Proceedings of the 2013 Winter Simulation Conference
R. Pasupathy, S.-H. Kim, A. Tolk, R. Hill, and M. E. Kuhl, eds.

MULTILEVEL MONTE CARLO METAMODELING

Imry Rosenbaum
Jeremy Staum

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

ABSTRACT

Multilevel Monte Carlo (MLMC) methods have been used by the information-based complexity community in order to improve the computational efficiency of parametric integration. We extend this approach by relaxing the assumptions on differentiability of the simulation output. Relaxing the assumption on the differentiability of the simulation output makes the MLMC method more widely applicable to stochastic simulation metamodeling problems in industrial engineering. The proposed scheme uses a sequential experiment design which allocates effort unevenly among design points in order to increase its efficiency. The procedure's efficiency is tested on an example of option pricing in the Black-Scholes model.

1 INTRODUCTION

Simulation has been used extensively to characterize complex stochastic systems. The system performance is a function of some input parameters, e.g., the expected number of people in a multi server queue is a function of the service rates of the servers stationed. One issue that arises in practice in applications such as risk analysis or system design is that we want to estimate performance for a continuum of inputs. For example, estimation of the expected number of people in a G/M/c queue as a function of the service rate would require estimating it for each positive number. Metamodeling schemes, see (Barton 1998; Kleijnen and Sargent 1997), such as stochastic kriging (Ankenman et al. 2010) and kernel smoothing (Hastie et al. 2003) are used to address this issue. Metamodeling allows us to approximate the response surface by fitting a function to the simulation output. Metamodeling could be viewed as function approximation as we are trying to approximate an unknown function. In this paper, we propose a multilevel Monte Carlo approach to metamodeling. It is an enhancement to existing metamodeling schemes in order to reduce their computational effort.

The connection between metamodeling and function approximation is further strengthened by the fact that the expectation of the simulation output is an integral over the sample space. Thus, metamodeling could be viewed as a parametric integration, a subclass of the function approximation problem where we try to approximate an integral as a function of some parameter vector. Therefore efficient Monte Carlo methods for parametric integration could potentially offer increased performance in comparison to existing metamodeling methods. An effective parametric integration technique is the Multilevel Monte Carlo (MLMC) scheme suggested by Heinrich (2000). Heinrich suggests a sequential experiment design with an increasing number of points in which the effort is distributed unevenly among the design points. The standard Monte Carlo method reduces bias in metamodeling by increasing the number of design points and reduces the variance by increasing the number of replications for each design point. In the standard Monte Carlo method the effort is distributed evenly among design points. Heinrich's method breaks the

paradigm that we should allocate the same number of replications to each design point. The purpose is to control the variance. This notion helps to reduce the magnitude of the computational cost. A limitation of using this method is that it requires assumptions on the differentiability of the underlying function. Giles (2008) extended the Multilevel Monte Carlo idea to the simulation of stochastic differential equations. However, Giles does not deal with metamodeling.

In our work we will extend Heinrich's work to a more general setting that requires a weaker assumption on the differentiability of the simulation output. We assume Hölder continuity, a property which is more fitting in most applications of stochastic simulation in industrial engineering and is easier to check. We will present a practical MLMC procedure that will generate a metamodel with a reduced effort in comparison to the standard Monte Carlo method. The outline of our work is as follows. Section 2 introduces the setting of our work and illustrates the effect of our assumptions. Section 3 presents the Multilevel Monte Carlo method and a range of metamodeling schemes that can be used with it. Section 4 analyzes the asymptotic complexity of the MLMC method and compares it to the theoretical performance of the standard Monte Carlo method. Section 5 introduces our MLMC procedure. Section 6 exhibits a numerical study of the performance of the MLMC in the Black-Scholes example.

2 SETTING

Our goal is to estimate the expected performance of a stochastic system, for example the expected cycle time of a tandem queue. Estimation of the expectation of a random variable $Y(\theta)$ could be viewed as an approximation of an integral, because every expectation can be written as an integral over the sample space. Moreover, we are interested in estimating $\mu = \mathbb{E}[Y(\theta)]$ as a function of some input vector θ, for example the service rates of the tandem queue. Therefore, our problem is parametric integration, i.e., estimating μ given the input vector θ.

Next we will present the characteristics of $Y(\theta)$ which we require for our method to be effective. The setting presented is similar to the one employed in the estimation of sensitivities in financial engineering (Glasserman 2003). The simulation output $Y(\theta)$ is a random variable whose output depends on $\theta \in \Theta$, i.e., Y is a measurable function $Y : \Theta \times \Omega \longrightarrow \Re$, where Ω is a sample space with probability measure $\mathbb{P}$ on it and Θ is a compact subset of $\Re^d$. Let us highlight the fact that $\mathbb{E}[Y(\theta_1) - Y(\theta_2)]$ denotes $\int_\Omega (Y(\theta_1,\omega) - Y(\theta_2,\omega))\,\mathbb{P}(d\omega)$. We also require some assumptions on the simulation output. Assume that Y is a Hölder continuous function of a random vector X. In other words, $Y(\theta,\omega) = f(X(\theta,\omega))$, where $f : \Re^d \longrightarrow \Re$ is Hölder continuous function, i.e., there exists c in $\Re_+$ and $\zeta \in (0,1]$ such that

$$|f(U) - f(V)| < c\,\|U - V\|^\zeta, \quad \text{for all } U, V \text{ in } \Re^d.$$

In addition we require that there exist a random variable, κ, with bounded second moment such that

$$||X(\theta_1) - X(\theta_2)|| \leq \kappa \cdot ||\,\theta_1 - \theta_2\,|| \quad \text{almost surely,} \quad \text{for all } \theta_1, \theta_2 \text{ in } \Theta.$$

Now that we have presented our assumptions, we present the structural properties of $Y(\theta)$ that can be derived from our assumptions. In the following sections we will be interested in the bias of estimating $Y(\theta + h)$ using $Y(\theta)$. Our interest in this bias stems from the fact that we will run the simulation at some design points such as θ and then use the simulation output to approximate the response surface at other points such as $\theta + h$. Under our assumptions the bias is

$$\begin{aligned} |\mathbb{E}[Y(\theta+h) - Y(\theta)]| &\leq \mathbb{E}[|Y(\theta+h) - Y(\theta)|] \leq \mathbb{E}\left[c \cdot \|X(\theta+h) - X(\theta)\|^\zeta\right] \\ &\leq \mathbb{E}\left[c \cdot \kappa \|h\|^\zeta\right] = \mathscr{O}\left(\|h\|^\zeta\right). \end{aligned} \tag{1}$$

From (1) it follows immediately that

$$\mathrm{Var}[Y(\theta+h) - Y(\theta)] \leq \mathbb{E}\left[(Y(\theta+h) - Y(\theta))^{2\zeta}\right] = \mathscr{O}\left(\|h\|^{2\zeta}\right). \tag{2}$$

We assume that $\|\Theta\| = 1$ because every compact subset of $\Re^d$ can be normalized via the use of linear transformations.

The figure of merit that will be used in this work is the Mean Integrated Square Error (MISE). It is defined as

$$\text{MISE}\left[\hat{Y}(\theta)\right] = \mathbb{E}\left[\int_{\Theta}\left(Y(\theta)-\hat{Y}(\theta)\right)^2 d\theta\right].$$

Let us now consider a one dimensional example that will help demonstrate the benefit of using the multilevel method. First we introduce the problem in this section and return to it after the presentation of our method.

Example 1 Let $Y(\theta)$ be exponentially distributed with parameter θ. For any $\theta \in \Theta = [1,2]$ we estimate $Y(\theta)$ by segmenting Θ with a equally spaced grid with $q+1$ points where $\theta_0 = 1$ and $\theta_q = 2$. We interpolate between the design points in order to approximate the response surface. In the interpolation we estimate $\hat{Y}(\theta_i)$ by using m replications and then for each θ the estimator is

$$\hat{Y}(\theta) = \hat{Y}(\underline{\theta}) + \frac{\theta-\underline{\theta}}{\overline{\theta}-\underline{\theta}} \cdot \left(\hat{Y}\left(\overline{\theta}\right) - \hat{Y}\left(\underline{\theta}\right)\right)$$

where $\underline{\theta} = \sup_{\theta_i \le \theta} \theta_i$ and $\overline{\theta} = \inf_{\theta_i \ge \theta} \theta_i$. See Figure 1 for illustration of how the interpolation is used in our example. It can be shown that $\text{MISE}\left[\hat{Y}(\theta)\right] \ge \frac{1}{32q^2} + \frac{1}{4m}$. To make it feasible to achieve a MISE of ε^2 we will need to choose $q = \lceil \varepsilon^{-1} \cdot \frac{1}{8} \rceil$ and $m = \lceil \varepsilon^{-2} \cdot \frac{1}{2} \rceil$. Therefore the computational budget will be at least $\mathscr{O}\left(\varepsilon^{-3}\right)$.

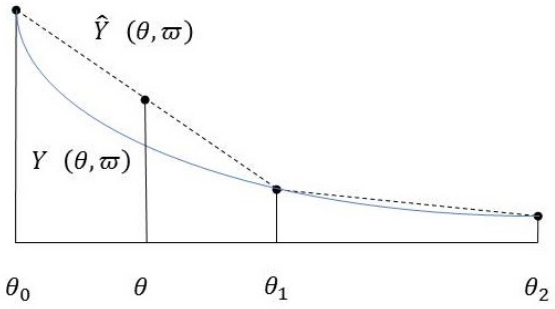

Figure 1: Illustration of the use of interpolation for a given ω

.

3 MULTI LEVEL METHOD

First let us assume that there is a sequence of experiment designs $\{\Delta_\ell\}$ in Θ with an increasing number of points $k^{d(\ell+1)}$. The experiment designs are structured such that $\|\Delta_\ell\|$, the supremum of the minimal distances between a point in Θ and a point in the experiment design, is $\mathscr{O}\left(k^{-\ell\tau}\right)$. Let $\hat{Y}_\ell(\theta,\omega)$ denote an approximation of $Y(\theta,\omega)$ using the ℓth experiment design and the same ω for each of the design points. In practice we will rely on the fact that our random variable can be written as $Y(\theta,U)$ where U is a finite dimension random vector whose components are uniformly distributed over $(0,1)$. Each realization of U could serve as a characterization of our ω. Therefore, the use of common random numbers can be used to simulate the use of the same ω across an experiment design. In other words we generate a realization of U and use it to create a sample of $Y(\theta)$ for each of the design points in the experiment design. Example 1 can be used to illustrate the notion of the approximation. In the example we used interpolation between

$Y\left(\overline{\theta},\omega\right)$ and $Y\left(\underline{\theta},\omega\right)$ as our approximation. For each instance of the approximation we will use common random numbers to generate $Y(\theta)$ for each design point θ_i in the experiment design (as stated previously we use the same realization of U for each point in the experiment design) and interpolate them. The resulting function will serve as the instance of $\hat{Y}(\theta)$. We can write the expectation of $\hat{Y}_\ell(\theta,\omega)$ as

$$\mathbb{E}\left[\hat{Y}_\ell(\theta)\right] = \mathbb{E}\left[\hat{Y}_0(\theta)\right] + \sum_{i=1}^{\ell} \mathbb{E}\left[\hat{Y}_i(\theta) - \hat{Y}_{i-1}(\theta)\right]. \tag{3}$$

As we stated in the previous section our goal is to estimate $\mathbb{E}[Y(\theta)]$, which we denote as $\mu(\theta)$. We define the estimator $\hat{\mu}_\ell(\theta,m,\{\omega_i\})$ as the average of m replications of $\hat{Y}_\ell(\theta,\omega)$, i.e., $\hat{\mu}_\ell\left(\theta,m,\{\omega_i^\ell\}\right) = \frac{1}{m}\sum_{i=1}^{m}\hat{Y}_\ell\left(\theta,\omega_i^\ell\right)$. We will denote $\left\{\omega_i^j\right\}$ as the set of samples used in the jth level. Equation (3) suggests the decomposition

$$\hat{\mu}_\ell\left(\theta,m,\{\omega_i^\ell\}\right) \quad = \quad \hat{\mu}_0\left(\theta,m,\{\omega_i^\ell\}\right) + \sum_{i=1}^{\ell}\hat{\mu}_i\left(\theta,m,\{\omega_i^\ell\}\right) - \hat{\mu}_i\left(\theta,m,\{\omega_i^\ell\}\right).$$

The first term in the expression acts as a baseline and each following term in the sum is a refinement of it. One question that arises is whether each term should use the same m. Changing the number of replications for each term would not change the expectation of the estimator, but it would change its variance. We will use different number of samples for each term in the decomposition. This modification leads us to the construction of the multilevel estimator

$$\hat{Z}_\ell\left(\theta,\{\omega_i^0\},\ldots,\{\omega_i^\ell\}\right) = \sum_{j=0}^{\ell}\Delta\hat{Z}_j\left(\theta,\{\omega_i^j\}\right)$$

where

$$\Delta\hat{Z}_j\left(\theta,\{\omega_i^j\}\right) = \hat{\mu}_j\left(\theta,M_j,\{\omega_i^j\}\right) - \hat{\mu}_{j-1}\left(\theta,M_j,\{\omega_i^j\}\right)$$

and, for ease of notation, we denote $\hat{\mu}_{-1}(\theta,m,\{\omega_i\}) = 0$.

Figure 2 illustrates the relations between $\hat{Y}_\ell(\theta)$, $\hat{Y}_{\ell-1}(\theta)$ and $\hat{Z}_\ell(\theta)$ in Example 1. The dots on the horizontal axis represent the ℓth experiment design while the boxes represent the $\ell-1$th experiment design.

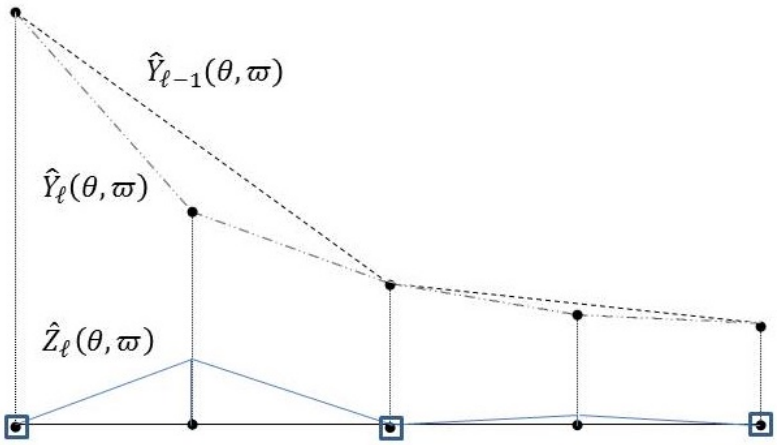

Figure 2: Illustration of $\hat{Y}_{\ell-1}(\theta,\omega)$, $\hat{Y}_\ell(\theta,\omega)$ and $\hat{Z}_\ell(\theta,\omega)$ for a given ω in Example 1.

Next we will show that under some assumptions on $\hat{Y}_\ell(\theta)$ the bias of our approximation decays at the same rate as $\|\Delta_\ell\|^\zeta$. Moreover, we will show that the variance decays at twice the rate of the bias. We focus on metamodeling schemes in which the approximation to $Y(\theta)$ has the form $\hat{Y}_\ell(\theta,\omega) = \sum_{i=0}^{k^{d\ell}} g_\ell\left(\theta-\theta_i^\ell\right)\cdot Y\left(\theta_i^\ell,\omega\right)$, where θ_i^ℓ are the design points. Note that the approximation $\hat{Y}_\ell(\theta,\omega)$ is also a random variable. Let us denote

$$\mathrm{Box}_r(\theta) = \left\{x : \forall i = 1,\ldots,d, \theta_i - \frac{r}{2} \le x_i \le \theta_i + \frac{r}{2}\right\}$$

i.e., a box centered around θ. Such boxes will be referred to as windows. Assume that for each $\xi > 0$ there exists a window size $r(\ell,\xi) > \|\Delta_\ell\|$ which is $\mathscr{O}\left(k^{-\nu\ell}\right)$ such that for each $s \ge r(\ell,\xi)$ we have $0 \le g_\ell(s) \le \left(\frac{\xi}{k^{(d+\nu)\ell}}\right)$ and for each $\theta \in \Theta$ we have

$$1 - \frac{\xi}{k^{\nu\ell}} \le \left|\sum_{\theta_i^\ell \in \mathrm{Box}_{r(\ell,\xi)}(\theta)} g_\ell(\theta-\theta_i)\right| \le 1 + \frac{\xi}{k^{\nu\ell}}.$$

Let us define the set of design points in a window of radius r around θ as $P(\ell,\theta,\xi) = \{\theta_i^\ell : \theta_i^\ell \in \mathrm{Box}_{r(\ell,\xi)}(\theta)\}$. There are numerous examples of estimators with such behavior, for example, k-nearest neighbor or Gaussian kernels (for an overview of both see Hastie et al. 2003). First we inspect the expectation of difference between the estimator and a sample of the true value using the same ω.

$$\begin{aligned}
\mathbb{E}\left|Y(\theta)-\hat{Y}_\ell(\theta)\right| &\le \mathbb{E}\left|Y(\theta) - \sum_{i=0}^{k^{(\ell+1)d}} g_\ell\left(\theta-\theta_i^\ell\right)\cdot Y\left(\theta_i^\ell\right)\right| \le \mathbb{E}\left|Y(\theta) - \sum_{\theta_i^\ell\in P(\ell,\theta,\xi)} g_\ell\left(\theta-\theta_i^\ell\right)\cdot Y\left(\theta_i^\ell\right)\right| \\
&\quad + \frac{\xi\cdot k^d}{k^{\nu\ell}}\cdot \sup_{\theta_i^\ell,\theta\in\Theta} \mathbb{E}\left|Y\left(\theta_i^\ell\right)\right| \\
&\le \mathbb{E}\left|Y(\theta) - \sum_{\theta_i^\ell\in P(\ell,\theta,\xi)} g_\ell\left(\theta-\theta_i^\ell\right)\cdot Y(\theta) + \sum_{\theta_i^\ell\in P(\ell,\theta,\xi)} g_\ell\left(\theta-\theta_i^\ell\right)\cdot Y(\theta)\right. \\
&\quad \left. - \sum_{\theta_i^\ell\in P(\ell,\theta,\xi)} g_\ell\left(\theta-\theta_i^\ell\right)\cdot Y\left(\theta_i^\ell\right)\right| + \frac{\xi\cdot k^d}{k^{\nu\ell}}\cdot \sup_{\theta_i^\ell,\theta\in\Theta} \mathbb{E}\left|Y\left(\theta_i^\ell\right)\right| \\
&\le \mathbb{E}\left|Y(\theta) - \sum_{\theta_i^\ell\in P(\ell,\theta,\xi)} g_\ell\left(\theta-\theta_i^\ell\right)\cdot Y(\theta)\right| \\
&\quad + \mathbb{E}\left|\sum_{\theta_i^\ell\in P(\ell,\theta,\xi)} g_\ell\left(\theta-\theta_i^\ell\right)\cdot Y(\theta) - \sum_{\theta_i^\ell\in P(\ell,\theta,\xi)} g_\ell\left(\theta-\theta_i^\ell\right)\cdot Y\left(\theta_i^\ell\right)\right| + \frac{\xi\cdot k^d}{k^{\nu\ell}}\cdot \sup_{\theta_i^\ell,\theta\in\Theta} \mathbb{E}\left|Y\left(\theta_i^\ell\right)\right| \\
&\le \sum_{\theta_i^\ell\in P(\ell,\theta,\xi)} \mathbb{E}\left|g_\ell\left(\theta-\theta_i^\ell\right)\cdot\left(Y(\theta)-Y\left(\theta_i^\ell\right)\right)\right| + \frac{\xi}{k^{\nu\ell}}|Y(\theta)| + \frac{\xi\cdot k^d}{k^{\nu\ell}}\cdot \sup_{\theta_i^\ell,\theta\in\Theta} \mathbb{E}\left|Y\left(\theta_i^\ell\right)\right| \\
&\le \sum_{\theta_i^\ell\in P(\ell,\theta,\xi)} g_\ell\left(\theta-\theta_i^\ell\right)\cdot \sup_{\theta_j^\ell,\in P(\ell,\theta,\xi)} \mathbb{E}\left|Y(\theta)-Y\left(\theta_j^\ell\right)\right| + \frac{\xi}{k^{\nu\ell}}\mathbb{E}|Y(\theta)| + \frac{\xi\cdot k^d}{k^{\nu\ell}}\cdot \sup_{\theta_i^\ell,\theta\in\Theta} \mathbb{E}\left|Y\left(\theta_i^\ell\right)\right| \\
&= \mathscr{O}\left(k^{-\nu\ell}\right)
\end{aligned}$$

This bound on the rate of decrease stems from the fact that the maximal distance between points in the box $P(\ell,\theta,\xi)$ is $\mathscr{O}\left(k^{-\nu\ell}\right)$ and that $\mathbb{E}\left|Y(\theta)-Y(\theta+h)\right| = \mathscr{O}\left(\|h\|^{\zeta}\right)$. We can now conclude that the rate of decrease of the bias of the estimator is bounded,

$$\left|\mu(\theta)-\mathbb{E}\left[\hat{\mu}_\ell(\theta)\right]\right| = \left|\mathbb{E}\left[Y(\theta)-\hat{Y}_\ell(\theta)\right]\right| \le \mathbb{E}\left[\left|Y(\theta)-\hat{Y}_\ell(\theta)\right|\right] = \mathscr{O}\left(k^{-\zeta\nu\ell}\right). \tag{4}$$

Furthermore, we can deduce that

$$\begin{aligned}
\mathrm{Var}\left[\hat{Y}_\ell(\theta)-Y(\theta)+Y(\theta)-\hat{Y}_{\ell-1}(\theta)\right] &= \mathrm{Var}\left[\hat{Y}_\ell(\theta)-Y(\theta)+Y(\theta)-\hat{Y}_{\ell-1}(\theta)\right] \\
&\le 2\mathrm{Var}\left[\hat{Y}_\ell(\theta)-Y(\theta)\right]+2\mathrm{Var}\left[\hat{Y}_{\ell-1}(\theta)-Y(\theta)\right] \\
&\le 2\mathbb{E}\left[\left(\hat{Y}_\ell(\theta)-Y(\theta)\right)^2\right]+2\mathbb{E}\left[\left(\hat{Y}_{\ell-1}(\theta)-Y(\theta)\right)^2\right] \\
&= \mathscr{O}\left(k^{-2\zeta\nu\ell}\right),
\end{aligned} \tag{5}$$

indicating that the variance of $\hat{Z}_\ell(\theta)$ is $\mathscr{O}\left(\frac{k^{-2\zeta\nu\ell}}{M_\ell}\right)$ for $\ell>0$. The computational cost of the estimator $\hat{Z}(\theta)$ will be $\sum_{\ell=0}^{L} k^{d(\ell+1)}\cdot M_\ell + \sum_{\ell=0}^{L-1} k^{d(\ell+1)}\cdot M_{\ell+1}$.

4 COMPUTATIONAL COMPLEXITY

The advantage of the MLMC is computational efficiency. In the following theorem it is demonstrated what would be the computational complexity of an estimator with structure similar to the MLMC estimator presented in Section 3. This theoretical structure will be the basis upon we build our practical method in Section 5. Let us restate Theorem 1 from Cliffe et al. (2011) in a manner more similar to Theorem 3.1 of Giles (2008).

Theorem 1 Let $\mu(\theta)$ denote a simulation response surface, and $\hat{\mu}_\ell(\theta)$ denote an estimator of the response surface using M_ℓ replications for each design point. Suppose there exist independent estimators $\Delta\hat{Z}_\ell(\theta)$, positive real valued constants $c_1,c_2,c_3,\alpha,\beta,\gamma$ and positive integer $s>1$ such that $\alpha \ge \frac{1}{2}\min(\beta,\gamma)$ and

1. $\left|\mathbb{E}\left[\hat{\mu}_\ell(\theta)-\mu(\theta)\right]\right| \le c_1\cdot s^{-\alpha\ell}$.
2. $\mathbb{E}\left[\Delta\hat{Z}_\ell(\theta)\right] = \begin{cases} \mathbb{E}\left[\hat{\mu}_0(\theta)\right], & \text{if } \ell=0, \\ \mathbb{E}\left[\hat{\mu}_\ell(\theta)-\hat{\mu}_{\ell-1}(\theta)\right], & \text{if } \ell>0. \end{cases}$
3. $\mathrm{Var}\left[\Delta\hat{Z}_\ell(\theta)\right] \le \frac{c_2}{M_\ell}\cdot s^{-\beta\ell}$.
4. The computational cost C_ℓ of $\Delta\hat{Z}_\ell(\theta)$, i.e., the total number of replications used, is bounded by $c_3\cdot M_\ell\cdot s^{\gamma\ell}$.

Then for every $\varepsilon < e^{-1}$ there exist values of L and M_ℓ for which the Mean Square Error of the MLMC estimator $\hat{Z}(\theta)=\sum_{\ell=0}^{L}\Delta\hat{Z}_\ell(\theta)$ is bounded by ε^2 with a computational cost $C=\sum_{\ell=0}^{L}C_\ell$ with bound,

$$C = \begin{cases} \mathscr{O}\left(\varepsilon^{-2}\right), & \text{if } \beta>\gamma, \\ \mathscr{O}\left(\varepsilon^{-2}(\log\varepsilon)^2\right), & \text{if } \beta=\gamma, \\ \mathscr{O}\left(\varepsilon^{-2-\frac{\gamma-\beta}{\alpha}}\right), & \text{if } \beta<\gamma. \end{cases}$$

This theorem deals with the MSE of the estimator $\hat{Z}(\theta)$ of $\mu(\theta)$. However, Proposition 1 implies that guaranteeing a pointwise MSE of ε^2 will guarantee that the MISE will be lower than the target of ε^2. Corollary 2 dictates that if $\sup_\theta \mathrm{MSE}\left[\hat{Y}(\theta)\right] \le Q$ for some real number Q then the $\mathrm{MISE}\left[\hat{Y}(\theta)\right]$ is

less than or equal to Q. Thus, if we can guarantee the above bound on the computational cost for the $\sup_\theta \mathrm{MSE}\left[\hat{Y}(\theta)\right]$, then the MISE will have the same asymptotic bound.

In our setting we choose $s = k^d$ because increase by a factor of k the number of points we use to segment each dimension in subsequent levels. The rate of decay of the bias of estimator $\hat{\mu}_\ell$, α, is dictated by the distance between the points we use for the estimation and the prediction point. Therefore, there will be two factors which will dictate the rate of the bias: the experiment design Δ_ℓ and our metamodeling scheme used in $\hat{Y}_\ell$. The bias of $\hat{\mu}_\ell$ is the same as the bias of $\hat{Y}_\ell$ as $\hat{\mu}_\ell$ is an average of several replications of $\hat{Y}_\ell$. In the previous section we have shown that the bias of $\hat{Y}_\ell$ is $\mathscr{O}\left(k^{-\zeta\nu\ell}\right)$ given that we use $k^{d\ell}$ points. Therefore, α will be equal to $\zeta\nu$ divided by d. However, ν depends on our choice of approximation and experiment designs. First, ν must be smaller or equal to τ, the rate of decay of the bound on the supremum of the minimal distances between a point in Θ and a point in the experiment design, which is dictated by our choice of experiment designs. Moreover, our choice of metamodeling scheme $\hat{Y}_\ell$ affects ν, i.e., the metamodeling scheme we use (e.g., kernel smoothing or k-nearest neighbor) and how we calibrate it will have impact on the performance of the MLMC. For example, the choice of the bandwidth of the kernels will determine ν. The rate of decay of the variance of $\hat{Y}_\ell$ is twice the rate of decay of the bias, i.e., $\beta = 2\alpha$. The cost is $\sum_{\ell=0}^{L} M_\ell \cdot k^{d\ell} + M_\ell \cdot k^{d(\ell-1)}$, which is proportional to $k^{d\ell}$, so $\gamma = 1$. Moreover, the requirement that $\alpha \geq \frac{1}{2}\min(\beta,\gamma)$ always holds, as either $\beta \leq \gamma$ and then $\alpha = \frac{1}{2}\min(\beta,\gamma)$, or $\beta > \gamma$ and then $\alpha = \frac{1}{2}\beta > \frac{1}{2}\min(\beta,\gamma)$. Let us now write the computational cost in terms of ν, ζ and d,

$$C = \begin{cases} \mathscr{O}\left(\varepsilon^{-2}\right), & \text{if } 2\zeta\nu > d, \\ \mathscr{O}\left(\varepsilon^{-2}(\log\varepsilon)^2\right), & \text{if } 2\zeta\nu = d, \\ \mathscr{O}\left(\varepsilon^{-2-\frac{d-2\nu}{\zeta\nu}}\right), & \text{if } 2\zeta\nu < d. \end{cases}$$

Cliffe et al. (2011) show that the computational cost of using single level Monte Carlo is bounded by $\mathscr{O}\left(\varepsilon^{-2-\frac{d}{\zeta\nu}}\right)$. Therefore using the multilevel method reduces the bound on the computational cost by a factor of $\varepsilon^{-\frac{\min(2\zeta\nu,d)}{\zeta\nu}}$. Their proof of the theorem suggests that if the number of levels is

$$L = \left\lceil \frac{d}{\zeta\nu} \log_k\left(\sqrt{2}c_1\varepsilon^{-1}\right) \right\rceil, \tag{6}$$

and the number of replications for each level is

$$M_\ell = \begin{cases} \left\lceil 2\varepsilon^{-2}c_2\left(1-k^{-(2\zeta\nu-d)/2}\right)^{-1}k^{-(2\zeta\nu+d)\ell/2} \right\rceil, & \text{if } 2\zeta\nu > d, \\ \left\lceil 2\varepsilon^{-2}(L+1)c_2k^{-2\zeta\nu\ell} \right\rceil, & \text{if } 2\zeta\nu = d, \\ \left\lceil 2\varepsilon^{-2}c_2k^{(L(d-2\zeta\nu))/2}\left(1-k^{-(d-2\zeta\nu)2}\right)^{-1}k^{-(2\zeta\nu+d)\ell/2} \right\rceil, & \text{if } 2\zeta\nu < d, \end{cases} \tag{7}$$

then these values satisfy the theorem.

It can be shown that for experiment design which is composed of multilevel grids with k^ℓ points in each dimension that $\tau = 1$ because the distance between points in the grid is $\mathscr{O}\left(k^{-\ell}\right)$. Moreover we will assume that we can construct an approximation with the same rate of decay of $r(\ell,\xi)$ as of $\|\Delta_\ell\|^\zeta$. For example, if we use k-nearest neighbors in the case of grids the distance is linearly dependent on the mesh size. Therefore the decay of the window size is the same as the rate of decay of $\|\Delta_\ell\|$. Let us note that

$$M_\ell \propto \sqrt{\mathrm{Var}\left(\hat{Y}_\ell(\theta)\right) \cdot k^{-\ell d}}.$$

We now return to Example 1. Due to the fact that our problem is one dimensional we use the binary Van der Corput sequence to generate experiment designs in $[0,1]$. The computational cost will be bounded by $M_\ell \cdot 2^\ell$, so $\gamma = 1$ and $c_3 = 1$. We can bound the absolute value of the bias of $\hat{Z}_\ell$ with $2 \cdot 2^{-\ell}$ and the variance of $\hat{Z}_\ell(\theta)$ with $\frac{1}{M_\ell} \cdot 2^{-2\ell}$. Therefore, we deduce that $\alpha = 1$, $c_1 = 2$, $\beta = 2$ and $c_2 = 4$. Theorem 1 dictates that the computational complexity for the MLMC will be $\mathcal{O}\left(\varepsilon^{-2}\right)$. By using the MLMC we are able to reduce the rate of increase of the computational cost by a factor of $\varepsilon^{-\frac{\nu\zeta}{d}} = \varepsilon^{-1}$ compared to the cost of the standard Monte Carlo, $\mathcal{O}\left(\varepsilon^{-3}\right)$.

5 Our Procedure

Theorem 1 offers an insight into how the multilevel method can improve the efficiency of the Monte Carlo scheme. However, the performance of the theoretical method in Theorem 1 requires an a priori knowledge of the bounding factors associated with bias and variance (c_1 and c_2 respectively), which is impossible to demand in real life situations. It is possible to overcome the lack of knowledge of the variance factor by padding M_ℓ slightly as will be shown later, but this is not the case for estimation of the bias. Precise estimation of the bias is important as it dictates how many levels should be generated. Estimation of the bias is a known issue in simulation and there is no easy way to tackle it.

Our estimator $\Delta\hat{Z}_\ell$ provides information that will allow us to better approximate the bias. For $\nu = 1$ and $\zeta = 1$ (assuming f is Lipschitz) it is true from the first condition of Theorem 1 that $|\mathbb{E}[\mu(\theta) - \hat{\mu}_\ell(\theta)]| \leq c \cdot k^{-\ell}$ for some constant c. Let us assume that this bound is tight asymptotically, i.e., $|\mathbb{E}[\mu(\theta) - \hat{\mu}_\ell(\theta)]| \approx c \cdot k^{-\ell}$ as $\ell \longrightarrow \infty$. This assumption is applicable in numerous applied settings. Therefore,

$$\left|\mathbb{E}\left[\Delta\hat{Z}_\ell(\theta)\right]\right| = |\mathbb{E}[\hat{\mu}_\ell(\theta) - \hat{\mu}_{\ell-1}(\theta)]| \geq (k-1) \cdot c \cdot \left|k^{-\ell}\right| \approx (k-1) \cdot \mathbb{E}[|\mu(\theta) - \hat{\mu}_\ell(\theta)|].$$

Hence, $\Delta\hat{Z}_\ell(\theta)$ will be used as an upper bound on the bias at level ℓ. We will continue generating levels until the following stopping criterion is met:

$$\left|\Delta\hat{Z}_\ell(\theta)\right| \leq (k-1)\frac{\varepsilon}{\sqrt{2}}.$$

However $\Delta\hat{Z}_\ell$ has an inherent variance, so we recommend using the following stopping criteria as a more cautious approach:

$$\max\left[\left|\Delta\hat{Z}_\ell(\theta)\right|, \frac{1}{k}\left|\Delta\hat{Z}_{\ell-1}(\theta)\right|\right] \leq (k-1)\frac{\varepsilon}{\sqrt{2}}.$$

Note that extra term follows from the belief that the bias should decrease at least by the factor of k.

Our figure of merit is MISE, therefore we would need to approximate the integrated square bias and variance of our estimator at each level. In essence we are trying approximate an integral over Θ. The simplest way to do so is by sampling uniformly over Θ and averaging across the samples. The more samples we use, the more precise our estimation of the MISE will be. These samples act as our prediction points. If we assume that the computational cost of replications is more significant than the calculation of the approximation $\hat{Y}(\theta)$, then such an approach would not change the computational complexity of our algorithm. In order to insure that the asymptotic result of the computational cost of Theorem 1 will hold we average our estimates of the variance and bias across the prediction points. Note that for the bias we are actually interested in the integrated squared bias, so we are averaging over the square of the biases of the prediction points.

The procedure is as follows:

1. **Initialization**
   Set $\ell = 0$.

2. **Variance Estimation**
   For each of the design points in $\ell-1$th experiment design (only if $\ell>0$) and ℓth experiment design we generate M^0 samples using common random numbers. We estimate the variance of $\Delta\hat{Z}_\ell$ for each prediction point and denote the average as V_ℓ.
3. **Sample Calculation**
   Calculate M_ℓ according to the formula $M_\ell = \left\lceil 2\cdot\varepsilon^2\cdot\sqrt{V_\ell\cdot k^{-\ell d}}\cdot\sum_{i=0}^{\ell}\sqrt{\frac{V_i}{k^{-i\cdot d}}}\right\rceil$.
4. **Sampling**
   For each design point in the $\ell-1$th experiment design and ℓ experiment design generate samples using common random numbers, such that the total number of samples for each point will be M_ℓ.
5. **Calculate the Estimator**
   Calculate $\Delta\hat{Z}_\ell(\theta)$ for each prediction point and denote the square root of the average of $\left(\Delta\hat{Z}_\ell(\theta)\right)^2$ as $\Delta\bar{Z}_\ell$.
6. **Test for Convergence**
   Check the condition $\max\left[\left|\Delta\bar{Z}_\ell\right|,\frac{1}{k}\left|\Delta\bar{Z}_{\ell-1}\right|\right]\leq(k-1)\frac{\varepsilon}{\sqrt{2}}$.
7. **Stopping criteria**
   If $\ell<2$ or the condition in the previous step is not met, set $\ell=\ell+1$ and go to step 2. Otherwise, set $L=\ell$.
8. **Update Sampling**
   For $\ell=0,\ldots,L$ we calculate a new M_ℓ (as $\sum_{i=0}^{L}\sqrt{\frac{V_i}{k^{-id}}}$ has changed) and generate additional samples such that the total of samples will be M_ℓ. We update $\hat{Z}_\ell(\theta)$ for each prediction point accordingly and then stop.

M^0 is a parameter set by the user to define what is the minimal number of samples he is willing to use in order to estimate the variance. (Giles 2008) suggested the following value for M_ℓ :

$$M_\ell = \left\lceil 2\cdot\varepsilon^2\cdot\sqrt{V_\ell\cdot k^{-\ell d}}\cdot\sum_{i=0}^{L}\sqrt{\frac{V_i}{k^{-i\cdot d}}}\right\rceil.$$

M_ℓ is defined in such a way to ensure that the sum of our estimates of the variance of $\Delta\hat{Z}_\ell$s would be below our target of $\frac{\varepsilon}{2}$, thus the estimated variance of $\hat{Z}_L(\theta)$ would be below the required threshold. We are taking conservative estimates of the bias and variance of $\hat{Z}_L(\theta)$. However, this over-cautiousness is not as severe as it would have been in the standard Monte Carlo scheme as increasing the number of levels does not increase the computational cost significantly. Furthermore, the construction of the M_ℓs is such that it maintains the structure of the asymptotic result and it is different only in its coefficients. Thus, the computational complexity of the algorithm should be similar to the theoretical one.

6 NUMERICAL ANALYSIS

In this section we present a numerical example that illustrates the advantage of using our algorithm. This example was chosen because it has a closed form solution. Thus, we are better able to test the performance of our method. We compare our method to a single level Monte Carlo method with the same experiment design of the finest grid used by MLMC and an equal number of replication at each design point.

The Black-Scholes model evaluates the price of a European call option, where the underlying asset S follows a geometric Brownian motion with constant volatility σ and drift $r-0.5\sigma^2$. The payoff function of the option can be written as $\max\{0,S-K\}$, where K is the strike price. We are interested in estimating the expectation of the payoff as a function of the strike price. Clearly, the payoff function is a Lipschitz continuous function and S does not depend on K. Thus, the Black-Scholes model response surface meets the necessary requirements for the multilevel method. We use linear interpolation as our metamodeling

scheme. Figure 3 presents the MISE results for different ε with S(0)=10, σ=0.25, $r = 0.05$, M^0=30, $\Theta = [5, 15]$, 50 randomly generated prediction points and 1000 macro replications.

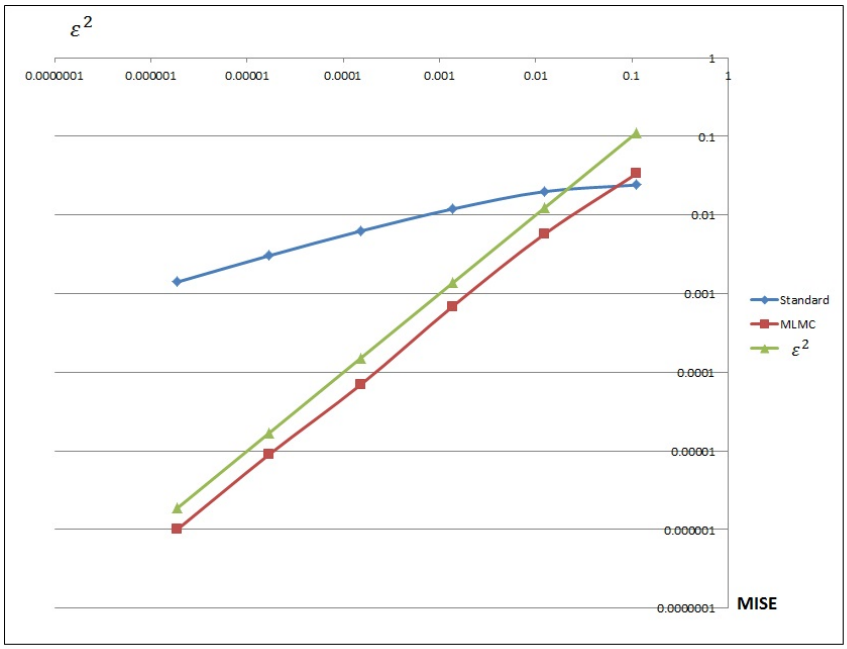

Figure 3: Mean Integrated Squared Error for the MLMC and the standard Monte Carlo.

As can be seen from Figure 3 we achieved MISE lower than the target MISE of ε^2. The MISE of the MLMC method decreases in the expected rate of square power of ε. The single level method is more efficient when not much precision is required but as ε decreases, the MLMC efficiency increases and overtakes the standard method performance.

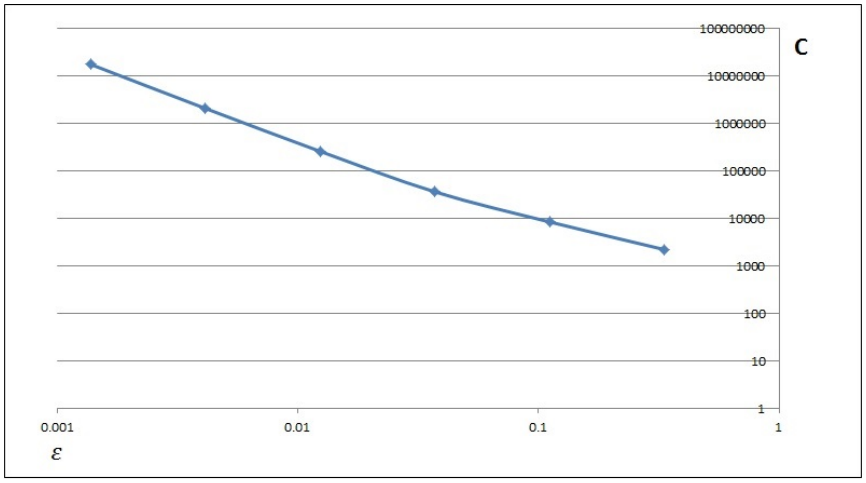

Figure 4: Computational cost of the MLMC as a function of ε

.

Figure 4 illustrates the computational cost, as measured by the total number of replications simulated, in relation to our target precision of ε. As expected according to Theorem 1 the computational cost in the figure has an approximate slope of 2.

ACKNOWLEDGMENTS

This article is based upon work supported by the National Science Foundation under Grant No. CMMI-0900354. The authors are grateful for the reviewers for their corrections and comments that led to improvements in the article.

Appendix

Proposition 1 $\text{MISE} = \int_\Theta \mathbb{E}\left[\left(\mu(\theta) - \hat{\mu}(\theta)\right)^2\right] d\theta = \int_\Theta \text{MSE}(\hat{\mu}(\theta))\, d\theta$.

Proof. We can write the MISE as $\text{MISE} = \mathbb{E}\left[\int_\Theta \left(\mu(\theta) - \hat{\mu}(\theta)\right)^2 d\theta\right]$. Using Tonelli's theorem we can write

$$\text{MISE} = \int_\Theta \mathbb{E}\left[\left(\mu(\theta) - \hat{\mu}(\theta)\right)^2\right] d\theta \quad = \quad \int_\Theta \text{MSE}(\hat{\mu}(\theta))\, d\theta$$

□

Corollary 2 If $\|\Theta\| < \infty$ and $\sup_\Theta \text{MSE}(\hat{\mu}(\theta)) < \infty$ then $MISE(\hat{\mu}(\theta)) \leq C \cdot \sup_\Theta \text{MSE}(\hat{\mu}(\theta))$, where C is a constant.

Proof.

$$\text{MISE} = \int_\Theta \text{MSE}(\hat{\mu}(\theta))\, d\theta \leq \|\Theta\| \cdot \sup_\Theta \text{MSE}(\hat{\mu}(\theta)). \tag{8}$$

□

REFERENCES

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

Barton, R. R. 1998. "Simulation metamodels". In *Proceedings of the 30th conference on Winter simulation*, WSC '98, 167–176. Los Alamitos, CA, USA: IEEE Computer Society Press.

Cliffe, K., M. Giles, R. Scheichl, and A. Teckentrup. 2011, January. "Multilevel Monte Carlo methods and applications to elliptic PDEs with random coefficients". *Computing and Visualization in Science* 14 (1): 3–15.

Giles, M. B. 2008. "Multi-level Monte Carlo path simulation". *Operations Research* 56 (3): 607–617.

Glasserman, P. 2003. *Monte Carlo Methods in Financial Engineering (Stochastic Modelling and Applied Probability) (v. 53)*. 1 ed. Springer.

Hastie, T., R. Tibshirani, and J. H. Friedman. 2003, July. *The Elements of Statistical Learning*. Corrected ed. Springer.

Heinrich, S. 2000. "The multilevel method of dependent tests". In *Advances in Stochastic Simulation Methods*, 47–62.

Kleijnen, J. P., and R. G. Sargent. 1997. "A Methodology for Fitting and Validating Metamodels in Simulation". *European Journal of Operational Research* 120:14–29.

AUTHOR BIOGRAPHIES

IMRY ROSENBAUM is a Ph.D. candidate in the Department of Industrial Engineering and Management Sciences at Northwestern University. His research interest is efficient computer simulation methods, with application to financial models. His e-mail address is imryrosenbaum2016@u.northwestern.edu.

JEREMY STAUM is an Associate Professor of Industrial Engineering and Management Sciences at Northwestern University. He coordinated the Risk Analysis track of the 2007 and 2011 Winter Simulation Conferences and serves as department editor for financial engineering at IIE Transactions and as an associate editor at Management Science. His website is users.iems.northwestern.edu/~staum.