}^J\sum_{i=1}^{n_j}(\ell_\alpha(\mathbf{X}_{ij})-\bar{\ell}_\alpha)\ell_\alpha(\mathbf{X}_{ij})Y_{ij}}{\sum_{j=1}^J\sum_{i=1}^{n_j}(\ell_\alpha(\mathbf{X}_{ij})-\bar{\ell}_\alpha)^2}.$$

Due to the control variate feature this estimator achieves a smaller variance than $\widehat{\mu}^{BLR-M}(\theta_0)$ provided n is not too small. To the best of our knowledge, $\widehat{\mu}^{RLR-M}(\theta_0)$ has not been studied in the literature.

Remark 2 The RLR-M metamodel combines all J design points into a single control variate, but it is also possible to treat them as J individual control variates. This has the advantage that one could try to select only effective control variates using methods such as those in Bauer and Wilson (1992). The disadvantage is that the variance inflation factor becomes approximately $(n-2)/(n-J-2)$ (Nelson 1990).

5 EMPIRICAL STUDY

Consider a stochastic activity network (SAN) with five activities for which the simulation response is the time to complete the network given by $Y(\theta) = \max\left\{X^{(1)}+X^{(4)}, X^{(1)}+X^{(3)}+X^{(5)}, X^{(2)}+X^{(5)}\right\}$, where the activity times are independent Exponentials, i.e. $X^{(k)} \sim \theta^{(k)} e^{-x\theta^{(k)}}$, $k = 1,2,\ldots,5$. Let $\mathbf{X} = \left(X^{(1)}, X^{(2)}, \ldots, X^{(5)}\right)$. We focus on LR metamodels for $\mu(\theta) = \mathbb{E}[Y(\theta)]$.

In the first setting, we want to make predictions at any $\theta = (\theta^{(1)}, \theta^{(2)}, \ldots, \theta^{(5)})$ in a space of possible values; i.e. we are building a global metamodel. We consider the extreme case in which we simulate at only a *single* θ to cover the entire parameter space. In the second setting, the true distribution of $\mathbf{X}$ is fixed and has the rate vector $\theta_0 = (\theta_0^{(1)}, \theta_0^{(2)}, \ldots \theta_0^{(5)})$. On each time click t we estimate θ_0 from tm accumulated i.i.d. observations of $\mathbf{X}$. In particular, the t-th design point θ_t is the maximum likelihood estimator of θ_0. Then $\theta_t \to \theta_0$ as $t \to \infty$ almost surely. We also simulate n i.i.d. activity times following the distribution of θ_t. We look at the improvement in estimating $\mu(\theta_0)$ at time t from using all simulations at $\theta_1, \theta_2, \ldots, \theta_t$ via LR metamodeling. This is a target θ_0 example.

5.1 SAN Estimation for Global θ_0

In this experiment we compare and contrast the three LR metamodels formed from one design point, i.e., the ones discussed in Section 4.1. In particular, we consider the SAN with service rates $\theta = (\theta_1, 1, 1, 1, \theta_5)$ where $(\theta_1, \theta_5) \in [0.5, 1.5] \times [0.5, 1.5]$. Let $\mathbf{\Theta} = [0.5, 1.5] \times 1 \times 1 \times 1 \times [0.5, 1.5]$ be the design space. We are interested in estimating $\mu(\theta)$ for all $\theta \in \mathbf{\Theta}$ using LR metamodels that are formed from simulation outputs at the center $\theta^* = (1, 1, 1, 1, 1)$. A 20-by-20 grid is formed in the space $(\theta_1, \theta_5) \in [0.5, 1.5] \times [0.5, 1.5]$. The true response surface at each grid point is estimated by the sample average of 10^7 independent replications; the resulting response surface is depicted in Figure 1a. We see that the time-to-completion is long when the activity rates are low (top left corner of the surface) and is short when the activity rates are high (bottom right corner of the surface).

The accuracy of each LR metamodel is assessed by its MSE at each grid point over 10^4 macro replications. In each macro replication i, we run 100 independent replications at θ^* then use those outputs to estimate $\widehat{\mu}^{\text{est},i}(\theta; 100)$ at all 20^2 grid points, where the superscript *est* could be BLR, SLR, or RLR. In addition, for comparison, in each macro replication we also run 100 independent replications at each of the 400 grid points to estimate the response surface using the standard Monte Carlo (SMC) estimator, i.e., the sample average. The estimated MSE (variance) of the SMC estimator using the 10^4 macro replications is depicted in Figure 1b. We see that the variance of the SMC estimator is large in regions where the value of the response surface is large. The MSE at each grid point is calculated as $\text{MSE}^{\text{est}}(\theta) = \frac{1}{10{,}000} \sum_{i=1}^{10{,}000} (\widehat{\mu}^{\text{est},i}(\theta; 100) - \mu(\theta))^2$. To facilitate easy comparisons, we plotted the MSEs for the three LR metamodels *relative to the variance of the SMC estimator* (the ratio) in Figures 1c–1e. Note that although the same colors are used in these figures, the ranges of MSEs are different.

Comparing Figures 1c—1e, we see that the accuracies for the three metamodels are quite different. Despite the computational savings, LR metamodels can sometimes provide predictions with inflated MSEs, especially when one of the two activity rates is low, i.e., $\theta_1 \in [0.5, 0.8]$ or $\theta_5 \in [0.5, 0.8]$; sometimes so much that the prediction becomes useless. In particular, the predictions made by BLR estimators could have MSEs 100 times larger than the SMC variance; we truncated the MSEs in Figure 1c at 100 for ease of visualization. However, the RLR metamodel predictions have smaller MSE than the variance of the SMC estimator in a large area of the design space. Based on the different ranges of relative MSEs, the RLR metamodel is more accurate than the SLR metamodel, which is more accurate than the BLR metamodel. Despite different ranges, all three LR metamodels suffer large MSEs when one of the two activity rates is low. Moreover, the degradation of MSE is much faster than the increase of variance for the SMC estimator. This means that, when using LR metamodels for prediction, one needs to be careful about which design point to employ. Lastly, we see from Figure 1e that the RLR metamodel can have lower MSEs for some θ's than the MSE at the design point θ^* itself. This suggests that careful consideration about where to

employ the LR metamodel, or combining the LR estimators from only a selected subset of the design points, could yield significant benefits at the cost of density evaluations to form the LR. However, in most practical problems simulation model execution is significantly more costly than density evaluation. Thus, LR metamodels can offer significant computational savings, and may be the preferred to SMC even for design points with inflated MSEs once the user considers the precision-computation trade off.

In a further experiment, we explore whether combining LR metamodel predictions and the SMC estimator has improved MSEs compared to each estimator in isolation. In particular, at each of the 20^2 grid points we consider the average of the SMC estimate and the LR metamodel prediction using simulation outputs at $\theta^* = (1,1,1,1,1)$. Similar to the first experiment, 10^4 macro replications are run and the resulting MSEs relative to the variance of merely the SMC estimator are depicted in Figures 1f–1h. Again these figures have very different ranges despite the similarity in colors. Comparing Figures 1f to 1c, 1g to 1d, and 1h to 1e, we see that the combined predictions have smaller MSEs, hence are more accurate than the LR metamodel prediction alone. Based on the values of the relative MSEs in these figures, we see that combining RLR metamodel predictions and the SMC estimator always lowers MSEs. When both activity rates are high (e.g., higher than 0.7 for SLR and higher than 0.9 for BLR), combining SLR or BLR metamodel predictions with the SMC estimator can also lower the MSE compared to the SMC estimator alone. Notice that incorporating samples from multiple design points facilitates using the more sophisticated combination methods discussed in Section 4.2, which will lead to even better performance.

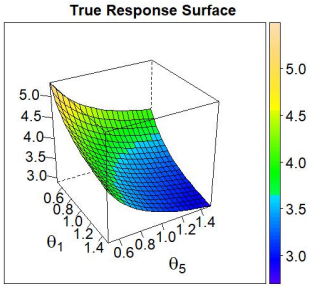
5.2 SAN Estimation for Target θ_0

In this experiment, we apply LR metamodels formed from multiple design points, i.e., the LR metamodels discussed in Section 4.2. At each time t, $t = 1,2,\ldots,100$, we obtain $m = 10$ new random observations of the five activity times, and use the accumulated tm observations to construct an estimator of θ_0, denoted θ_t. All of the observations are exponentially distributed with a common target activity rate vector $\theta_0 = (1,1,1,1,1)$. Therefore, the strong law of large numbers implies that $\lim_{t\to\infty} \theta_t \xrightarrow{a.s.} \theta_0$, hence the name *target* θ_0. At time t, we simulate $n = 50$ independent replications of time-to-completion $Y(\theta_t)$. Our goal is to construct an estimator of $\mu(\theta_0)$ using the samples generated at θ_s, $1 \le s \le t$. We denote the estimator constructed at t as $\hat{\mu}^{\text{est}}(\theta_t)$.

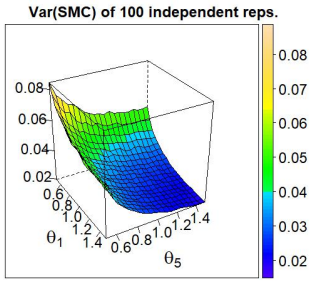
We consider 9 different LR metamodels: each of the three basic LR metamodels, BLR, SLR, RLR, with each of the three combination methods, PW, OW, and M, as discussed in Section 4.2. As a benchmark for comparison, the sample average of the n independent replications is included as the *crude* estimator. As a second benchmark, we consider a *brute force* estimator, which is a sample average of nt independent replications of $Y(\theta_t)$ at time t.

To access the accuracies for the LR estimators, we conducted 10^4 independent macro replications, each of which consists of $t = 1,2,\ldots,100$ experiments as described above. The MSEs are calculated over the macro replications at each of the 100 experiments, i.e., $\text{MSE}_t = \frac{1}{10{,}000}\sum_k^{10{,}000}(\widehat{\mu}_k^{\text{est}}(\theta_t) - \mu(\theta_0))^2$, where *est* is any of the 9 LR metamodels or the 2 benchmark estimators. The true $\mu(\theta_0)$ is estimated by the SMC estimator using 10^7 independent replications. The resulting MSEs are plotted in log-scale in Figure 2.

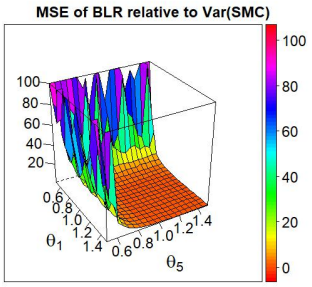
We next summarize our observations. We find that self-normalization is beneficial in all LR metamodels, regardless the method of combination (see Figure 2a, for example). Note that in the regression-based LR estimators we treat each design point as an individual regression before combining. The regression estimators have fairly good performance in general, but it deteriorates a bit as more experiments are run; we conjecture that the control-variate correction is less effective, because there is less difference from 1 as θ_t approaches θ_0. We observe poor performance when the optimal weights are estimated (OW, see Figure 2b); this observation is consistent with those in Feng and Staum (2017). Comparing the MSEs for the three M-based LR metamodels in Figure 2c, we see that the M-based estimators perform reasonably well regardless of the basic LR metamodel employed. When comparing the MSEs of the LR metamodels to the two benchmark estimators, we see that all three M-based LR metamodels produce estimates that are almost as accurate as the brute force estimator, but with significant computational savings relative to brute force.



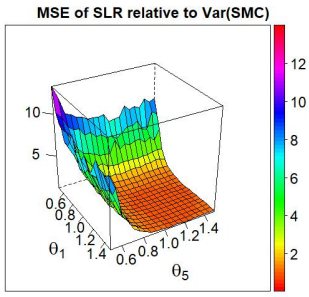
(a) The true response surface.



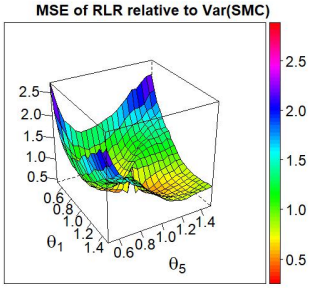
(b) Var of SMC estimator.



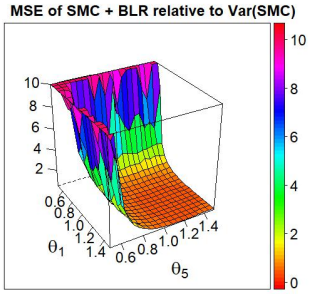
(c) MSE for BLR estimator.



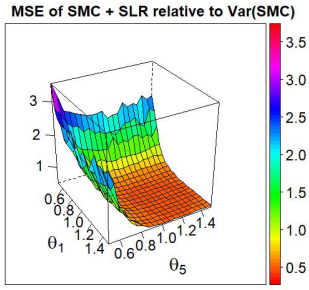
(d) MSE for SLR estimator.



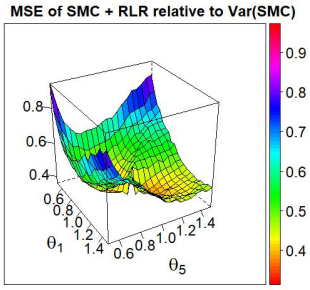
(e) MSE for RLR estimator.



(f) MSE for SMC + BLR.

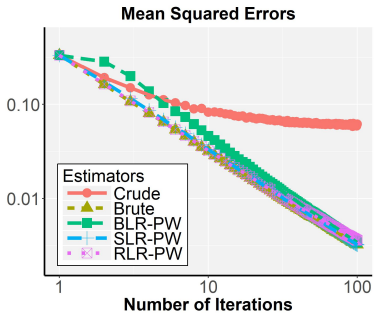


(g) MSE for SMC + SLR.

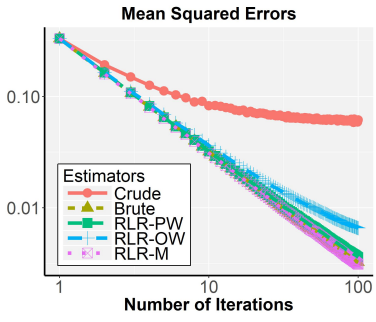


(h) MSE for SMC + RLR.

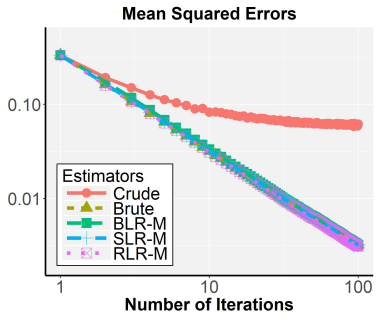
Figure 1: Global metamodels for SAN example.



(a) PW-based LR metamodels.



(b) Different combination methods.



(c) Different basic LR metamodels.

Figure 2: MSEs vs. number of iterations for LR metamodels formed from multiple design points.

ACKNOWLEDGEMENTS

This research was partially supported by the National Science Foundation of the United States under Grant Number CMMI-1634982.

REFERENCES

Ankenman, B., B. L. Nelson, and J. Staum. 2010. "Stochastic Kriging for Simulation Metamodeling". *Operations Research* 58(2):371–382.

Barton, R. R., and M. Meckesheimer. 2006. "Metamodel-Based Simulation Optimization". In *Handbooks in Operations Research and Management Science: Simulation*, edited by S. Henderson and B. L. Nelson, Chapter 18, 535–574. New York: Elsevier.

Bauer, K. W., and J. R. Wilson. 1992. "Control-Variate Selection Criteria". *Naval Research Logistics* 39(3):307–321.

Beckman, R. J., and M. D. McKay. 1987. "Monte Carlo Estimation under Different Distributions using the same Simulation". *Technometrics* 29(2):153–160.

Feng, M., and J. Staum. 2017. "Green Simulation: Reusing the Output of Repeated Experiments". *ACM Transactions on Modeling and Computer Simulation* 27(4):23:1–23:28.

Hesterberg, T. 1995. "Weighted Average Importance Sampling and Defensive Mixture Distributions". *Technometrics* 37(2):185–194.

Kleijnen, J. P. C. 1974. *Statistical Techniques in Simulation, Part I*. New York: Marcel Dekker.

Kleijnen, J. P. C. 1975. *Statistical Techniques in Simulation, Part II*. New York: Marcel Dekker.

Nelson, B. L. 1990. "Control Variate Remedies". *Operations Research* 38(6):974–992.

Owen, A., and Y. Zhou. 2000. "Safe and Effective Importance Sampling". *Journal of the American Statistical Association* 95(449):135–143.

Owen, A. B. 2013. "*Monte Carlo Theory, Methods and Examples*". http://statweb.stanford.edu/~owen/mc/. [Online; accessed 2-April-2018].

Schruben, L. W., and B. H. Margolin. 1978. "Pseudorandom Number Assignment in Statistically Designed Simulation and Distribution Sampling Experiments". *Journal of the American Statistical Association* 73(363):504–520.

Veach, E., and L. J. Guibas. 1995. "Optimally Combining Sampling Techniques for Monte Carlo Rendering". In *Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques*, edited by S. G. Mair and R. Cook, 419–428. New York: ACM.

AUTHOR BIOGRAPHIES

JING DONG is an Assistant Professor in the Division of Decision, Risk and Operations at Columbia Business School. Her research interests are in applied probability, stochastic simulation and stochastic modeling with applications in service operations management. Her e-mail address is jing.dong@gsb.columbia.edu.

M. BEN FENG is an assistant professor in actuarial science at the University of Waterloo. He earned his Ph.D. in the Department of Industrial Engineering and Management Sciences at Northwestern University. He is an Associate of the Society of Actuaries (ASA). His research interests include stochastic simulation design and analysis, optimization via simulation, nonlinear optimization, and financial and actuarial applications of simulation and optimization methodologies. His e-mail address is ben.feng@uwaterloo.ca.

BARRY L. NELSON is the Walter P. Murphy Professor in the Department of Industrial Engineering and Management Sciences at Northwestern University. He is a Fellow of INFORMS and IIE. His research centers on the design and analysis of computer simulation experiments on models of stochastic systems, and he is the author of *Foundations and Methods of Stochastic Simulation: A First Course*, from Springer. His e-mail address is nelsonb@northwestern.edu.