Multilevel Monte Carlo method with application to uncertainty quantification in oil reservoir simulation

Dan Lu, Guannan Zhang, Clayton Webster, Charlotte Barbier

Dan Lu, Computer Science and Mathematics Division, Oak Ridge National Laboratory, Oak Ridge, TN, 37831 (lud1@ornl.gov).

Guannan Zhang, Computer Science and Mathematics Division, Oak Ridge National Laboratory, Oak Ridge, TN, 37831

Clayton Webster, Computer Science and Mathematics Division, Oak Ridge National Laboratory, Oak Ridge, TN, 37831

Charlotte Barbier, Computational Sciences and Engineering Division, Oak Ridge National Laboratory, Oak Ridge, TN, 37831

Abstract. The rational management of oil and gas reservoirs requires understanding of their response to existing and planned schemes of exploitation and operation. Such understanding requires analyzing and quantifying the influence of the subsurface uncertainty on predictions of oil and gas production. As the subsurface properties are typically heterogeneous causing a large number of model parameters, the dimension independent Monte Carlo (MC) method is usually used for uncertainty quantification (UQ). However, the standard MC simulation is computationally expensive because a large number of model executions are required and each model execution is costly simulated on a fine scale spatial grid to ensure accuracy. This study describes a multilevel Monte Carlo (MLMC) method for UQ in reservoir simulation. MLMC is a variance reduction technique for the standard MC. It improves computational efficiency by conducting simulations on a geometric sequence of grids, a larger number of simulations on coarse grids and fewer simulations on fine grids. In this study, we applied the MLMC method to a highly heterogeneous reservoir model modified from the tenth SPE project. We estimated both the expectation and the distribution function of oil production to quantify its uncertainty influenced by the subsurface uncertainty. The results indicate that MLMC can achieve the same accuracy as standard MC with a significantly reduced computational cost, e.g., about 82-97% and 65-97% computational savings in estimating expectations and approximating distribution functions, respectively. The MLMC method is model independent and can be applied in environmental modeling and many other fields.

1. Introduction

{intro}

Reservoir models are routinely employed in prediction of reservoir performance, and in making economic and management decisions. These models describe fluid flow (e.g., oil, gas, and water) through the porous medium, so the porous media properties (e.g., permeability, porosity and capillary) have a significant impact on the model outputs such as mass flow rate and pressure. Almost all the porous media properties used in reservoir simulation are subject to uncertainty. This uncertainty may be quite large, as direct measurements of these properties are available at only a limited number of boreholes and indirect measurements inferred from logs, well tests or reservoir data involve averaged responses over a large scale. Analyzing the uncertainties of these properties and quantification of their influence on predictions of oil and gas production has become increasingly important for petroleum companies to make better field development decisions.

A wildly used approach for addressing uncertainty in reservoir simulation is to treat the under-sampled model parameters (representing one or more porous medium properties) as random fields that satisfy certain statistical correlations. This naturally results in simulating the reservoir using stochastic partial differential equations. Several stochastic techniques like perturbation/moment equation methods [*Zhang*, 2002], generalized polynomial chaos expansions [*Ghanem*, 1998; *Laloy et al.*, 2013], and stochastic collocation method [*Li and Zhang*, 2007; *Lin and Tartakovsky*, 2009; *Zhang et al.*, 2013] have been applied to flow problems in porous media. However, solving the moment equations is very time consuming [*Franssen et al.*, 2009] and it appears that the computation time necessary to solve the equations will be of the same order of magnitude as the time necessary to run a sufficient number of Monte Carlo simulations. Moreover, the Guadagnini and Neuman solutions [*Guadagnini and Neuman*, 1999a, b] developed for the moment equa-

tion may be poor approximations for strongly heterogeneous model because the equations were developed by the perturbation method. The polynomial chaos and stochastic collocation methods are not suitable for strongly heterogeneous problem either. For the highly heterogeneous reservoir, to properly resolve high frequency space fluctuations in model parameters, a fine (high-resolution) numerical grid is usually employed to parameterize the random parameter field. This results in an extremely large number of model parameters especially for a large reservoir domain. Since the computational cost of both polynomial chaos and stochastic collocation methods grows exponentially with the number of model parameters, and truncating to any feasible numbers leads to large systematic errors, the two methods cannot be efficiently and accurately used for the highly heterogeneous problems, like the reservoir model considered in this study.

Hence, standard Monte Carlo (MC) simulation is still chosen in applications. The MC method is very attractive for high-dimensional problems due to its dimension independent convergence rate. In MC simulation, the effect of the uncertain model parameters on predicted oil and gas production is commonly analyzed in the following way: first equally likely random realizations of model parameters are generated; then for each realization the reservoir model is executed to obtain one simulation result of the predictions; and last the predictive results over all realizations are analyzed statistically, for example, calculating the moments, typically the mean and variance, and estimating the distribution functions. To generate the random realizations, the parameter fields must be geostatistically characterized, either based on site characterization data, or by simulating the correlation of the field with a covariance or variogram function. The generated parameter fields should be conditioned to honor measurements data of these parameters. These MC calculations are, however, very computationally expensive not only because one realization is time costly

simulated in a fine numerical grid to reflect the high spatial variation, but also because a large number of such simulations is required to reduce the sampling variance due to the notoriously slow convergence rate of standard MC.

In this study, we address the problem of expensive computational cost in uncertainty quantification for large-scale reservoir models with high dimensional parameter spaces. The approach we used is called multilevel Monte Carlo (MLMC) simulation. The basic idea of MLMC was first introduced by *Heinrich* [2001] to accelerate computation of high-dimensional, parameter-dependent integrals. Similar ideas were used by *Brandt and Ilyin* [2003] to accelerate statistical mechanical calculations. The MLMC method was then extended by *Giles* [2008] to infinite-dimensional integration related to stochastic differential equations arising in mathematical finance. Since then, it has been applied in many areas of mathematics related to differential equations, particularly stochastic differential equations [*Dereich and Heidenreich*, 2011; *Kloeden et al.*, 2011; *Abdulle and Blumenthal*, 2013] and several types of stochastic partial differential equations with random forcing [*Giles and Reisinger*, 2012] or random coefficients [*Barth et al.*, 2011; *Cliffe et al.*, 2011; *Charrier et al.*, 2013; *Teckentrup et al.*, 2013].

The MLMC simulation is shown to be very computationally efficient in solving stochastic differential equations with random parameters, but this state-of-the-art method has never been applied in reservoir simulation to evaluate the uncertainty of the predictions. In addition, currently the MLMC method is mainly applied to estimate the expectation of the interested quantity, and few literature shows its application in estimation of distribution function. To our best knowledge, only *Giles et al.* [2014] discussed the estimation of distribution functions and densities with MLMC in their very recent work. In this study, we apply the MLMC method to a reservoir model with a highly heterogeneous permeabil-

ity field. We are interested in investigating the influence of permeability uncertainty not only on the mean value of the oil production but also on its distribution.

In the estimation of expectation, MLMC exploits the linearity of the expectation, by expressing the quantity of interest on the finest spatial grid (e.g., the grid defined at the measurement scale) in terms of the same quantity on a geometric sequence of coarser grids, each typically twice as fine in each direction as its predecessor. It is easy to understand that defining a numerical model in a finest spatial grid, the simulation error is small but the computational cost of the mode execution is very large. On a coarser grid, the cost is diminished, but also reduced is the simulation accuracy. However, MLMC, by conducting simulations on coarse grids, can achieve the same accuracy as standard MC simulated on the finest grid. The reason is that MLMC conducts the simulations on a sequence of coarse grids, so that the less accurate estimate on the preceding coarser grid can be sequentially corrected by estimates on the following finer grids. The dramatic reduction in cost of MLMC is due to the fact that many realizations are simulated on the computationally frugal coarse grids and a few on the computationally expensive fine grids. In this study, we explain how the computational cost is saved by MLMC and demonstrate the significant computational efficiency of MLMC in estimating expectation.

In the estimation of cumulative distribution function (CDF), we first redefine the CDF as an expectation of an indicator function, and then use MLMC to estimate the expectation. The estimation of CDF exhibits two features: (1) we estimate the expectation of an indicator function instead of just a real number; and (2) the indictor function has a singularity. To deal with the first feature, we first use MLMC to estimate the CDF at discrete points, and then extend to a function on a compact interval by interpolation. For the second feature, to overcome the influence of the singularity and for MLMC to work

well, we introduce a smoothing step, i.e., using a smoothing function to approximate the indicator function first and then using MLMC to estimate the expectation of the smoothing function. In this study, we develop an algorithm of CDF estimation with MLMC. We demonstrate that the algorithm can efficiently and reliably estimate the CDF by achieving the desired estimation accuracy with significantly less computational time than standard MC.

The rest of the paper is organized as follows. In Section 2, the stochastic problem is defined. In Section 3, we discuss the estimation of expectation through MC simulation, where we focus on the computational comparison between the standard MC and MLMC. In Section 4, both standard MC and MLMC are used to estimate the CDF where a reliable and efficient algorithm of MLMC estimation is introduced. In Section 5, the MLMC approach is applied to a reservoir model for estimation of the expectation and CDF of mass flow rate at a production well, and the results are discussed in Section 6. Finally, in Section 7 we give our conclusions.

The main novelty of our work lies in application of the state-of-the-art MLMC method to a challenging computationally demanding and highly parameterized real-world reservoir problem. This paper therefore plays an important role in bridging the gap between earlier theoretical work on relatively simple simulation models [*Cliffe et al.*, 2011] and more complex real-world inference problems. More importantly, we develop an efficient and reliable algorithm to estimate the distribution function using MLMC, which is rather new and have not been applied before.

2. Problem Formulation

{s:method}

Consider a immiscible and incompressible two-phase flow reservoir equation on a d-dimensional bounded domain $\Omega \subset \mathbb{R}^d (d \leq 3)$ with boundary Γ

$$\frac{\partial(\phi\rho_a s_a)}{\partial t} + \nabla \cdot (\rho_a \mathbf{v}_a) - q_a = 0, \text{ where } \mathbf{v}_a = -\frac{\mathbf{k}k_{ra}}{\mu_a}(\nabla p_a - \rho_a G),$$

subject to initial and boundary conditions

$$s_a(\mathbf{x}, 0) = s_a^0; \tag{1}$$

$$p_a = p_a^* \text{ on } \Gamma_D; \; s_a = s_a^* \text{ on } \Gamma_D; \; \mathbf{n} \cdot \mathbf{v}_a = \mathbf{v}_a^* \text{ on } \Gamma_N.$$

{problem}

where a denotes the two phases (e.g., water and oil), $\rho_a, s_a, \mathbf{v}_a, q_a, k_{ra}, \mu_a$ and p_a are the density, saturation, Darcy velocity, source/sink term, relative permeability, viscosity and pressure of the a-phase, respectively, and G is the gravity vector. Here, porosity ϕ and permeability tensor $\mathbf{k}$ are uncertain porous medium properties of a heterogeneous subsurface environment Ω. Flow, e.g., changes in pressure and mass flow rate $Q = \rho_a \mathbf{v}_a A$ are affected by these uncertain properties, where A is across-sectional area. Initial saturation distribution is s_a^0 and boundary condition is jointly prescribed by the Dirichlet (Γ_D) and Neumann (Γ_N) segments of the boundary $\Gamma = \Gamma_D \cup \Gamma_N$ with the unit normal vector $\mathbf{n}$.

In a typical application, the spatially varying parameters, e.g., permeability $\mathbf{k}(\mathbf{x})$, are measured at limited ns locations $\mathbf{x}_i, i = 1, 2, ..., ns$, throughout the domain Ω, where Ω is discretized into $\bar{M}$ cells and represented by $\mathbf{x}_{\bar{M}}$ at measurement scale. Uncertainty about parameter values at points $\mathbf{x}_j \in \mathbf{x}_{\bar{M}}$ where measurements are not available (i.e., $\mathbf{x}_j \neq \mathbf{x}_i$) can be quantified by treating these parameters as random fields, e.g., permeability field $\mathbf{k}(\mathbf{x}_{\bar{M}})$, whose ensemble statistics are inferred from available data. In this sense, the corresponding problem defined in equation (1) becomes stochastic. Its solutions are given in terms of probability density functions (PDFs) or cumulative distribution functions (CDFs) of the system states such as pressure and mass flow rate at some locations and specific times.

In this study, we are interested in investigating the influence of uncertain permeability field $\mathbf{k}(\mathbf{x}_{\bar{M}})$ on mass flow rate Q at a point $(\mathbf{x}, t)$, specifically, in estimating the expectation $\mathbb{E}[Q]$ and the distribution function F using Monte Carlo simulation. Here, $\mathbf{k}(\mathbf{x}_{\bar{M}})$ and Q are random variables defined at the measurement scale. Practically, the numerical grid at the measurement scale is too fine to make the MC simulation affordable. Therefore, the MC simulation is commonly performed on a relatively coarse numerical scale generating samples of Q_M, where M represents grid size at the numerical scale. Then by estimating the expectation and distribution function of Q_M, we approximate $\mathbb{E}[Q]$ and F. We assume that as $M \to \bar{M}$, $\mathbb{E}[Q_M] \to \mathbb{E}[Q]$ and $F \to F_M$. Define the convergence rate as α, then

$$|\mathbb{E}[Q_M - Q]| = c_1 M^{-\alpha}, \tag{2}$$ {alpha}

for a constant c_1. The convergence of F_M can be defined in the similar way.

3. Estimation of Expectation

{s:expectation}

In this section, we discuss the estimation of $\mathbb{E}[Q]$ using MC simulations. We first review the standard MC estimator of $\mathbb{E}[Q]$ briefly and then move to introduce the MLMC estimator of $\mathbb{E}[Q]$ in detail. We focus on comparing the computational cost between these two estimators.

3.1. Standard Monte Carlo simulation for estimating expectation

{s:mc}

Based on the definition of Anderson[*Anderson*, 1999], Monte Carlo is the art of approximating an expectation by the sample mean of a function of simulated random variables. Mathematically, the standard MC estimator for $\mathbb{E}[Q_M]$ is

$$\hat{Q}_M^{MC} = \frac{1}{N_{MC}} \sum_{i=1}^{N_{MC}} Q_M^{(i)}, \tag{3}$$ {mc}

where $Q_M^{(i)}$ is the ith sample of Q_M and N_{MC} is the total number of independent samples. Note that $\hat{Q}_M^{MC}$ is an unbiased estimator for $\mathbb{E}[Q_M]$, i.e., $\mathbb{E}[\hat{Q}_M^{MC}] = \mathbb{E}[Q_M]$.

Using equation (3) to estimate $\mathbb{E}[Q]$ includes two sources of error: (1) the approximation of $\mathbb{E}[Q]$ by $\mathbb{E}[Q_M]$, where the error is related to the spatial discretization in our problem (i.e., the difference between the measurement scale $\bar{M}$ and the numerical scale M), and (2) the estimation of the expected value $\mathbb{E}[Q_M]$ by a finite sample average $\hat{Q}_M^{MC}$, where the difference is caused by the MC sampling error. Here we use mean square error (MSE) to measure the accuracy of the standard MC estimator and analyze the contribution of both errors.

The MSE is defined as:

$$\begin{aligned} e(\hat{Q}_M^{MC})^2 &= \mathbb{E}\left[\left(\hat{Q}_M^{MC} - \mathbb{E}[Q]\right)^2\right] \\ &= \mathbb{E}\left[\left(\hat{Q}_M^{MC} - \mathbb{E}[\hat{Q}_M^{MC}]\right)^2\right] + \left(\mathbb{E}[\hat{Q}_M^{MC}] - \mathbb{E}[Q]\right)^2 \\ &= \mathbb{V}[\hat{Q}_M^{MC}] + \left(\mathbb{E}[\hat{Q}_M^{MC}] - \mathbb{E}[Q]\right)^2. \end{aligned} \tag{4}$$

Since $\mathbb{V}[\hat{Q}_M^{MC}] = N_{MC}^{-1}\mathbb{V}[Q_M]$ and $\mathbb{E}[\hat{Q}_M^{MC}] = \mathbb{E}[Q_M]$, we get

$$e(\hat{Q}_M^{MC})^2 = N_{MC}^{-1}\mathbb{V}[Q_M] + (\mathbb{E}[Q_M - Q])^2 . \tag{5}$$

The first term in the MSE of equation (5) is the variance of the MC estimator, which is small as $\mathbb{V}[Q_M]$ is small and decays inversely with the number of samples N_{MC}. The second term is the square of the error in expectation between Q_M and Q, which can be reduced by using a fine grid with large grid size M. When the grid size M is the same as $\bar{M}$, the second term is zero.

Hence, a sufficient condition to achieve a root mean square error (RMSE) of ε with the MC estimator is that both of the terms in equation (5) are less than $\varepsilon^2/2$. This can be achieved by choosing $N_{MC} \geq 2V[Q_M]\varepsilon^{-2}$ and $M \geq (\varepsilon/(\sqrt{2}c_1))^{-1/\alpha}$ where α and c_1

are defined in equation (2). When $M = \bar{M}$, N_{MC} can be chosen as $N_{MC} \geq \mathbb{V}[Q_M]\varepsilon^{-2}$. Therefore, for the estimator $\hat{Q}_M^{MC}$ being a sufficiently accurate approximation of $\mathbb{E}[Q]$ with small ε, a large enough number of samples needs to be simulated on a sufficient fine grid. This results in huge computational cost.

Assume that the cost to compute one sample of Q_M is $\mathcal{C}_M = c_2 M^\gamma$ for a constant c_2 and some $\gamma > 0$, the total cost of the standard MC estimator with N_{MC} samples is

$$\mathcal{C}(\hat{Q}_M^{MC}) = c_2 N_{MC} M^\gamma. \tag{6}$$ {cmc0}

The γ is the rate of cost increase with M and its value depends on the efficiency of the numerical model solver; an efficient solver generally causes a small γ. The γ value can be calculated by fitting the curve of $\mathcal{C}_M$ with changes of M, where $\mathcal{C}_M$ can be measured by the computational time. For a certain numerical solver, the total computational cost of the standard MC estimator increases with N_{MC} and M. To reduce the cost, the N_{MC} and M should be small, but small N_{MC} and M cause large MSE. So an efficient and effective estimator is desired to reduce the computational cost without satisfying the accuracy.

3.2. Multilevel Monte Carlo simulation for estimating expectation

{s:mlmc_exp}

This section describes the MLMC simulation for estimating $\mathbb{E}[Q]$. First the idea of MLMC simulation is introduced, followed by analyzing its computational cost savings compared to standard MC, at last the algorithm of MLMC for expectation estimation is presented.

3.2.1. The idea of MLMC

{s:mlmc1}

The basic idea of MLMC simulation is simple. It exploits the linearity of expectation. Instead of estimating $\mathbb{E}[Q]$ by $\mathbb{E}[Q_M]$ on a single fine grid with size M, MLMC estimates $\mathbb{E}[Q]$ using several $\mathbb{E}[Q_{M_\ell}]$ based on an increasing sequence of relatively coarse grids with

size M_ℓ, i.e., $M_0 < M_1 < ... < M_L = M$. Here $\ell = 0, 1, ..., L$ is called levels and the highest level L has the grid size M. The relation of the grids between two consecutive levels is $M_{\ell-1} = s^{-1} M_\ell$ for $\ell \geq 1$, where s^{-1} is the factor by which the grid is coarsened at a lower level and s is an integer greater than 1. The value of s affects the efficiency of MLMC method and usually $s = 2^d$ is an optimal choice for our stochastic problem with random coefficients according to *Giles* [2008], here d is the number of spatial dimensions of the simulation domain.

Mathematically, the idea of MLMC can be formulated as:

$$\mathbb{E}[Q_M] = \mathbb{E}[Q_{M_0}] + \sum_{\ell=1}^{L} \mathbb{E}[Q_{M_\ell} - Q_{M_{\ell-1}}] = \sum_{\ell=0}^{L} \mathbb{E}[Y_\ell], \tag{7}$$

where $Y_0 = Q_{M_0}$ and $Y_\ell = Q_{M_\ell} - Q_{M_{\ell-1}}$ for $\ell \geq 1$. Hereinafter, for notational simplicity we represent Q_{M_ℓ} with Q_ℓ for $\ell \geq 0$. Equation (7) indicates that the expectation on the finest level grid M is equal to the expectation on the coarsest level grid M_0 plus a sum of difference in expectation between simulations on consecutive levels. In this way, the less accurate estimate on the coarsest grid is sequentially corrected by the estimates on the following finer grids, thereby achieving the finest grid accuracy. In MLMC, each of the expectations on the right-hand side of equation (7) is independently estimated by standard MC (equation (3)) using different number of samples. For example, let $\hat{Y}_0$ be an unbiased standard MC estimator for $\mathbb{E}[Y_0]$ using N_0 samples, and let $\hat{Y}_\ell$ for $\ell \geq 1$ be the MC estimator for $\mathbb{E}[Y_\ell]$ using N_ℓ samples, then we have

$$\hat{Y}_0 = \frac{1}{N_0} \sum_{i=1}^{N_0} Q_0^{(i)} \text{ and } \hat{Y}_\ell = \frac{1}{N_\ell} \sum_{i=1}^{N_\ell} \left(Q_\ell^{(i)} - Q_{\ell-1}^{(i)} \right). \tag{8}$$

The MLMC estimator for $\mathbb{E}[Q_M]$ is simply

$$\hat{Q}_M^{ML} = \sum_{\ell=0}^{L} \hat{Y}_\ell, \tag{9}$$

where $\hat{Y}_\ell$ is calculated based on above equation (8). It is important to note that the quantities $Q_\ell^{(i)}$ and $Q_{\ell-1}^{(i)}$ in equation (8) are the samples from the same set of realizations of $\boldsymbol{X}$ but simulated on different levels of grid, i.e., on grids M_ℓ and $M_{\ell-1}$, respectively.

3.2.2. The computational cost of MLMC

{s:cost}

Since all the expectations on the right-hand side of equation (7) is independently estimated by the standard MC, the variance of the MLMC estimator is $\mathbb{V}[\hat{Q}_M^{ML}] = \sum_{\ell=0}^{L} N_\ell^{-1}\mathbb{V}[Y_\ell]$. We expand the MSE of the MLMC estimator as the way in equations (4) and (5), we get

$$e(\hat{Q}_M^{ML})^2 = \sum_{\ell=0}^{L} N_\ell^{-1}\mathbb{V}[Y_\ell] + \left(\mathbb{E}[Q_M - Q]\right)^2 . \tag{10}$$

{mlmc_mse}

Equation (10) indicates that the MSE of the MLMC estimator, like the standard MC estimator, consists of two terms, the variance of the estimator due to the sampling error and the approximation error due to domain spatial discretization. In comparison between equations (5) and (10), the second term is exactly the same and its accuracy depends on the value of M. Like the standard MC case, when the grid size M is the same as $\bar{M}$, the second term is zero. In fact, it is the first term that reflects the advantages of the MLMC estimator in achieving the same accuracy as the standard MC but with less computational cost.

For example, we limit the first term of MSE in both equations (5) and (10) as ε_1. To achieve this accuracy, the number of samples N_{MC} in equation (5) simulated on the grid M should be larger than $\mathbb{V}[Q_M]/\varepsilon_1$. Correspondingly, the total cost of standard MC defined in equation (6) is at least

$$\mathcal{C}(\hat{Q}_M^{MC}) = c_2 V[Q_M]/\varepsilon_1 M^\gamma. \tag{11}$$

{cmc}

For MLMC, assume the cost to compute one sample of Y_ℓ on grid M_ℓ is $\mathcal{C}_{M_\ell} = c_2 M_\ell^\gamma$, then the total cost of MLMC is

$$\mathcal{C}(\hat{Q}_M^{ML}) = c_2 N_0 M_0^\gamma + c_2 \sum_{\ell=1}^{L} N_\ell M_\ell^\gamma. \tag{12}$$ {cmlmc0}

To limit the first term of equation (10) to ε_1, a sufficient condition is that the value of $N_\ell^{-1}\mathbb{V}[Y_\ell]$ at each level ℓ is less than $\varepsilon_1/(L+1)$, i.e., the number of samples N_ℓ is greater than $\mathbb{V}[Y_\ell](L+1)/\varepsilon_1$. Assume that $\mathbb{V}[Q_\ell]$ is approximately constant at different levels, i.e., $\mathbb{V}[Y_0] = \mathbb{V}[Q_0] \approx \mathbb{V}[Q_M]$ (this assumption is verified in our numerical examples in Section 5), above equation (12) becomes

$$\mathcal{C}(\hat{Q}_M^{ML}) = c_2 \mathbb{V}[Q_M](L+1)/\varepsilon_1 {M_0}^\gamma + c_2 \sum_{\ell=1}^{L} \mathbb{V}[Y_\ell](L+1)/\varepsilon_1 {M_\ell}^\gamma. \tag{13}$$

The cost of MLMC defined in equation (13) is cheaper than that of standard MC defined in equation (11) reflected in the following two aspects. (1) At level $\ell = 0$, all the samples are simulated on the coarsest grid M_0, and the computational cost savings of the coarsest grid M_0 over the grid M is about $(L+1)(M_0/M)^\gamma$. Depending on the value of γ and how we coarsen the grids, the cost savings can be significantly huge. For example, if we coarsen the grid M with factor $s = 2^d$, i.e., $M_0 = (1/2)^L M$ for 1D problem and $M_0 = (1/2)^{2L} M$ for 2D problem, then for $L = 3$ and $\gamma = 1$, the cost of MLMC at $\ell = 0$ is about 1/2 of standard MC in 1D and 1/16 in 2D. With larger value of γ, the cost savings can be improved further. (2) At levels $\ell \geq 1$, the number of samples N_ℓ for each level ℓ is significantly smaller than N_{MC} due to the smaller value of $\mathbb{V}[Y_\ell]$ compared to $\mathbb{V}[Q_M]$. It can be shown that if Q_M converges to Q in mean square, then $\mathbb{V}[Y_\ell] \to 0$ as $\ell \to \infty$. We assume the convergence rate of $\mathbb{V}[Y_\ell]$ with ℓ can be measured by a positive value β, i.e., $\mathbb{V}[Y_\ell] = c_3 M_\ell^{-\beta}$. For a large value of β, the computational saving of MLMC can be substantial. The β depends on how sensitive the interested quantity Q

to the parameter field $\mathbf{X}_{M_\ell}$ defined in different levels; a small sensitivity of Q generally causes a large value of β and correspondingly a fast variance decay. For example, in the groundwater simulation of *Cliffe et al.* [2011], they investigated the influence of uncertain conductivity field on a cumulative outflow from a region and a flux rate at the center of the region using MLMC. Their results indicated that the estimation of expectation of the cumulative outflow gives a larger cost saving than that of the flux rate at the center of the region. Practically, β can be calculated by fitting the curve of $\mathbb{V}[Y_\ell]$ with changes of M_ℓ.

To sum up, if the variance $\mathbb{V}[Y_\ell]$ decays faster with ℓ than the cost $\mathcal{C}_{M_\ell}$ increases, i.e., if $\beta > \gamma$, the first term of equation (13) at level $\ell = 0$ will be dominant. In this case, the cost savings of MLMC compared to standard MC will be asymptotically $(M_0/M)^\gamma$, reflecting the ratio of the costs of simulations on the coarsest grid compared to those on the finest grid used. On the other hand, if the variance $\mathbb{V}[Y_\ell]$ decays slower than $\mathcal{C}_{M_\ell}$ increases, i.e., if $\beta < \gamma$, the second term of equation (13) at the highest level L will be dominant, and the cost savings of MLMC compared to standard MC will be asymptotically $\mathbb{V}[Y_L]/\mathbb{V}[Q_M]$. Therefore, in both cases the MLMC estimator significantly improves the computational efficiency. *Giles* [2008] and *Cliffe et al.* [2011] presented a theorem about how much efficiency can be improved by the MLMC estimator for different relations between β and γ.

3.2.3. The procedure of MLMC for estimating expectation

{s:procedure1}

Before we step into the procedure of the MLMC simulation, we first discuss three fundamental issues related to the MLMC algorithm: (1) how to choose the coarsest level of grid M_0 to obtain the largest cost savings, (2) how many levels should be used to

achieve a desired accuracy, and (3) what is the optimal number of samples for each level to minimize the overall variance for a fixed cost.

In MLMC theory, there is no requirement of the choice of M_0. In practical application, generally the coarsest grid M_0 can be chosen as small as the simulation allows. The coarser the grid is, the more levels can be included in the MLMC estimator, and thus the larger the potential cost savings are with respect to the standard MC. In our problem, this choice depends on the regularity of the covariance function of the permeability field and on the correlation length. The permeability field considered in this study varies on a very fine scale and is highly oscillatory due to the sparse measurement data, so when the grid is coarse enough that the length of a grid cell is larger than the correlation length, this coarse grid does not yield a good representation of the problem and including it in the MLMC estimator can lead to a larger cost than necessary. For example, when the coarsest grid M_0 is coarse enough, it is possible that the variance $\mathbb{V}[Y_\ell]$ at a certain level ℓ is larger than $\mathbb{V}[Q_M]$. In this situation, the contributions to the computational cost of the MLMC estimator from the level ℓ and lower levels will be greater than those using the standard MC, rendering the inclusion of the lower levels meaningless. As discussed in Section 5.2, it turns out that for our problem the optimal choice for the coarsest grid is such that its cell length is slightly smaller than the correlation length of the permeability field.

The answers to the last two questions are related to the desired accuracy of the MLMC estimator. To achieve a MSE of ε^2, both terms in equation (10) have to be less than $\varepsilon^2/2$. According to *Giles* [2008], to make the second term in equation (10) smaller than $\varepsilon^2/2$, the level L is suggested increasing till

$$\max\left\{s^{-1}|\hat{Y}_{L-1}|, |\hat{Y}_L|\right\} < \varepsilon/\sqrt{2}(s-1), \tag{14}$$

where s is the factor discussed at the beginning of Section 3.2.1, and ε is a user-specified RMSE accuracy.

According to *Giles* [2008] and *Cliffe et al.* [2011], treating the N_ℓ as continuous variables, the variance of the MLMC estimator (i.e., the first term in equation (10)) is minimized for a fixed computational cost by choosing

$$N_\ell = c_4 \sqrt{\mathbb{V}[Y_\ell]/\mathcal{C}_{M_\ell}}, \tag{15}$$ {nl}

where c_4 is a constant and $\mathcal{C}_{M_\ell} = c_2 M_\ell^\gamma$ as defined before. To make $\sum_{\ell=0}^{L} N_\ell^{-1}\mathbb{V}[Y_\ell] \leq \varepsilon^2/2$, the optimal N_ℓ is chosen as

$$N_\ell \geq 2\varepsilon^{-2}\sqrt{\mathbb{V}[Y_\ell]/M_\ell^\gamma}\left(\sum_{\ell=0}^{L}\sqrt{\mathbb{V}[Y_\ell]M_\ell^\gamma}\right), \tag{16}$$ {term1}

when $M_L < \bar{M}$ according to equation (15). When $M_L = \bar{M}$, the second term in equation (10) is zero and the first term less than ε^2 can achieve the RMSE accuracy ε. So in this case,

$$N_\ell \geq \varepsilon^{-2}\sqrt{\mathbb{V}[Y_\ell]/M_\ell^\gamma}\left(\sum_{\ell=0}^{L}\sqrt{\mathbb{V}[Y_\ell]M_\ell^\gamma}\right). \tag{17}$$ {term12}

Putting together the elements discussed above, the MLMC algorithm to estimate expectations can be implemented in practice as follows:

Algorithm I:

1. Start with $L = 0$;
2. Estimate $\mathbb{V}[Y_L]$ by the sample variance using an initial set of $N_L = 1000$ samples;
3. Determine the optimal N_ℓ for each level $\ell = 0, 1, ..., L$ based on equation (16) or (17);
4. Evaluate extra samples Y_ℓ at each level as needed for new N_ℓ;
5. If $L \geq 1$, test for convergence using equation (14);

6. If $L < 1$ or it is not converged, set $L = L + 1$ and go back to step 2.

7. Stop when converged or the grid size M_L hits the measurement scale size $\bar{M}$.

Note that in above algorithm, for each itexration with L increasing the N_ℓ increases for all levels based on equation (16). So in step 4, the appropriate number of increased additional samples needs to be evaluated and the estimate of $\mathbb{V}[Y_\ell]$ needs to be updated. The updated $\mathbb{V}[Y_\ell]$ is then used in step 3 to calculate N_ℓ if step 3 is revisited. In addition, the accuracy of the estimate of $\mathbb{V}[Y_\ell]$ at each level depends on the size of the initial sample set. The large the sample size, the more accurate the estimate and in the meantime the larger the computational cost. It is possible that for a large value of ε, the initial N_L at a higher level L may be larger than necessary causing a waste of computational time. So it is suggested adjusting the initial sample set N_L in above step 2 based on the user-specify accuracy of ε.

4. Estimation of Distribution Function

{s:cdf}

In this section, for the continuous random variable Q, we discuss the estimation of its cumulative distribution function (CDF) F on a compact interval $[U_0, U_1]$, with $U_0 < U_1$ being fixed throughout this section.

In definition, the CDF of a continuous random variable Q can be expressed as an integral of its probability density function $p(Q)$ as follows:

$$F(q) = \int_{-\infty}^{q} p(Q)dQ. \tag{18}$$

This integral can be formulated as an expectation of an indicator function $f(q)$ in the following way,

$$F(q) = \mathbb{E}[f(q)] = \int_{-\infty}^{+\infty} f(q)p(Q)dQ, \text{ where } f(q) = \begin{cases} 1, \text{ for } [-\infty, q] \\ 0, \text{ for } (q, +\infty] \end{cases} \tag{19}$$

Thus, we can use standard MC and MLMC simulation as discussed above to estimate the CDF, $F(q)$.

4.1. Standard Monte Carlo simulation for estimating CDF

{s:mc_cdf}

To estimate $F(q)$ on a compact interval $[U_0, U_1]$, the algorithm of standard MC works as follows: first generate N_{MC} independent samples Q_M on a certain numerical grid M; then use sample mean of the indicator function $f(q)$ (i.e., the empirical distribution function of Q_M) for $F(q)$ estimation at finite t points with $U_0 = u_1 < u_2 < \cdots < u_t = U_1$, that is,

$$F_{M,t}^{MC}(q) = \frac{1}{N_{MC}} \sum_{i=1}^{N_{MC}} 1\{Q_M^{(i)} \leq q\} = \frac{\text{Number of samples } \leq q}{\text{Total number of samples } N_{MC}}, \tag{20}$$ {mc_cdf}

where $q = \{u_1, u_2, \cdots, u_t\}$; and last use interpolation to get the entire CDF estimation.

The standard MC estimator $F_{M,t}^{MC}(q)$ has two main sources of errors, the interpolation error of using F_t to approximate F and the MC error in estimation of F_t with the MC estimator $F_{M,t}^{MC}$, where the MC error, like equation (5), can be further expanded into two parts, the sampling error and spatial discretization error. To sum up, the mean square error of $F_{M,t}^{MC}$ can be expressed as,

$$\begin{aligned} \mathbb{E}\|F - F_{M,t}^{MC}\|_\infty^2 &\leq \underbrace{\mathbb{E}\|F - F_t\|_\infty^2}_{e_1^{MC}} + \mathbb{E}\|F_t - F_{M,t}^{MC}\|_\infty^2 \\ &\leq e_1^{MC} + \underbrace{\|\log t N_{MC}^{-1} \mathbb{V}[f_{M,t}]\|_\infty}_{e_2^{MC}} + \underbrace{\|\mathbb{E}[f_{M,t} - f_t]\|_\infty^2}_{e_3^{MC}}, \end{aligned} \tag{21}$$

where $\|\cdot\|_\infty$ denotes the maximum norm on $\mathbb{R}^t$. When the numerical grid M is chosen as the same as measurement scale $\bar{M}$, the discretization error $e_3^{MC} = 0$. In this case, a sufficient condition to achieve a RMSE accuracy of ε is that both interpolation error e_1^{MC} and sampling error e_2^{MC} are less than $\varepsilon^2/2$.

When cubic spline interpolation is used at t equidistant points between $[U_0, U_1]$, the interpolation error $\|F - F_t\|_\infty$ is bounded by $\frac{\|F^{(4)}(x)\|_\infty}{4!} \triangle x^4$, where $\triangle x = (U_1 - U_0)/(t -$

1). So to achieve the interpolation error $e_1^{MC} = \|F - F_t\|_\infty^2 \leq \varepsilon^2/2$, t can be chosen as $t \geq (U_1 - U_0)/(\frac{24\epsilon}{\sqrt{2}\|F^{(4)}(x)\|_\infty})^{1/4} + 1$. To achieve the sampling error $e_2^{MC} \leq \varepsilon^2/2$, N_{MC} can be chosen as $N_{MC} \geq 2\varepsilon^{-2}\log t\|\mathbb{V}[f_{M,t}]\|_\infty$.

4.2. Multilevel Monte Carlo simulation for estimating CDF

mlmc_cdf}

When using MLMC to estimate the CDF, i.e., $\mathbb{E}[f(q)]$, a smoothing step is necessary on every level for computational efficiency. As discussed above, the dramatic computational cost savings of MLMC are due to the fact that the variance at each level ℓ for $\ell \geq 1$ (i.e., the variance of difference of estimated quantity between ℓ and $\ell-1$) is smaller than that in standard MC (i.e., the variance of estimated quantity itself). For the indicator function $f(q)$ the variance at location q is large because of the discontinuity and application of MLMC to $f(q)$ directly would barely improve computational efficiency. So, we first use a smoothing function to approximate $f(q)$ to reduce the variance, and then apply MLMC to estimate the expectation of the smoothing function to obtain the CDF estimation.

We assume the density function $p(Q)$ is r-times continuously differentiable on $[U_0, U_1]$. According to *Giles et al.* [2014], the smoothing function is a rescaled translation of function $g(q)$ which can be defined as follows:

(1) $g(q) = 1$ for $q < -1$ and $g(q) = 0$ for $q > 1$;

(2) For $q \in [-1, 1]$, $g(q)$ is a $r+1$ degree polynomial whose coefficients can be determined by solving the equation $\int_{-1}^{1} q^j g(q)dq = (-1)^j/(j+1)$, for $j = 0, \ldots, r-1$;

(3) If r is even, the function $g(q)$ is the same for r and $r+1$.

Rescaling the function $g(q)$ with a smoothing factor δ, $g((Q-q)/\delta)$ is the smoothing function used to approximate the indicator function $f(q)$ defined in equation (19). The smoothing factor δ plays an important role in MLMC estimation. The larger the δ is, the smoother the function, consequently the smaller the variance. So using a smoothing

function with a large δ, MLMC can be very computationally efficient, but in the same time the estimated CDF may deviate from the true CDF substantially with a large smoothing error. The reasonable choice of δ can be determined by limiting the smoothing error to a required accuracy as discussed below.

We denote the MLMC estimator using smoothing as $\tilde{F}_{M,t}^{ML}(q)$. It has three main sources of errors, the interpolation error of using F_t to approximate F, the smoothing error of using $\tilde{F}_t$ to approximate F_t and the MLMC error in estimation of $\tilde{F}_t$ with the MLMC estimator $\tilde{F}_{M,t}^{ML}$, where the MLMC error, like equation (10), can be further expanded into two parts, the sampling error and spatial discretization error. To sum up, the mean square error of $\tilde{F}_{M,t}^{ML}$ can be expressed as

$$\begin{aligned}\mathbb{E}\|F-\tilde{F}_{M,t}^{ML}\|_\infty^2 &\le \underbrace{\mathbb{E}\|F-F_t\|_\infty^2}_{e_1^{ML}} + \underbrace{\mathbb{E}\|F_t-\tilde{F}_t\|_\infty^2}_{e_2^{ML}} + \mathbb{E}\|\tilde{F}_t-\tilde{F}_{M,t}^{ML}\|_\infty^2 \\ &\le e_1^{ML}+e_2^{ML}+\underbrace{\|\log t\sum_{\ell=0}^{L}N_\ell^{-1}\mathbb{V}[Y_{\ell,t}^g]\|_\infty}_{e_3^{ML}} + \underbrace{\|\mathbb{E}[g_{M,t}^\delta-g_t^\delta]\|_\infty^2}_{e_4^{ML}},\end{aligned} \tag{22}$$

{mse_mlmc_cdf}

where $Y_{\ell,t}^g = g((Q_{M_\ell}-q)/\delta)-g((Q_{M_{\ell-1}}-q)/\delta)$, $g_{M,t}^\delta = g((Q_M-q)/\delta)$ and $g_t^\delta = g((Q-q)/\delta)$, and $q=\{u_1,u_2,\cdots,u_t\}$ at the t points between $[U_0,U_1]$ as defined before.

A sufficient condition to achieve a RMSE accuracy of ε is that all the four errors are less than $\varepsilon^2/4$. When cubic spline interpolation is used like above standard MC case, t can be chosen as

$$t \ge (U_1-U_0)/(\frac{12\varepsilon}{\|F^{(4)}(x)\|_\infty})^{1/4}+1 \tag{23}$$

{t_mlmc}

by limiting the interpolation error e_1^{ML} smaller than $\epsilon^2/4$.

The smoothing error e_2^{ML} depends on two parameters, r and δ, of the smoothing function. For a certain r, the δ value can be determined by setting $e_2^{ML}\le\varepsilon^2/4$, i.e.,

$$\|\mathbb{E}[f(q)]-\mathbb{E}[g((Q-q)/\delta)]\|_\infty \le \varepsilon/2. \tag{24}$$

{delta}

Practically, for each q, we solve the equation $\int_{-\infty}^{+\infty} p(Q)\,(f(q) - g((Q-q)/\delta))\,dQ = \varepsilon/2$ to get a δ value, and finally we take the smallest δ as the value which satisfies the smoothing error accuracy requirement and use it to determine the smoothing function for MLMC estimation.

As the t points and smoothing function are determined, we then use MLMC to estimate the expectation of the smoothing function at the t points and then extend the estimation to the entire interval of $[U_0, U_1]$ to get the estimated CDF. The estimation of $\tilde{F}_t$ using MLMC can be implemented according to Algorithm I, with the optimal N_ℓ chosen as $N_\ell \geq 4\varepsilon^{-2}\log t \left\| \sqrt{\mathbb{V}[Y^g_{\ell,t}]/M^\gamma_\ell} \left(\sum_{\ell=0}^{L} \sqrt{\mathbb{V}[Y^g_{\ell,t}]M^\gamma_\ell} \right) \right\|_\infty$ when $M_L < \bar{M}$ to achieve the sampling error accuracy of $\varepsilon^2/4$. When $M_L = \bar{M}$, e_4^{ML} is zero. In this case, the first three errors in equation (22) less than $\varepsilon^2/3$ can achieve the RMSE accuracy ε. For this accuracy setting, the $t, \delta, \text{and} N_\ell$ values can be determined accordingly.

Putting together the elements discussed above, the MLMC algorithm to estimate the CDF with smoothing function can be implemented in practice as follows:

Algorithm II:

1. Determine the estimated interval $[U_0, U_1]$ based on the approximated CDF with an initial set of 1000 samples;

2. Determine t based on equation (23) where the fourth derivative of F can be estimated with the initial samples;

3. Calculate δ based on equation (24) with a fixed r;

4. Implement Algorithm I to estimate $\tilde{F}_t$ at the t points;

5. Extend the estimation to the entire interval $[U_0, U_1]$ by interpolation to get the estimated CDF.

Note that in above algorithm, for a high accuracy requirement, the initial samples used in Steps 1 and 2 are better simulated at the highest level. When samples from some coarse level are used, the interval in Step 1 may be underestimated and the t value is Step 2 may be overestimated caused by a large estimated derivative of F. An relative accurate estimation of high-order derivatives of F can be calculated according to *Schuster* [1969].

The key point in Algorithm II is to calculate δ because δ critically determines the MLMC computational efficiency. *Giles et al.* [2014] calculated δ simply by $\delta = \varepsilon^{1/(r+1)}$ on the basis of asymptotic analysis. This calculation produces a very small δ value and thus results in a marginal time saving of MLMC over the standard MC. We calculate δ by setting the smoothing error equal to the required accuracy and solved the equation exactly. In this way, the calculated δ is guaranteed to have a largest value and thus results in a significant computational efficiency improvement of MLMC estimation.

5. Application to Oil Reservoir Simulation

{s:example}

To illustrate effectiveness and efficiency of the MLMC method, it is applied to a synthetic study of oil reservoir simulation. The synthetic case is designed based on the fine grid model from the tenth SPE comparative solution project [*Christie and Blunt*, 2001]. The tenth SPE fine grid problem is chosen for the following reasons: (1) the model has a sufficient fine grid making the use of the standard MC simulation on the fine grid solution very time costly, though not impossible; (2) full details of the problem and data files are available for free downloading from the project website; and (3) it is a typical oil reservoir problem and many researches have been worked on this problem [*Subbey et al.*, 2004; *Di Donato et al.*, 2003].

5.1. Model description

{s:model}

The fine grid geological model from the tenth SPE project is represented in a three-dimensional domain with $365.76 \times 670.56 \times 51.816$ cubic meters (m^3) and discretized into $60 \times 220 \times 85$ cells. There are four production wells at the four corners of the model, each producing at 4000 psi bottom hole pressure, and a central injection well with an injection rate of 5000 barrels/day. All wells are vertical and completed throughout the formation. The simulation of this sufficiently fine model is very time costly. With the efficient reactive flow and transport simulator, PFLOTRAN [*Mills et al.*, 2007], running in ten processors, it takes about five hours to get the simulation results of the oil production after 100 days. This amount of time is almost unaffordable for the MC simulation with thousands of simulation runs.

To demonstrate the MLMC method in a reasonable time and without lose of generality, our synthetic model has the following modifications: (1) the synthetic model is two dimensional but with the same spatial discretization as one layer of the fine grid tenth SPE model, i.e., total 60×220 cells in a 365.76×670.56 m^2 domain; (2) the permeability (k) field of the synthetic model is taken from the first layer of the fine grid tenth SPE model as shown in Figure 1 (a) and the porosity is treated as a constant with value of 0.1; (3) to make the 2-D synthetic model physically reasonable, the injection rate of the central injection well is revised as 62.90 barrels/day; and (4) the simulation time is reduced to 2 days since after about 2 days the oil production tends to be steady. With these simplifications, one simulation takes about 150 seconds with PFLOTRAN running on one processor, so a thousand simulation runs can be completely less than two days. This amount of computational time is affordable for testing and demonstrating our MLMC method for uncertainty quantification. The application of MLMC simulation on the 2-D problem can be extended to a 3-D problem easily and straightforwardly.

5.2. Application of the MLMC method

lication}

We are interested in investigating the influence of the permeability uncertainty on the mass flow rate Q of production well P whose location is shown in Figure 1 (a). Specifically, we want to estimate the expectation and the CDF of Q using the standard MC and the MLMC methods.

We assume that the random permeability field $log(\mathbf{k})$ is a Gaussian process whose mean and covariance structure can be inferred from the 36 synthetic data regularly situated in Figure 1 (b). Here we use the entire synthetic data set to infer the covariance structure in the aim of capturing the spatial correlation of the $log(\mathbf{k})$ field as accurate as possible. The accurate spatial correlation is important in this study because we want to verify our statement about the relation between the correlation length and the choice of the coarsest level in MLMC. Based on the sample variograms shown in Figure 2, we consider two variogram models, exponential (Exp) and spherical (Sph). Both models have two parameters, sill and correlation length. They are estimated by fitting the variogram models to the sample variograms through the least square estimation. As shown in Figure 2, the spherical model has better fits in both x- and y-directions than the exponential model. Then based on the fitted spherical model and conditioned on the 36 synthetic data, random realizations of permeability $log(\mathbf{k})$ field are generated using the sequential Gaussian simulation program (SGSIM) in GSLIB [*Deutsch and Journel*, 1998], where one realization is shown in Figure 1 (c). The generated realizations differ from the synthetic $log(\mathbf{k})$ field in Figure 1 (a) but at the 36 locations with data available they are exactly the same.

When using standard MC to estimate $\mathbb{E}[Q]$ and F, the simulations are conducted at the measurement scale with grid size $M = 60 \times 220$ cells. When using MLMC, we set

the highest level at the measurement scale and the lower levels have half number of grid cells of its preceding higher level in both x- and y-directions, i.e., $M_{\ell-1} = (1/2)^2 M_\ell$ for $\ell \geq 1$. The coarsest level M_0 is chosen as its cell length h_0 smaller than the correlation length λ. Based on the fitted spherical model in Figure 2, the correlation length in x- and y-directions are $\lambda_x = 217m$ and $\lambda_y = 114m$, respectively. So the coarsest level grid has 2 cells in x-direction with cell length $h_{0x} = 182.88m < \lambda_x$ and 7 cells in y-direction with cell length $h_{0y} = 95.79m < \lambda_y$. In this setting, total six levels (level $\ell = 0, 1, ...5$) are defined in Figure 3. For simulations on the levels $\ell < 5$ the generated permeability field is upscaled using the simple average method. More advanced upscaling methods as reviewed in *Wen and Gómez-Hernández* [1996] can be used and they are expected to improved the MLMC efficiency further, but this topic is out of the scope of this study and is deemed to pursue in future.

The coarsest level is chosen as $h_0 < \lambda$ because as the grid coarsened further when the cell length h_0 is greater than the correlation length λ, the variance $\mathbb{V}[Y_\ell]$ at certain ℓ will be greater than $\mathbb{V}[Q_M]$, making the inclusion of lower levels meaningless with respect to computational efficiency. For example, we consider two cases, Case I with $h_0 < \lambda$ and Case II with $h_0 > \lambda$. In Case I the coarsest level has grid 2×7 and it has total six levels as shown in Figure 3. In Case II the coarsest level has grid 1×4 which is one level coarser than Case I. So Case II has total seven levels with the six higher levels the same as Case I. The behavior of the variance $\mathbb{V}[Q_\ell]$ and $\mathbb{V}[Y_\ell]$ for all the six levels in Case I and the seven levels in Case II are plotted in Figure 4 (a) and (b), respectively.

Figure 4 (a) indicates that $\mathbb{V}[Q_\ell]$ is approximately constant on all levels, numerically verifying the assumption $\mathbb{V}[Q_0] \approx \mathbb{V}[Q_M]$ we made above equation (13). The variance $\mathbb{V}[Y_\ell]$ decreases as the level ℓ increases and the decay rate β is about 0.3,

which is estimated as minus half of the slope of line $log_2 V[Y_\ell]$ according to relation $\mathbb{V}[Y_\ell] = c_3 M_\ell^{-\beta} = c_3(2^{2\ell}M_0)^{-\beta}$. In comparison of $\mathbb{V}[Q_\ell]$ and $\mathbb{V}[Y_\ell]$, Figure 4 (a) indicates that the two values get closer as the level decreases, i.e., as the grid cell length increases. It seems that when the cell length increases further, the two lines will eventually cross and $\mathbb{V}[Y_\ell]$ will be larger than $\mathbb{V}[Q_\ell]$. This situation is observed in Figure 4 (b) of Case II with the grid coarsening one more level and the cell length $h_0 > \lambda$. In Figure 4 (b), at level $\ell = 1$, $\mathbb{V}[Y_\ell] > \mathbb{V}[Q_\ell] \approx \mathbb{V}[Q_M]$, suggesting that further coarsening from this level is helpless in improving computational efficiency. Also, when $h_0 > \lambda$, at level $\ell = 0$ the variance $\mathbb{V}[Q_\ell]$ ceases to be constant and $\mathbb{V}[Q_0]$ is actually greater than $\mathbb{V}[Q_\ell]$ for $\ell > 0$.

Note that not all the six levels defined in Figure 3 are needed for MLMC to achieve a desired accuracy. For a low RMSE accuracy requirement with large ε value, the needed highest level L for convergence in Algorithm I may be smaller than the defined highest level 5. More discussion about the relation between ε and L is presented in Section 6.1.

The γ value in calculation of N_ℓ is estimated based on relation $\mathcal{C}_{M_\ell} = c_2 M_\ell^\gamma = c_2(2^{2\ell}M_0)^\gamma$ where the cost $\mathcal{C}_{M_\ell}$ is measured using the computational time simulated at level M_ℓ. As shown in Figure 5, the estimated γ is about 0.9, half of the slope of the time increasing line. In this study the estimated $\gamma > \beta$, i.e., the computational cost $\mathcal{C}_{M_\ell}$ increases faster than the variance $\mathbb{V}[Y_\ell]$ decays, suggesting that the cost in the higher levels is dominant.

6. Results and Discussions

{s:results}

In this section, we compare the results of MLMC with those of standard MC in estimating the expectation of Q and the CDF of $log(Q)$. $log(Q)$ is used in estimation of CDF because the range of Q samples is very wide and a large portion of samples are close to zero. All the simulations are conducted with the computer code PFLOTRAN running on

one processor. Computational efficiency of MLMC is evaluated from two perspectives: (1) the computational time required to obtain an estimate of $\mathbb{E}[Q]$ and F within a prescribed accuracy, and (2) the accuracy of the estimated $\mathbb{E}[Q]$ and F for a given amount of time. These two criteria are complementary in that the first criterion is for the situation when a specific accuracy is required and a large amount of computational time is affordable while the second criterion when limited amount of time is affordable and the accuracy is not a serious concern.

6.1. Estimation of the expectation

{s:ex1}

We first illustrate the algorithm using MLMC to estimate $\mathbb{E}[Q]$. Figure 6 depicts the number of levels and the number of samples N_ℓ at each level required for the MLMC expectation estimator to achieve RMSE accuracy $\varepsilon = 0.3, 0.2, 0.1,$ and 0.05. The figure indicates that for higher accuracy requirement with smaller ε values, more levels are needed and each level needs a larger number of samples for convergence. For example, when $\varepsilon = 0.3$, only three levels are sufficient and the number of samples at the highest level $L = 2$ is just 220; when $\varepsilon = 0.05$, all the six levels are needed and the number of samples at the highest level $L = 5$ increases significantly to 1832. In addition, we note that for a certain ε the N_ℓ decreases as the level ℓ increases and the decrease rate is about a constant independent of the accuracy ε. Equation (16) indicates that the decrease rate of N_ℓ relies on the variance decay rate β and the cost increase rate γ as $N_\ell \propto \sqrt{\mathbb{V}[Y_\ell]/M_\ell^\gamma} \propto M_\ell^{(-\beta-\gamma)/2}$. Larger β and γ values cause faster decay of N_ℓ resulting in larger computational efficiency. This conclusion is consistent with the discussion in Section 3.2.2.

We now evaluate the computational efficiency of the MLMC estimator from the two perspectives. Figure 7 plots the RMSE and the computational time for the standard MC

and the MLMC estimators. We compare (1) the computational time of the two estimators with the RMSE fixed at 0.2, 0.1, and 0.05, and (2) their RMSEs with the time fixed at 25, 50, and 100 hours. The comparison results are summarized in Table 1. Table 1 indicates that to achieve the same RMSE, MLMC needs significantly less computational time, and the time savings on average are about 90%. For example, for RMSE=0.1, in comparison with the 177.49 hours of standard MC, the time for MLMC is just 15.64 hours with cost savings more than 90%. On the other hand, for the same computational time, MLMC can achieve higher accuracy with smaller RMSEs than standard MC, and the accuracy improvement on average is more than 60%. For example, within 100 hours, the RMSE of the standard MC estimator is about 0.158, but for MLMC the RMSE reduces to 0.061. The two comparison suggests that MLMC is more efficient in estimating the expectation than standard MC.

We explore reasons of high computational efficiency in the MLMC estimator from two aspects, the computational cost in each level and the increase of estimation accuracy with inclusion of more levels. Take the results for $\varepsilon = 0.05$ as an example, Figure 8 (a) plots the time for one sample run and the required number of samples at all the six levels. The figure shows that as the level increases and the grid resolution gets finer, the required time for one sample run increases (the red solid line), but in the meantime the required number of samples decreases (the dashed blue line). This suggests that MLMC puts the right effort where it is needed, i.e., most of model executions conducted on the computationally frugal coarse grids (lower levels) and just a few on the computationally expensive fine grids (higher levels). In this way, the total computational time is significantly saved compared to standard MC which spends all its effort on the computationally most expensive finest grid. Though with less computational time, MLMC can achieve the similar estimation

as standard MC because it considers the estimates on a sequence of grids so that the less accurate estimates on the preceding coarser grids can be sequentially corrected by estimates on the following finer grids. Figure 8 (b) illustrates how the MLMC estimate gets closer to the standard MC estimate as more levels included.

6.2. Estimation of the CDF

{s:ex2}

We first illustrate Algorithm II using MLMC to estimate the CDF of $log(Q)$. To assess the accuracy of our Algorithm II, the actually calculated RMSE of the MLMC estimator is compared with the desired accuracy ε. Here we take the standard MC results with 50000 samples simulated on measurement scale as a reference, and evaluate the MLMC estimator at five desired accuracy, $\varepsilon = 0.05, 0.04, 0.03, 0.02,$ and 0.01. Figure 9 shows the estimated CDF of $log(Q)$ based on 1000 samples simulated on the six levels. The figure indicates that the estimated CDF from the coarsest level samples Q_0 deviates from the reference the most, suggesting that the determination of estimation interval and t values using the samples of the coarsest level may not be accurate. The estimated CDF based on the 1000 samples Q_5 of the highest level is very close to the reference, indicating that the suggested samples in Steps 1 and 2 of Algorithm II are reasonable in this study. We then, based on the 1000 samples of Q_5, choose the estimation interval as $[-3, 5]$ since the CDF values beyond this interval are almost zero, and determine the t values for different ε requirement.

The calculation of δ requires r in the smoothing function to be determined first. Since r is typically unknown in practice and currently there is no theory about the optimal choice of r, here we choose a range of r, $r = 3, 5, 7, 9,$ and 11, to illustrate its impact on δ. Figure 10 (a) plots δ values for the range of r at the five ε accuracy requirement. The figure shows that for a large r value, i.e., constructing the smoothing function with a high

degree polynomial, the δ value is large, and this is true for all ε. Though different δ values are resulted, the calculated smoothing errors for the range of r are close to each other and all meet the accuracy requirement, as shown in Figure 10 (b). The calculated smoothing errors are not exactly equal to the error bound due to the numerical error in calculation of δ based on equation (24).

A large δ produced by a large r results in a relatively small variance of $Y^g_{\ell,t}$. Figure 11 (a) shows $\mathbb{V}[Y^g_\ell] = \left\|\mathbb{V}[Y^g_{\ell,t}]\right\|_\infty$ for $r = 3$ (the solid red line), and the relative variance reductions of $r = 5, 7, 9, 11$ compared to $r = 3$ (the dashed blue lines). The figure indicates that $\mathbb{V}[Y^g_{\ell,t}]$ decays with levels and for larger r values, the decay is faster and the variance at each level is smaller. The smaller variance and faster decay suggest that a smaller number of samples are needed at each level and with the level increases, the required number of samples decreases faster. Thus, when using the smoothing function of $r = 11$, MLMC needs less time to achieve the same ε accuracy compared to the case when $r = 3$, as shown in Figure 11 (b). Since the parameters used in Algorithm II are determined by limiting the errors smaller than their corresponding required accuracy bounds, the actual RMSE of the MLMC estimator is smaller than the ε demand. Figure 11 (c) plots the ratio of calculated RMSE and the desired accuracy ε for the range of r. The figure indicates that the ratio is below one regardless of r.

To sum up, the above analysis suggests the following: (1) Algorithm II can be reasonably and efficiently used for CDF estimation. Especially, the way we proposed to calculate δ significantly improves the computational efficiency. For example, *Giles et al.* [2014] calculated δ as $\delta = \varepsilon^{1/(r+1)}$, which gives $\delta = 0.68$ for $\varepsilon = 0.01$ and $r = 11$. This value is much smaller than our δ of 3.89 for the same situation as shown in Figure 10 (a). As a result, for their small δ value, it requires 553 hours to achieve $\varepsilon = 0.01$, whereas our larger

δ only requires 163 hours for the same accuracy demand (Figure 11 (b)), resulting in more than three times computational efficiency improvement. (2) For the same ε, MLMC requires less time for a large r.

However, no matter what r value is, the MLMC estimator with smoothing is more efficient than the standard MC. Take $r = 3$ for example, Figure 12 shows that the time used by standard MC is significantly more than that used by MLMC to achieve a certain accuracy ε. For $r = 11$, the computational efficiency improvement of MLMC is more outstanding. As shown in Table 2, with the RMSE fixed at 0.03, 0.02, and 0.01, the time saving of MLMC with $r = 3$ over standard MC is on average about 86.1% and for $r = 11$ the averaged time saving is about 88.3%. This is also observed when we compared the RMSE with time fixed. For example, with time of 25, 100, and 150 hours, Table 2 shows that MLMC with $r = 3$ can improve the accuracy on average about 43.7% and for $r = 11$ the averaged accuracy improvement is about 48%.

Thus, due to the computational efficiency, for a limited time the MLMC estimator can capture the main features of the true CDF, whereas the standard MC estimator may deviate from the true CDF a lot. For example, as shown in Figure 13 (a), with only 0.2 hours, the estimated CDF of MLMC with $r = 3$ is close to the reference with a small RMSE of 0.05, but with the same time the estimated CDF by standard MC differs from the reference dramatically with RMSE about 0.45. The reason is that, this small amount of time can only simulate five samples in standard MC case, but it can generate 968, 501, and 176 samples for the first three levels, respectively in MLMC and this number of samples is enough to capture the main features of the underlying CDF. As shown in Figure 9, 1000 samples of Q_1 from level two can mimic the shape of the reference CDF very well. On the other hand, with a large amount of time affordable, both standard MC and

MLMC can make a equally good estimation, but MLMC estimator needs a significantly fewer time. As shown in Figure 13 (b), the estimated CDFs of standard MC and MLMC are equally well, both are close enough to the reference. But to attain the same accuracy, MLMC estimator needs only 195.45 hours while standard MC needs 560.83 hours.

7. Summaries and Conclusions

{s:conclusion}

This paper presents a multilevel Monte Carlo method to improve computational efficiency of uncertainty quantification. MLMC is a variance reduction technique for the standard MC. It improves computational efficiency by conducting simulations on a geometric sequence of grids, a larger number of simulations on coarse grids and fewer simulations on fine grids. Thus, this study tackles a challenging problem of the prohibitively large computational cost in uncertainty quantification for large-scale applications with high dimensional parameter spaces.

The MLMC method was first applied in a reservoir model to estimate the expectation of mass flow rate in a production well. In the estimation of expectation, the MLMC estimator can achieve the same accuracy as the standard MC, but requires dramatically less computational time with time savings on average about 90%. For the same computational time, MLMC can obtain higher accuracy than the standard MC with accuracy improvement on average more than 60%.

We then proposed an algorithm using MLMC to estimate the distribution function. The numerical results in the reservoir simulation show that our algorithm is efficient and reliable in achieving the desired accuracy. With this algorithm, to achieve the same accuracy as the standard MC, MLMC can obtain time savings about 88%. For the same time requirement, MLMC can improve accuracy of CDF estimation about 48%. And

within a very few computational time, MLMC estimator can capture the main features of the distribution.

The MLMC method is model independent and can be applied in environmental modeling and many other fields. The underlying multilevel concept is flexible to be used together with other sampling techniques such as stratified sampling and quasi Monte Carlo methods to further improve computational efficiency. To construct the multiple levels, it can use a geometric sequence of grids spatially or a geometric sequence of different timesteps temporally. Moreover, there was not special restriction in discretizing the grids and timesteps. Provided a suitable hierarchy of levels can be constructed, the MLMC estimator is expected to perform equally well on nonuniform and anisotropic problems as the uniform and isotropic ones.

The key for MLMC to improve computational efficiency is that the variance $\mathbb{V}[Y_\ell]$ has a meaningful decay compared to $\mathbb{V}[Q_M]$. For a highly heterogeneous problem with very small correlation length, the use of MLMC may barely obtain computational savings. For these problems, using a smoother approximation of the random field may be a solution, e.g., using the Karhunen-Loève (KL) expansion to generate random field realizations and truncating the KL-expansion to a few terms.

8. Acknowledgments

D. Lu, C. Webster, and C. Barbier were supported by the Director's Strategic Hire Funds through the Laboratory Directed Research and Development (LDRD) Program of Oak Ridge National Laboratory (ORNL). G. Zhang was supported by the Advanced Simulation Computing Research (ASCR), Department of Energy, through the Householder Fellowship at ORNL. The ORNL is operated by UT-Battelle, LLC, for the United States Department of Energy under Contract DE-AC05-00OR22725.

References

Heinrich, S. (2001), Multilevel Monte Carlo methods, *Lecture Notes in Computer Science*, *2179*, 3624-3651, Springer, Philadelphia, PA.

Deutsch C. V., and A. G. Journel (1998), GSLIB: Geostatistical software library and user's guide, 2nd ed., Oxford Univ. Press, New York.

Mill, R. T., G. E. Hammond, P. C. Lichtner, V. Sripathi, G. Mahinthakumar, and B. F. Smith (2007), Modeling subsurface reactive flows using leadership-class computing, *Journal of Physics: Conference Series*, *180*, doi: 10.1088/1742-6596/180/1/012062.

Christie, M. A., and M. J. Blunt (2001), Tenth SPE comparative solution project: A comparison of upscaling techniques, *SPEREE*, *4*(4), 308-317.

Di Donato, G., W. Huang, and M. Blunt (2003), Streamline-based dual porosity simulation of fractured reservoirs, *SPE 84036*.

Anderson E. C. (1999), Monte Carlo methods and importance sampling, *Lecture Notes for Stat 578C, Statistical Genetics*.

Barth, A., C. Schwab, and N. Zollinger (2011), Multilevel Monte Carlo finite element method for elliptic PDEs with stochastic coefficients, *Numer. Math.*, *119*: 123-161.

Brandt, A., and V. Ilyin (2003), Multilevel Monte Carlo methods for studying large scale phenomena in fluids, *J. Mol. Liq.*, *105*(2-3), 245-248.

Cliffe, K. A., M. B. Giles, R. Scheichl, and A. L. Teckentrup (2011), Multilevel Monte Carlo methods and applications to elliptic PDEs with random coefficients, *Comput. Vis. Sci.*, *14*: 3-15.

Franssen, H. J. H, A. Alcolea, M. Riva, M. Bakr, N. van der Wiel, and F. Stauffer(2009), A comparison of seven methods for the inverse modeling of groundwater flow: Application to the characterization of well catchments, *Adv. Water Resour.*, *32*(6), 851-72.

Giles, M. (2008), Multilevel Monte Carlo path simulation, *Operations Research*, *56*(3), 607-617.

Subbey, S., M. Christie, and M. Sambridge (2004), Prediction under uncertainty in reservoir modeling, *Journal of Petroleum Science and Engineering*, *44*, 143-153.

Laloy, E., B. Rogiers, J. A. Vrugt, D. Mallants, and D. Jacques (2013), Efficient posterior exploration of a high-dimensional groundwater model from two-stage Markov Chain Monte Carlo simulation and polynomial chaos expansion, *Water Resour. Res.*, *49*, 2664-2682.

Dereich, S., and F. Heidenreich (2011), A multilevel Monte Carlo algorithm for Lévy-driven stochastic differential equations, *Stochastic Process. Appl.*, *121*(7), 1565-1587.

Kloeden, P., A. Neuenkirch, and R. Pavani (2011), Multilevel Monte Carlo for stochastic differential equations with additive fractional noise, *Annals of Operations Research*, *189*(1), 255-276.

Abdulle A., A. Blumenthal (2013), Stabilized multilevel Monte Carlo method for stiff stochastic differential equations, *Journal of Computational Physics*, *251*, 445-460.

Giles, M., and C. Reisinger (2012), Stochastic finite difference and multilevel Monte Carlo for a class of SPEDs in finance, *SIFIN*, *1*(3), 575-592.

Charrier J., R. Scheichl, and A. L. Techentrup (2013), Finite element error analysis of elliptic PDEs with random coefficients and its application to multilevel Monte Carlo methods, *SIAM J. Numer. Anal.*, *51*(1), 322-352.

Teckentrup A. L., R. Scheichl, M. B. Giles, and E. Ullmann (2013), Further analysis of multilevel Monte Carlo methods for elliptic PDEs with random coefficients, *Numer. Math.*, *125*, 569-600.

Ghanem, R. (1998), Scales of fluctuation and the propagation of uncertainty in random porous media, *Water Resour. Res.*, *34*, 2123-2136.

Schuster E. F. (1969), Estimation of a probability density function and its derivatives, *The Annals of Mathematical Statistics*, *40*(4), 1187-1195.

Giles M., T. Nagapetyan, and K. Ritter (2014), Multilevel Monte Carlo approximation of distribution functions and densities, Preprint.

Zhang, D. (2002), Stochastic method for flow in porous media: coping with uncertainties, *Academic Press*, San Diego.

Li, H., and D. Zhang (2007), Probabilistic collocation method for flow in porous media: Comparisons with other stochastic methods, *Water Resour. Res.*, *43*(9), W09409, doi: 10.1029/2006WR005673.

Lin, G., and A. Tartakovsky (2009), An efficient, high-order probabilistic collocation method on sparse grids for three-dimensional flow and solute transport in randomly heterogeneous porous media, *Advances in Water Resources*, *32*(5), 712–722, doi: 10.1016/j.advwatres.2008.09.003.

Guadagnini, A, and S. P. Neuman (1999), Nonlocal and localized analyses of conditional mean steady state flow in bounded, randomly nonuniform domains 1. Theory and computational approach, *Water Resour. Res.*, *35*(10), 2999-3018.

Guadagnini, A, and S. P. Neuman (1999), Nonlocal and localized analyses of conditional mean steady state flow in bounded, randomly nonuniform domains 2. Computational examples, *Water Resour. Res.*, *35*(10), 3019-39.

Wen, X, and J. Gómez-Hernández (1996), Upscaling hydraulic conductivities in heterogeneous media: An overview, *Journal of Hydrology*, *183*.

Zhang, G., D. Lu, M. Ye, M. Gunzburger, and C. Webster (2013), An adaptive sparse-grid high-order stochastic collocation method for Bayesian inference in groundwater reactive transport modeling, *Water Resour. Res.*, *49*, 6871-6892, doi:10.1002/wrcr.20467.

Table 1. Comparison of computational efficiency between standard MC and MLMC in estimating expectation. {table1}

Compare computational time for a fixed RMSE			
RMSE	0.2	0.1	0.05
Time (h) (MC)	44.38	177.49	709.92
Time (h) (MLMC)	1.35	15.64	124.34
Time saving of MLMC (%)	96.96%	91.19%	82.49%
Compare RMSE for a fixed time			
Time (h)	25	50	100
RMSE (MC)	0.279	0.196	0.158
RMSE (MLMC)	0.096	0.084	0.061
Accuracy improvement of MLMC (%)	65.67%	57.00%	61.32%

Time saving of MLMC is calculated as $(Time_{MC} - Time_{MLMC})/Time_{MC} \times 100\%$.

Accuracy improvement of MLMC is calculated as $(RMSE_{MC} - RMSE_{MLMC})/RMSE_{MC} \times 100\%$.

Table 2. Comparison of computational efficiency between standard MC and MLMC in estimating CDF.

{table2}

Compare computational time for a fixed RMSE			
RMSE	0.03	0.02	0.01
Time (h) (MC)	49.66	129.29	560.83
Time (h) (MLMC (r=3))	1.62	4.61	195.45
Time (h) (MLMC (r=11))	1.45	3.97	163.23
Time saving of MLMC (r=3) (%)	96.75%	96.43%	65.15%
Time saving of MLMC (r=11) (%)	97.07%	96.93%	70.89%
Compare RMSE for a fixed time			
Time (h)	25	100	150
RMSE (MC)	0.0449	0.0237	0.0195
RMSE (MLMC (r=3))	0.0189	0.0150	0.0124
RMSE (MLMC (r=11))	0.0187	0.0140	0.0108
Accuracy improvement of MLMC (r=3) (%)	57.83%	36.64%	36.57%
Accuracy improvement of MLMC (r=11) (%)	58.40%	41.00%	44.52%

Time saving of MLMC is calculated as $(Time_{MC} - Time_{MLMC})/Time_{MC} \times 100\%$. Accuracy improvement of MLMC is calculated as $(RMSE_{MC} - RMSE_{MLMC})/RMSE_{MC} \times 100\%$. MLMC (r=3) means when using MLMC to estimate CDF, the smoothing function is constructed with r equal to 3.

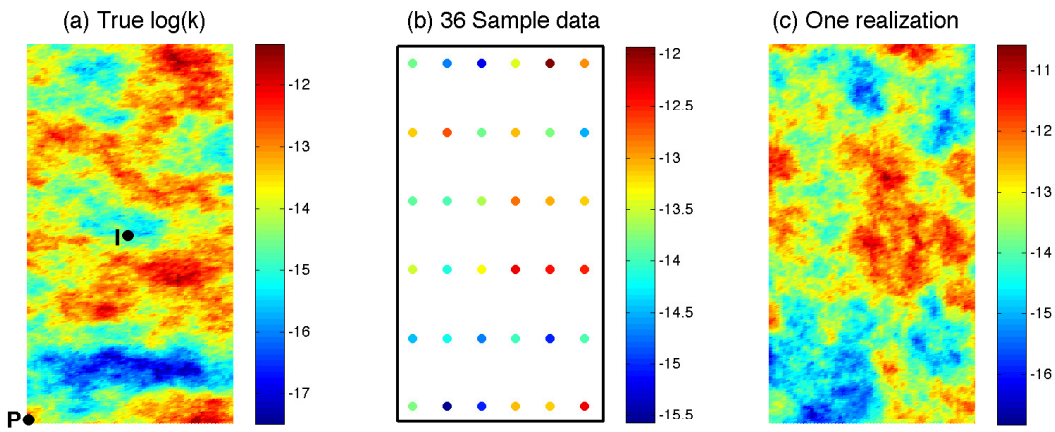

Figure 1. (a) True log permeability field log(k); I represents the injection well and P is the production well. (b) Locations of 36 sample data drawn from (a); conditioning on these samples, the realizations of random log(k) field are generated. (c) An example of one realization of random log(k) field. {TrueK}

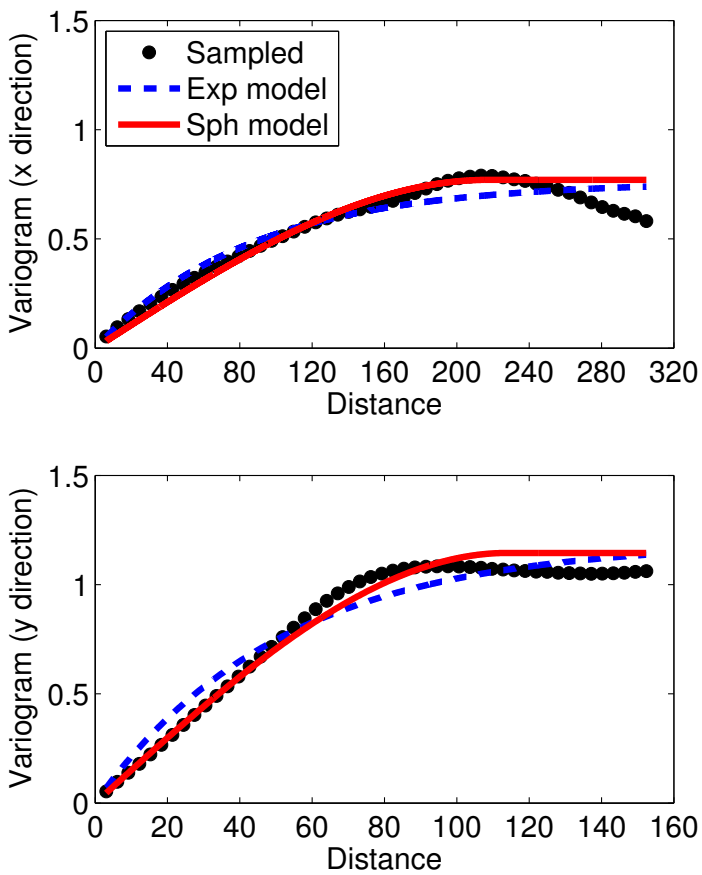

Figure 2. Sampled and fitted variograms with exponential (Exp) and spherical (Sph) models

{Gamfit} in x- and y-directions based on all synthetic permeability log(k) data.

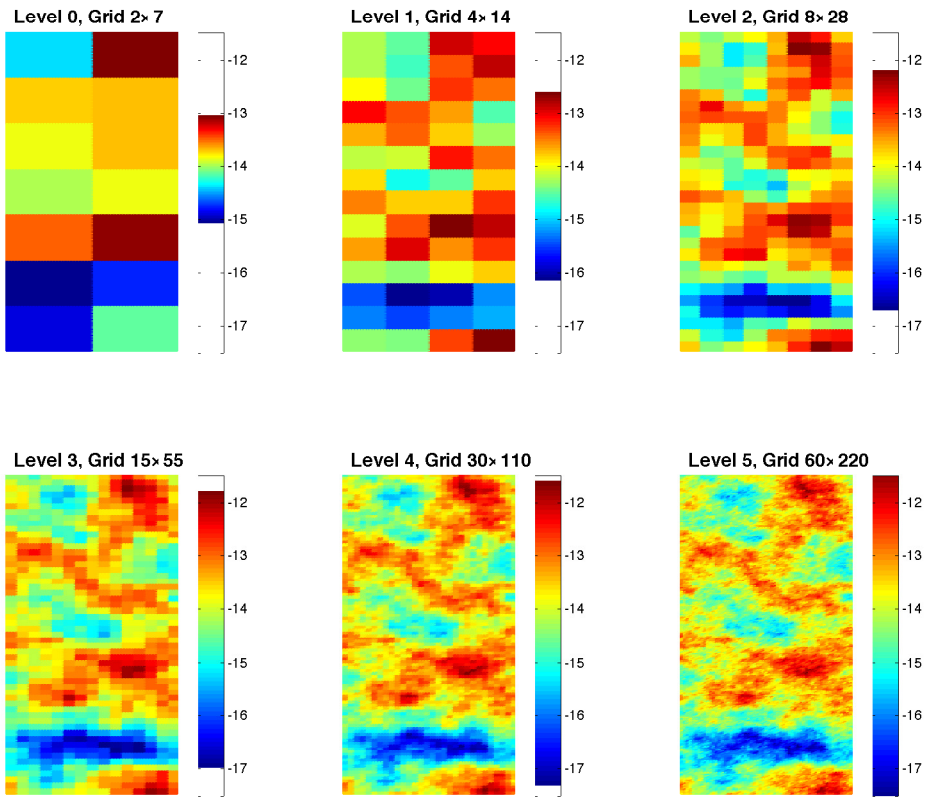

Figure 3. Grid size of different level grids used in MLMC. The highest level grid (level 5) has the same size at the measurement scale. {Levels}

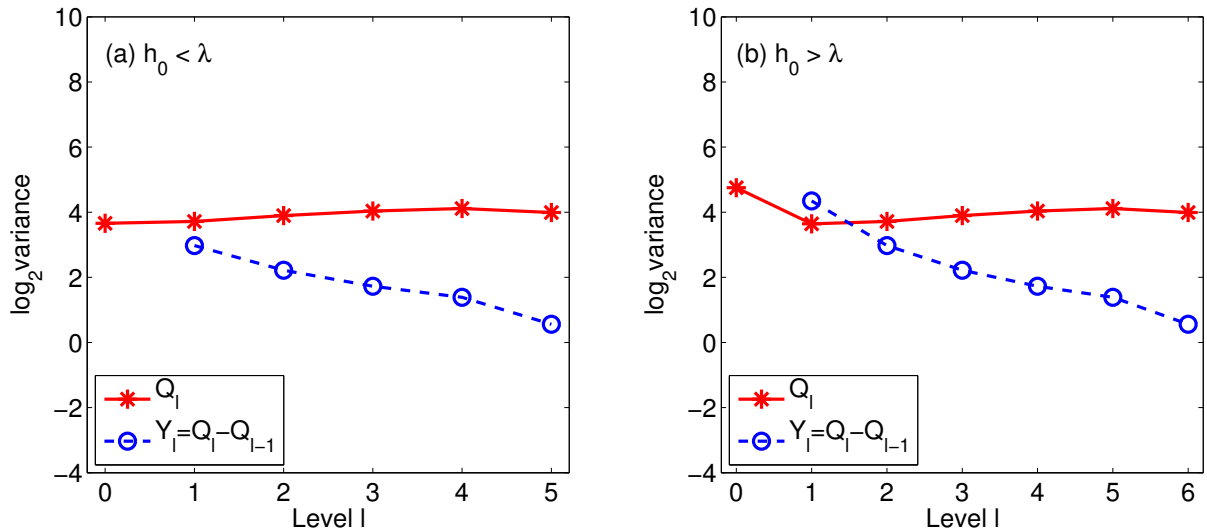

Figure 4. Plots of $\mathbb{V}[Q_\ell]$ and $\mathbb{V}[Y_\ell] = \mathbb{V}[Q_\ell - Q_{\ell-1}]$ for each level ℓ when (a) $h_0 < \lambda$ and (b) $h_0 > \lambda$ where h_0 is the cell length of the coarsest level grid and λ is the correlation length of the permeability field.

ar_decay}

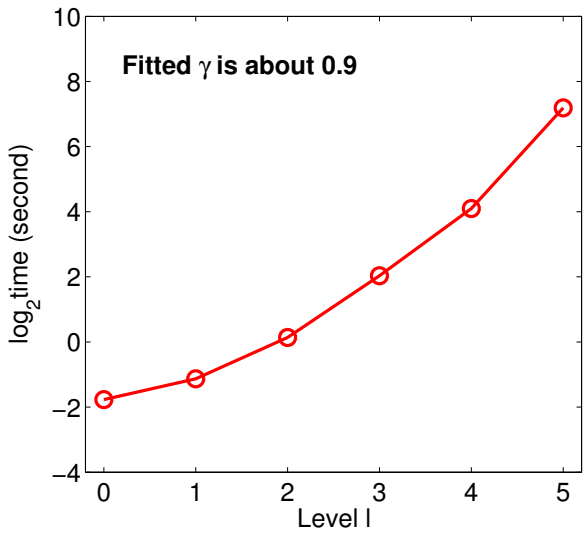

Figure 5. Relation between the time used to compute one sample on level ℓ and the level ℓ. The slope is roughly 1.8, indicating the increase rate of cost $\gamma \approx 0.9$. {Gamma}

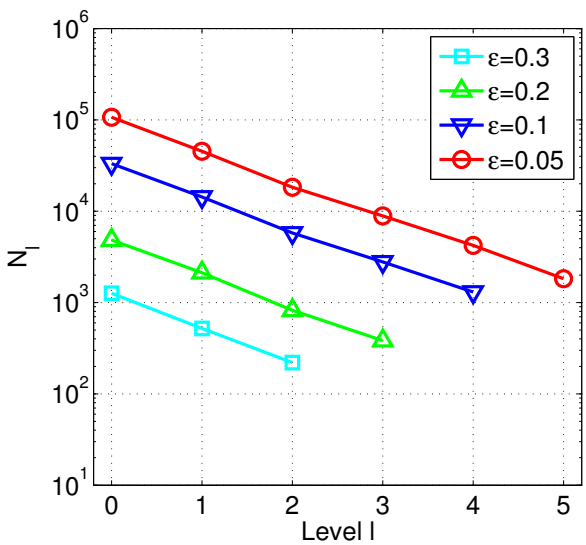

Figure 6. For a certain RMSE accuracy ε, the number of levels needed for convergence and the number of samples N_ℓ used in each level in $\mathbb{E}[Q]$ estimation with MLMC.

{NL}

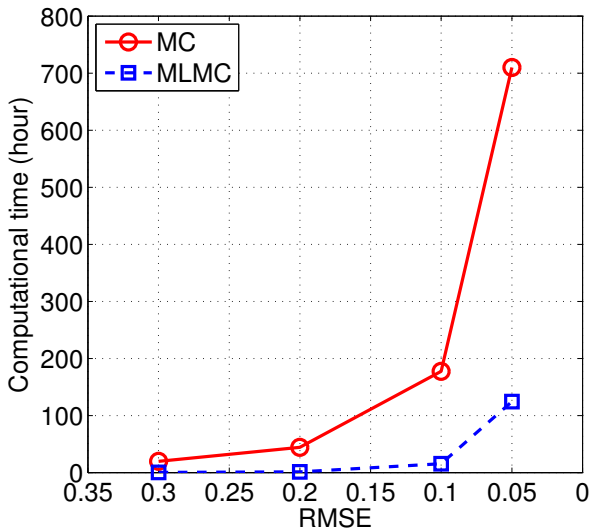

Figure 7. Achieved RMSE and used computational time by standard MC and MLMC in estimation of $\mathbb{E}[Q]$. Table 1 lists the comparison of computational time between the two estimators for fixed RMSE, and the comparison of RMSE for fixed time. {Expectati

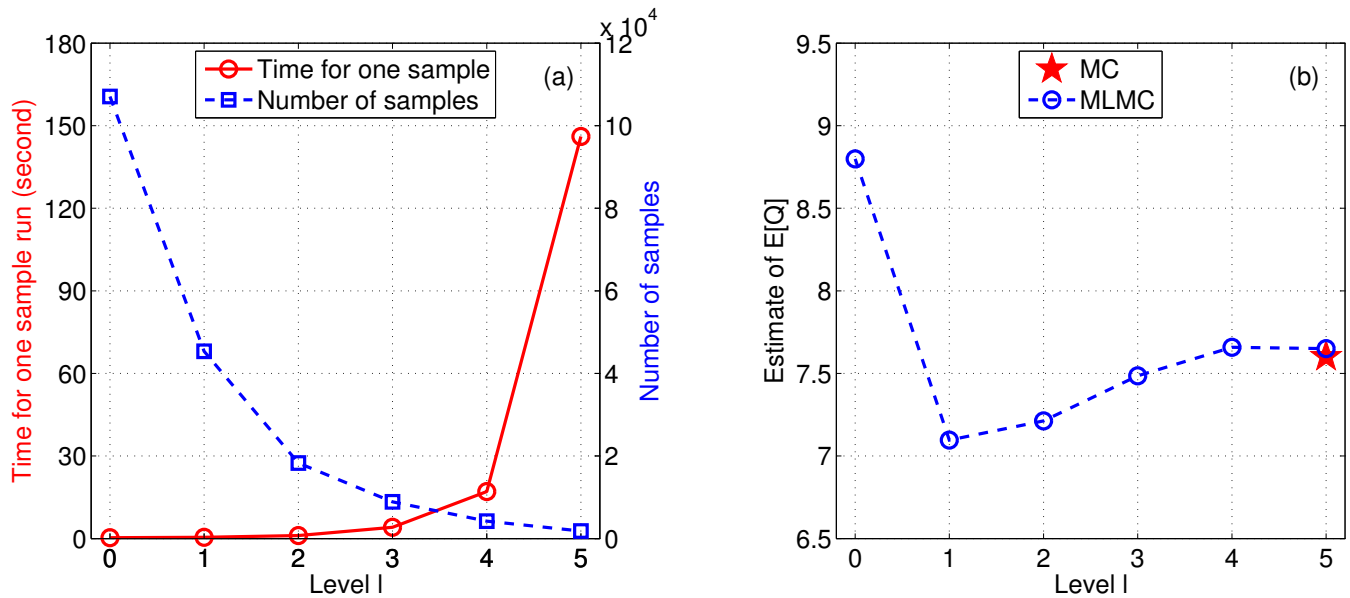

Figure 8. (a) Time for one sample run (red solid line corresponds to left y-axis) and the required number of samples (blue dashed line corresponds to right y-axis) at the six levels of MLMC; (b) MLMC estimates of $\mathbb{E}[Q]$ with considering different number of levels, and the standard MC estimate of $\mathbb{E}[Q]$.

{Time}

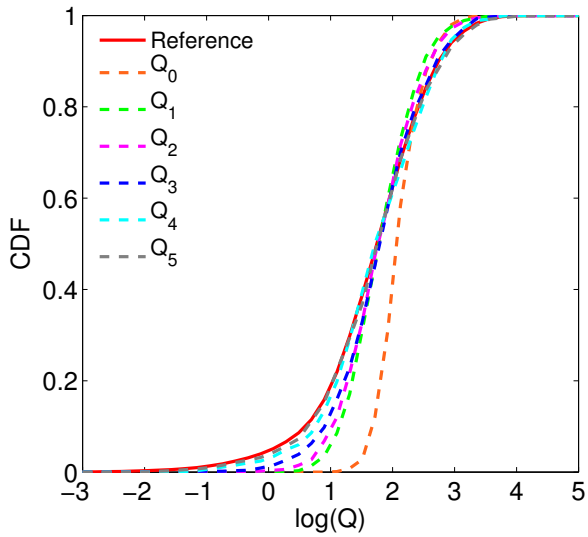

Figure 9. Estimated CDF $F_{log(Q)}$ based on 1000 samples simulated on the six levels. The reference CDF estimate is based on 50000 samples simulated on the highest level grid. {CDF0}

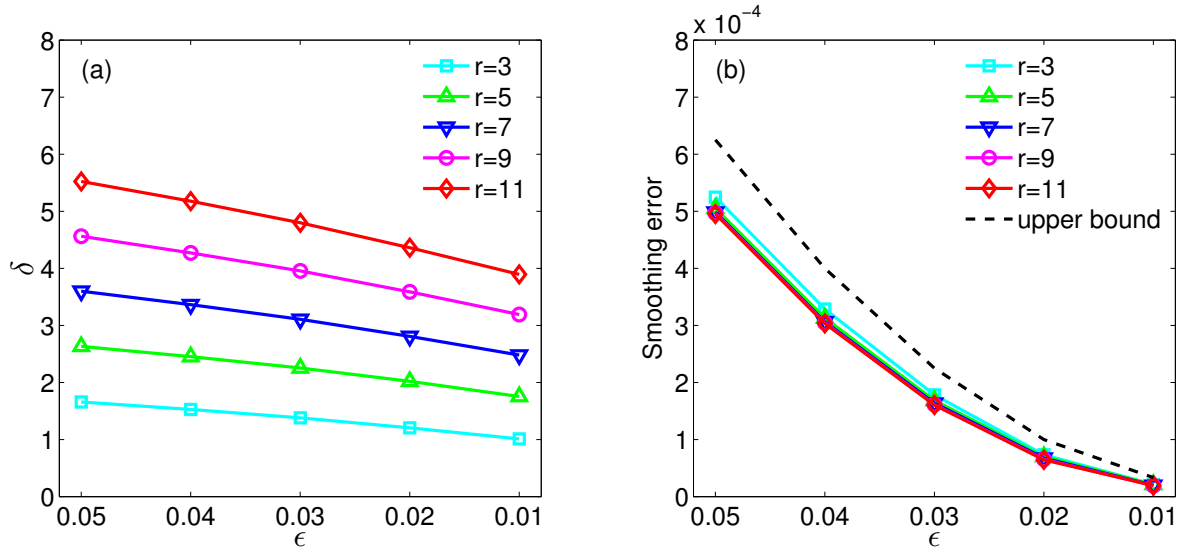

Figure 10. (a) Calculated δ for a range of r at five desired accuracy ε. (b) The smoothing errors of smoothing functions constructed with different r; the dashed black line is the accuracy requirement of smoothing error, i.e., error upper bound.

{Delta}

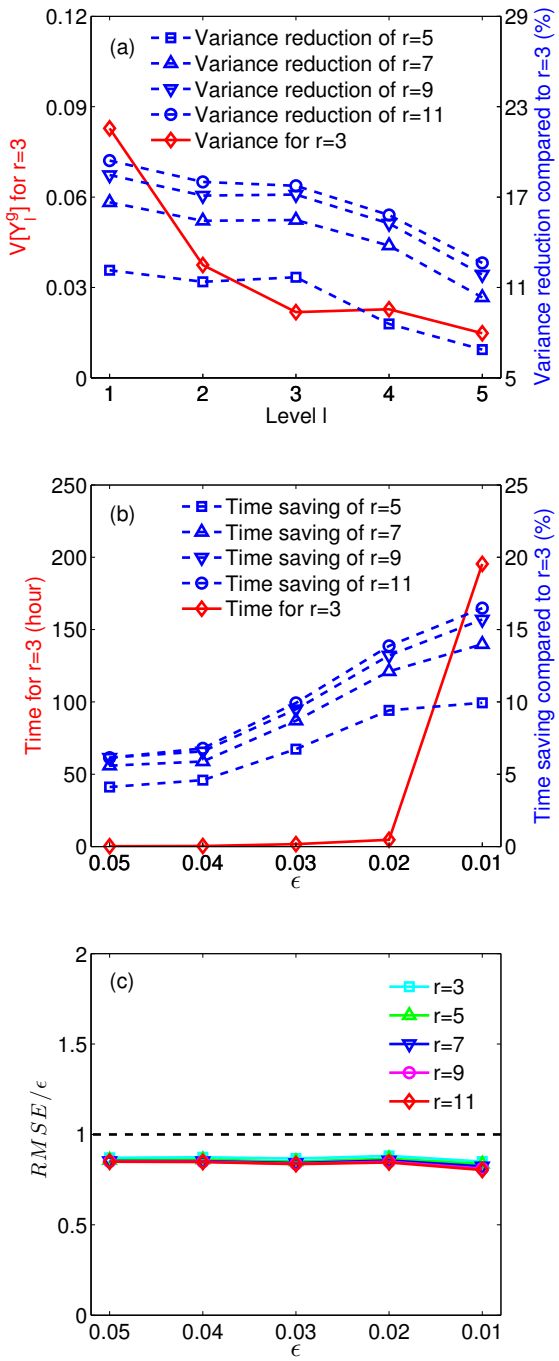

Figure 11. (a) Variance $\mathbb{V}[Y_\ell^g] = \left\|\mathbb{V}[Y_{\ell,t}^g]\right\|_\infty$ decay with levels for $r = 3$ (red solid line corresponds to left y-axis) and the relative variance reduction (%) of $r = 5, 7, 9, 11$ compared to $r = 3$ (blue dashed lines correspond to right y-axis). (b) Compuational time used for MLMC with $r = 3$ to achieve desired accuracy ε and the relative time savings (%) of $r = 5, 7, 9, 11$ compared to $r = 3$. (c) The ratio of calculated RMSE and desired accuracy ε for a range of r. {Var_CDF}

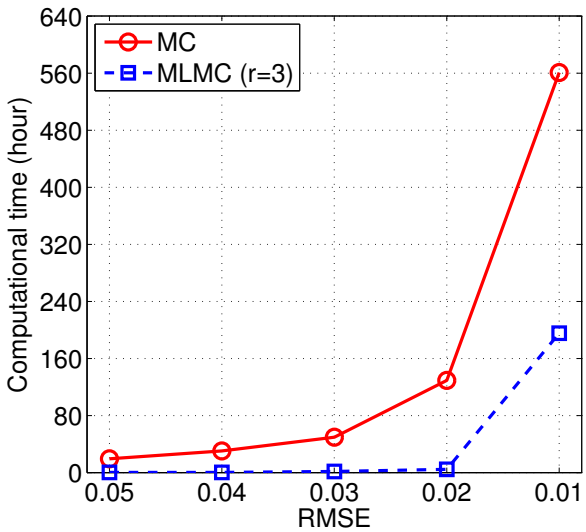

Figure 12. Achieved RMSE and used computational time of standard MC and MLMC for CDF estimation. Table 2 lists the comparison of computational time between the two estimators for fixed RMSE, and the comparison of RMSE for fixed time.

CDF_RMSE}

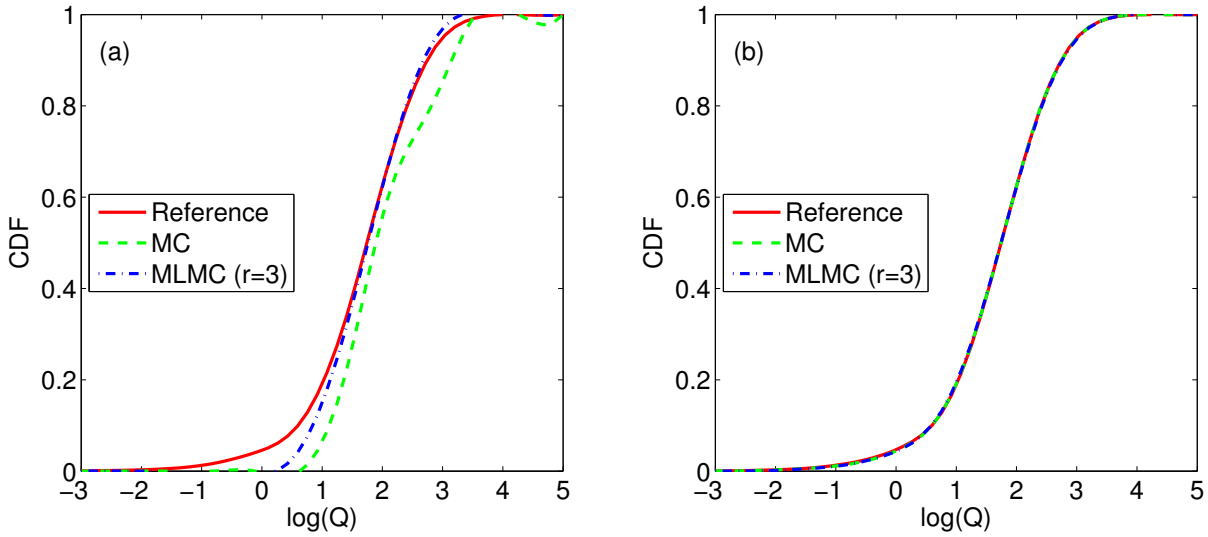

Figure 13. Comparison of estimated CDFs between the standard MC and MLMC when (a) computational time is 0.2 hours and (b) the accuracy requirement δ is 0.01. In (a), for the same time, the RMSE of standard MC is 0.45 but for MLMC is only 0.05; in (b), for the same desired accuracy, the time of standard MC is 560.83 hours but for MLMC is only 195.45 hours. {CDF}