AN IMPROVED MULTILEVEL MONTE CARLO METHOD WITH APPLICATION TO SOLVING PARTIAL DIFFERENTIAL EQUATIONS WITH RANDOM COEFFICIENS

JINGRUN CHEN, RUI DU, XUETING GUO, LING LIN, ZHIWEN ZHANG, AND XIANG ZHOU

Abstract. In this paper, we develop a novel multilevel Monte Carlo (MLMC) method to reduce the well-known computational cost in the MLMC method with a better convergence rate. Our main innovative idea is to optimize the initial mesh size h_0 with the total number of levels L fixed, in contrast to tuning the integer parameter L in the classical MLMC method. The refined result from our method outperforms the scaling of the total computational cost in the classical MLMC construction and further quantitatively demonstrates the powerful improvement. To show the efficiency of the proposed method, we apply our multilevel construction of mesh sizes and sample sizes to elliptic partial differential equations with random coefficients. By testing exponential and Gaussian covariance kernels in both one dimension and two dimensions, we observe that the improved MLMC method can save several times or even an order of magnitude of computation cost than the existing MLMC methods at certain regimes.

1. Introduction

Many physical and engineering applications involving uncertainty quantification (UQ) can be described by stochastic partial differential equations (SPDEs, i.e., PDEs driven by Brownian motion) or partial differential equations with random coefficients (RPDEs). In recent

---

Date: June 6, 2022.

2010 *Mathematics Subject Classification.* 35R60, 65C05, 65N30, 65N12, 65Y20.

Key words and phrases. Uncertainty quantification, multilevel Monte Carlo method, elliptic partial differential equations, random coefficients, computational complexity.

years, there has been an increased interest in the simulation of systems with uncertainties, and many numerical methods have been developed in the literature to solve SPDEs and RPDEs; see e.g. [17, 18, 19, 20, 21, 25] and reference therein. These methods can be effective when the dimension of stochastic input variables is low or moderate. However, their performance deteriorates when the dimension of stochastic input variables is high because of the curse of dimensionality.

There are some attempts in developing problem-dependent or data-driven basis functions to attack these challenging problems. Most of them take advantage of the fact that even though the stochastic input has high dimension, the solution actually lives in a relatively low dimensional space. Therefore, one can develop certain efficient numberical methods to solve SPDEs and RPDEs. In [22], Hou et al. explored the Karhunen-Loève expansion of the stochastic solution, and constructed problem-dependent stochastic basis functions to solve these SPDEs and RPDEs. In [23], the compressive sensing technique is employed to identify a sparse representation of the solution in the stochastic direction. In [24], Schwab et al. studied the sparse tensor discretization of elliptic RPDEs. By exploring the low-dimensional structures in the solution space, these methods can alleviate the curse of dimensionality to a certain extent. For general UQ problems with high-dimensional inputs, however, these methods are still very expensive.

The standard Monte Carlo (MC) method is a very general method for computing the expected value of a solution to a UQ problem and it has the advantage that its performance does not depend on the dimension of the random input. However, its computational cost will be very expensive because random fields have low spatial regularity, making solving such problems computationally inefficient. In addition, we know that the standard MC method has a slow convergence rate, which means that a large number of simulations are required to obtain accurate results. Therefore, the standard MC method is more suitable for low-precision simulation, and high-precision simulation is less used due to the large computational cost. The computational cost of solving the elliptic partial differential equation (PDE) with random coefficients is thus a major challenge in quantifying uncertainty in groundwater flow studies and composite material simulations.

To address the limitations of the standard MC method, Giles proposed the multilevel Monte Carlo method (MLMC) method in [1], which is inspired by the multilevel grid method, based on the expected linear properties, through a geometric step sequence $\frac{h_{\ell-1}}{h_\ell} = s \in \mathbb{N}/\{1\},\ \forall \ell = 1, \ldots, L$ to iteratively solve the system of linear equations produced by the discretization of elliptic PDEs [2, 3]. Multilevel grids are corrected by using all grid calculations, solving equations on the finest grid, i.e. maintaining the accuracy associated with the smallest step size, but using larger step size calculations to reduce variance, thus minimizing the overall computational complexity. In short, the MLMC method assumes a fixed initial grid h_0 and implements the optimal policy by testing different layers L.

In this paper, we propose an improved MLMC method, which aims to further improve the computational efficiency of the MLMC method. We assume that the number of grid layers L is fixed, and the optimal policy is achieved by adjusting the initial grid h_0. The idea is to omit computations on particularly coarse grids to achieve faster convergence, while taking into account the advantages of standard MLMC methods that balance computation and accuracy. We explain how these computational costs are reduced through theoretical demonstration, and demonstrate the effectiveness of the improved method through numerical results of multiple sets of random coefficient elliptic differential equations in both one-dimensional and two-dimensional cases. The improved MLMC can indeed save several times or even an order of magnitude of computation compared to the classical MLMC method under the accuracy requirement.

The outline of this article is as follows. In Section 2, we review the MLMC method and show its computational complexity, which estimates the cost of an method under certain, problem-related assumptions. In Section 3, we propose an improved MLMC method and prove the computational complexity of the improved MLMC method. In Section 4, we present the equations for the groundwater flow modeling problem to describe our stochastic model, and show numerical results for one-dimensional and two-dimensional problems to confirm the effectiveness of the improved method. Finally, conclusions and discussions are drawn in Section 5.

2. Review of Multilevel Monte Carlo method

this section is revised and looks fine. To illustrate the MLMC method, we consider the following stochastic PDE as an example

$$\mathcal{L}u(x,\omega) = f(x,\omega), \qquad x \in D, \omega \in \Omega, \tag{2.1}$$ eqn:de

where $\mathcal{L}$ is a generic linear operator, $D \subset \mathbb{R}^d$ is the domain of spatial variables, and Ω is the domain of the random variable. Some boundary conditions should be imposed for the well-posedness.

We first briefly review the MLMC method for solving the stochastic PDE (2.1). Suppose the quantity of interest is the expected value of a functional $Q(\omega) = \mathcal{Q}(u(\cdot,\omega))$ of the solution $u(x,\omega)$ to the stochastic PDE (2.1). In general, u cannot be solved exactly. Therefore, Q is often approximated by $Q_h := \mathcal{Q}(u_h)$ with u_h a finite dimensional approximation to u, such as the finite difference or finite element solution on a fine spatial grid $\mathcal{T}_h$.

To estimate $\mathbb{E}[Q]$, we compute a statistical estimator $\hat{Q}_h$ to $\mathbb{E}[Q_h]$ first, and the error of the approximation to $\mathbb{E}[Q]$ is quantified by the root mean square error (RMSE)

$$e(\hat{Q}_h) \triangleq \left(\mathbb{E}\left[(\hat{Q}_h - \mathbb{E}[Q])^2\right]\right)^{1/2}.$$

The computational cost $C(\hat{Q}_h)$ of the estimator is quantified by the number of floating point operations that are needed to achieve a RMSE of $e(\hat{Q}_h) \leqslant \delta$ with δ the given tolerance.

The classical Monte Carlo estimator for $\mathbb{E}[Q_h]$ is

$$\hat{Q}^{\mathrm{MC}}_{h,N} = \frac{1}{N}\sum_{i=1}^{N} Q_h(\omega^{(i)}), \tag{2.2}$$ eqn:mc

where $Q(\omega^{(i)})$ is the i-th sample of Q_h.

There are two sources of error in the estimator (2.2): the approximation error of Q by Q_h, which depends on the spatial discretization, and the sampling error due to the replacement of the expected value by the finite sample average. Since $\mathbb{E}[\hat{Q}^{\mathrm{MC}}_{h,N}] = \mathbb{E}[Q_h]$ and $\mathbb{V}[\hat{Q}^{\mathrm{MC}}_{h,N}] = N^{-1}\mathbb{V}[Q_h]$, we have the decomposition of the error as the contribution from the variance and the bias as follows

$$e(\hat{Q}^{\mathrm{MC}}_{h,N})^2 = N^{-1}\mathbb{V}[Q_h] + (\mathbb{E}[Q_h - Q])^2. \tag{2.3}$$ eqn:mcerro

To control these two terms of errors, in general we require both terms are less than $\delta^2/2$ so that we can achieve a RMSE of δ. Therefore, the sample size is set as $N = \mathcal{O}(\delta^{-2})$ for the first term and the variance can be further reduced by importance sampling for rare-event type of Q [?]. **cite my own work** For the second term of bias error, one needs to choose a sufficiently fine grid size such that $\mathbb{E}[Q_h - Q] = \mathcal{O}(\delta)$. The total computational complexity can be obtained easily.

The MLMC method was proposed to reduce the computational complexity of the classical MC method in computing $\mathbb{E}[Q]$. The main idea of the MLMC estimator is as follows. Instead of sampling the same number of realizations on one mesh, MLMC computes $\mathbb{E}[Q_h]$ on a sequence of nested meshes and by the linearity of the expectation, it is obvious that

$$\mathbb{E}[Q_h] = \mathbb{E}[Q_{h_0}] + \sum_{\ell=1}^{L} \mathbb{E}[Q_{h_\ell} - Q_{h_{\ell-1}}], \tag{2.4}$$

eqn:meanML

where $\{h_\ell\}_{\ell=0}^{L}$ are mesh sizes of a sequence of increasingly refined meshes with $h = h_L$ the finest mesh size and $h = h_0$ the coarsest one. We set

$$h_{\ell-1}/h_\ell \equiv s > 1$$

so that $h_L = h_0 s^{-L}$.

Define $Q_{h_{-1}} = 0$ and define the MLMC estimator

$$\hat{Q}^{\mathrm{ML}}_{h,\{N_\ell\}} \triangleq \sum_{\ell=0}^{L} \frac{1}{N_\ell} \sum_{i=1}^{N_\ell} \left(Q_{h_\ell}(\omega^{(\ell,i)}) - Q_{h_{\ell-1}}(\omega^{(\ell,i)}) \right). \tag{2.5}$$

eqn:mlmc

where the subindex"h" means a sequence of (h_ℓ). Then, the MLMC error is

$$e(\hat{Q}^{\mathrm{ML}}_{h,\{N_\ell\}})^2 = \mathbb{E}\left[(\hat{Q}^{\mathrm{ML}}_{h,\{N_\ell\}} - \mathbb{E}[Q])^2\right] = \sum_{\ell=0}^{L} N_\ell^{-1} \mathbb{V}[Q_{h_\ell} - Q_{h_{\ell-1}}] + (\mathbb{E}[Q_{h_L} - Q])^2. \tag{2.6}$$

eqn:mlmcer

Let C_ℓ denote the cost of obtaining one sample of Q_{h_ℓ}. Then the following result for the MLMC estimator (e.g., [1, 3, 4, 5, 7]) is classic.

thm:mlmc **Proposition 2.1** ([3], **Theorem 2.3**). *Suppose there exist positive constants* $\alpha, \beta, \gamma, c_1, c_2, c_3$ *such that* $\alpha \geq \frac{1}{2}\min(\beta, \gamma)$ *and*

A1. $|\mathbb{E}[Q_h - Q]| \leq c_1 h_\ell^{\alpha}$,

A2. $\mathbb{V}[Q_{h_\ell} - Q_{h_{\ell-1}}] \leqslant c_2 h_\ell^{\beta}$,

A3. $C_\ell \leq c_3 h_\ell^{-\gamma}$.

where c_1, c_2, c_3 is independent of h_l. Then, for a fixed initial meshsize $h_0 > 0$ and a fixed mesh refinement ratio $s > 1$, there exists a positive constant c_4 depending on c_1, c_2, c_3, h_0 and s, such that for any $\delta < e^{-1}$ there exist a positive integer $L > 0$ and a sequence of positive integers $N_\ell, 0 \leq \ell \leq L$, for which the multilevel estimator satisfies the error $e(\hat{Q}^{\mathrm{ML}}_{h,\{N_\ell\}}) < \delta$ and the total cost is

$$C = \begin{cases} c_4\delta^{-2}, & \beta > \gamma, \\ c_4\delta^{-2}(\log\delta)^2, & \beta = \gamma, \\ c_4\delta^{-2-(\gamma-\beta)/\alpha}, & \beta < \gamma. \end{cases} \tag{2.7}$$

eqn:mlmcco

remark:mc3 **Remark 2.1.** *By (2.3), the total computational complexity of the MC method with Assumptions* ***A1*** *and* ***A3*** *is*

$$C(\hat{Q}^{\mathrm{MC}}_{h,N}) \leqslant c\delta^{-2-\gamma/\alpha}. \tag{2.8}$$

eqn:comple

where c depends on the variance $\mathbb{V}[Q_h]$. It is easy to see that the MLMC method is superior to the MC method, as far as the scaling of δ is concerned.

Remark 2.2. *There are many works trying to eliminate the bias term in (2.6) completely in Markov chain setting or diffusion setting, like exact simulation and exact simulation* [**?**] [**?**]. **cite both** *These works could be viewed as a randomized version of MLMC.*

3. An improved Multilevel Monte Carlo method

In this section, we propose an improved MLMC method that achieves the same or less computational complexity than in (2.7). Our proof is also constructive as before. But our analysis is based on the investigation of the initial mesh size h_0, while the number of mesh levels L is fixed. Our result is the following theorem.

Theorem 3.1. *Suppose that there exist positive constants $\alpha, \beta, \gamma, c_1, c_2, c_3$ such that $\alpha \geq \beta/2$ and assume* ***A1***, ***A2***, ***A3*** *as in Proposition 2.1. Then, for a fixed total number of mesh levels $L \geq 1$ and a fixed mesh refinement ratio $s > 1$, there exists a positive constant c_4 depending on c_1, c_2, c_3, L, and s such that, for any $\delta < 1$, there are a sequence of $\{N_\ell\}$ for which the multilevel estimator has the error bound $e(\hat{Q}^{\mathrm{ML}}_{h,\{N_\ell\}}) < \delta$ and the total cost*

$$C = c_4\delta^{-2+(\beta-\gamma)/\alpha} = \begin{cases} c_4\delta^{-2+(\beta-\gamma)/\alpha}, & \beta > \gamma, \\ c_4\delta^{-2}, & \beta = \gamma, \\ c_4\delta^{-2-(\gamma-\beta)/\alpha}, & \beta < \gamma. \end{cases} \tag{3.1}$$

Remark 3.2. *Compared with the complexity (2.7) of the classical MLMC method in Proposition 2.1, the improved complexity estimate (3.1) reduces the cost by a factor of $\delta^{\beta/\alpha}$ when $\beta > \gamma$ and by a factor of $(\log\delta)^2$ when $\beta = \gamma$.*

Proof. From Assumption ***A3***, the (upper bound of) total complexity reads as

$$C = \sum_{\ell=0}^{L} c_3 h_\ell^{-\gamma} N_\ell. \tag{3.2}$$

In the sequel, we assume that the total number of mesh levels L and the mesh refinement ratio s are both fixed. Using $h_\ell = h_0 s^{-\ell}$, (3.2) becomes

$$C = c_3 h_0^{-\gamma} \sum_{\ell=0}^{L} N_\ell s^{\gamma\ell}. \tag{3.3}$$

To achieve a RMSE of δ, from (2.6) and Assumptions ***A1***, ***A2***, we need

$$\sum_{\ell=0}^{L} c_2 h_\ell^{\beta} N_\ell^{-1} + c_1^2 h_L^{2\alpha} \leqslant \delta^2. \tag{3.4}$$

We require that each term in (3.4) is less than $\delta^2/2$, i.e.,

$$c_2 h_0^{\beta} \sum_{\ell=0}^{L} N_\ell^{-1} s^{-\beta\ell} \leqslant \frac{\delta^2}{2}, \tag{3.5}$$

$$c_1^2 h_0^{2\alpha} s^{-2\alpha L} \leqslant \frac{\delta^2}{2}. \tag{3.6}$$

A upper bound of the free parameter h_0 is derived from (3.6) that

$$0 < h_0 \leqslant \left(\frac{\delta}{\sqrt{2}c_1}\right)^{1/\alpha} s^L, \tag{3.7}$$

which will be used later. With the upper bound of h_0 and the constraint (3.5) for both h_0 and N_ℓ, we aim to reduce (3.3).

Let $\lceil z \rceil$ be the unique integer satisfying the inequalities $z \leqslant \lceil z \rceil < z+1$. We now proceed with different possible values for β and γ.

(a) Firstly we consider the case when $\beta = \gamma$. We set $N_\ell := \lceil 2\delta^{-2}(L+1)c_2 h_0^{\beta} s^{-\beta\ell} \rceil$ and hence

$$2\delta^{-2}(L+1)c_2 h_0^{\beta} s^{-\beta\ell} \leq N_\ell < 2\delta^{-2}(L+1)c_2 h_0^{\beta} s^{-\beta\ell} + 1. \tag{3.8}$$

It is easy to verify that

$$c_2 h_0^{\beta} \sum_{\ell=0}^{L} N_\ell^{-1} s^{-\beta\ell} \leq c_2 h_0^{\beta} \sum_{\ell=0}^{L} \frac{1}{2\delta^{-2}(L+1)c_2 h_0^{\beta} s^{-\beta\ell}} s^{-\beta\ell} = \frac{\delta^2}{2},$$

i.e., (3.5) holds true. Substituting (3.8) into (3.3) and using the fact that $\beta = \gamma$, we can obtain the upper bound for the complexity as below

$$\begin{aligned} C = c_3 h_0^{-\gamma} \sum_{\ell=0}^{L} N_\ell s^{\gamma\ell} &< c_3 h_0^{-\gamma} \left(2\delta^{-2}(L+1)^2 c_2 h_0^{\beta} + \sum_{\ell=0}^{L} s^{\gamma\ell} \right) \\ &= 2c_2c_3\delta^{-2}\left(L+1\right)^2 + c_3 h_0^{-\gamma} \frac{1-s^{\gamma(L+1)}}{1-s^{\gamma}}. \end{aligned}$$

Since $\alpha \geq \gamma/2$, we take h_0 exactly as the upper bound in (3.7) and get

$$C \leqslant 2c_2c_3\delta^{-2}\left(L+1\right)^2 + c_3\left(\sqrt{2}c_1\right)^{\gamma/\alpha}\delta^{-\gamma/\alpha}\frac{1-s^{\gamma(L+1)}}{1-s^{\gamma}}s^{-\gamma L} \leq c_4\delta^{-2}$$

with

$$c_4 \triangleq 2c_2c_3\left(L+1\right)^2 + c_3\left(\sqrt{2}c_1\right)^{\gamma/\alpha}\frac{1-s^{\gamma(L+1)}}{1-s^{\gamma}}s^{-\gamma L}. \tag{3.9}$$

eqn:c41mlm

(b) Secondly we consider the case when $\beta > \gamma$. We set

$$N_\ell := \lceil 2\delta^{-2}c_2h_0^{\beta}\frac{1-s^{-\frac{(\beta-\gamma)}{2}(L+1)}}{1-s^{-\frac{\beta-\gamma}{2}}}s^{-(\beta+\gamma)\ell/2}\rceil \tag{3.10}$$

eqn:Nbound

to guarantee the inequality (3.5). Substituting (3.10) into (3.3), we have

$$\begin{aligned} C &= c_3h_0^{-\gamma}\sum_{\ell=0}^{L}N_\ell s^{\gamma\ell} < c_3h_0^{-\gamma}\left(2\delta^{-2}c_2h_0^{\beta}\left(\frac{1-s^{-\frac{(\beta-\gamma)}{2}(L+1)}}{1-s^{-\frac{\beta-\gamma}{2}}}\right)^2 + \sum_{\ell=0}^{L}s^{\gamma\ell}\right) \\ &= 2c_2c_3h_0^{\beta-\gamma}\delta^{-2}\left(\frac{1-s^{-\frac{(\beta-\gamma)}{2}(L+1)}}{1-s^{-\frac{\beta-\gamma}{2}}}\right)^2 + c_3h_0^{-\gamma}\frac{1-s^{\gamma(L+1)}}{1-s^{\gamma}}. \end{aligned} \tag{3.11}$$

eqn:comple

Since $\alpha \geq \beta/2$, we again choose h_0 exactly as the upper bound in (3.7) and obtain

$$\begin{aligned} C \leq\ & \frac{2c_2c_3}{(\sqrt{2}c_1)^{\frac{\beta-\gamma}{\alpha}}}s^{(\beta-\gamma)L}\left(\frac{1-s^{-\frac{(\beta-\gamma)}{2}(L+1)}}{1-s^{-\frac{\beta-\gamma}{2}}}\right)^2\delta^{-2+\frac{\beta-\gamma}{\alpha}} \\ &+ c_3(\sqrt{2}c_1)^{\frac{\gamma}{\alpha}}s^{-\gamma L}\frac{1-s^{\gamma(L+1)}}{1-s^{\gamma}}\delta^{-\frac{\gamma}{\alpha}} \\ \leqslant\ & c_4\delta^{-2+\frac{\beta-\gamma}{\alpha}} \end{aligned}$$

with

$$c_4 \triangleq \frac{2c_2c_3}{(\sqrt{2}c_1)^{\frac{\beta-\gamma}{\alpha}}}s^{(\beta-\gamma)L}\left(\frac{1-s^{-\frac{(\beta-\gamma)}{2}(L+1)}}{1-s^{-\frac{\beta-\gamma}{2}}}\right)^2 + c_3(\sqrt{2}c_1)^{\frac{\gamma}{\alpha}}s^{-\gamma L}\frac{1-s^{\gamma(L+1)}}{1-s^{\gamma}}. \tag{3.12}$$

defofc4

(c) Finally we consider the case when $\beta < \gamma$. By choosing N_ℓ as defined in (3.10), the inequality (3.5) is satisfied. Moreover, the estimate of the total complexity in (3.11) is still valid while the upper bound in (3.11) is an decreasing function of h_0. It follows from (3.7) and $\alpha \geq \beta/2$ that

$$\begin{aligned} C \leq & \frac{2c_2c_3}{(\sqrt{2}c_1)^{\frac{\beta-\gamma}{\alpha}}} s^{-(\gamma-\beta)L} \left(\frac{1 - s^{\frac{(\gamma-\beta)}{2}(L+1)}}{1 - s^{\frac{\gamma-\beta}{2}}} \right)^2 \delta^{-2+\frac{\beta-\gamma}{\alpha}} \\ & + c_3(\sqrt{2}c_1)^{\frac{\gamma}{\alpha}} s^{-\gamma L} \frac{1 - s^{\gamma(L+1)}}{1 - s^{\gamma}} \delta^{-\frac{\gamma}{\alpha}} \\ & \leqslant c_4 \delta^{-2+\frac{\beta-\gamma}{\alpha}} \end{aligned}$$

with c_4 defined in (3.12).

□

Remark 3.3. *When $\beta > \gamma$, the optimal upper bound in (3.11) can be achieved when*

$$h_0 = \frac{1}{(2c_2)^{1/\beta}} \left(\frac{1 - s^{\gamma(L+1)}}{1 - s^{\gamma}} \left[\frac{1 - s^{-\frac{(\beta-\gamma)}{2}(L+1)}}{1 - s^{-\frac{\beta-\gamma}{2}}} \right]^{-2} \right)^{1/\beta} \delta^{2/\beta}. \tag{3.13}$$

eqn:h0opti

However, we did not use the optimal choice of h_0 since it may not satisfy (3.7) for the given parameters.

4. Numerical results

In this section, we will conduct numerical experiments to investigate the performance of the improved MLMC method. To be specific, we consider the following elliptic PDE with random coefficients

$$\begin{cases} -\nabla \cdot (a(x,\omega)\nabla u(x,\omega)) = f(x,\omega) & x \in D, \\ u(x,\omega) = 0 \quad x \in \partial D \end{cases} \tag{4.1}$$

eqn:PDE

for almost all $\omega \in \Omega$.

Following [5], we consider a log-normal field where $a(x, \omega) = \exp(g(x, \omega))$ and $g : \bar{D} \times \Omega \to \mathbb{R}$ is a Gaussian field with zero mean and Lipschitz continuous covariance kernel

$$C(x, y) = \mathbb{E}\left[(g(x, \omega) - \mathbb{E}[g(x, \omega)])\,(g(y, \omega) - \mathbb{E}[g(y, \omega)])\right] = k(\|x - y\|)$$

for some Lipschitz continuous function $k \in C^{0,1}(\mathbb{R}^+)$ and for some norm $\|\cdot\|$ in $\mathbb{R}^d$.

Two types of covariance functions will be taken into account: one is the Gaussian function

$$k(r) = \sigma^2 \exp\left(-r^2/\lambda^2\right) \tag{4.2}$$

eqn:Gauss

which is smooth, and the other is the exponential function

$$k(r) = \sigma^2 \exp\left(-r/\lambda\right) \tag{4.3}$$

eqn:exp

which is only Lipschitz continuous. In both covariance functions, λ is the correlation length and σ^2 is the variance.

From Proposition 4.2 and Proposition 4.3 in [5], we can derive the parameters in Theorem 3.1. For a log-normal field with Gaussian covariance function (4.2), the constants $\alpha = 1, \beta = 2$ when the quantity of interest $\mathcal{Q}(u) = |u|_{H^1(D)}$ and $\alpha = 2, \beta = 4$ when $\mathcal{Q}(u) = \|u\|_{L^2(D)}$, respectively. For a log-normal field with exponential covariance function (4.3), $\alpha = 1/2, \beta = 1$ when $\mathcal{Q}(u) = |u|_{H^1(D)}$ and $\alpha = 1, \beta = 2$ when $\mathcal{Q}(u) = \|u\|_{L^2(D)}$, respectively. In all the cases, $\alpha = \beta/2$ which implies the validity of the assumption $\alpha \geq \beta/2$ in Theorem 3.1. Moreover, we have $\gamma = d$ if a solver with linear computational complexity is employed, such as multigrid solver; see [7, 5] for more details.

Table 1 lists the theoretical upper bounds for δ-costs of the MC method, the classical MLMC method, and the improved MLMC (iMIMC for short) method in the case of a log-normal field with Gaussian covariance function and with exponential covariance function, respectively. For simplicity we keep the δ terms while drop the logarithmic factors. We can see that the theoretical upper bounds for δ-cost of the improve MLMC method is the least among the three methods which is highlighted in red.

	Gaussian covariance function						Exponential covariance function					
	$\mathcal{Q}(u) = \lvert u\rvert_{H^1(D)}$			$\mathcal{Q}(u) = \lVert u\rVert_{L^2(D)}$			$\mathcal{Q}(u) = \lvert u\rvert_{H^1(D)}$			$\mathcal{Q}(u) = \lVert u\rVert_{L^2(D)}$		
d	MC	MLMC	iMLMC	MC	MLMC	iMLMC	MC	MLMC	iMLMC	MC	MLMC	iMLMC
1	δ^{-3}	δ^{-2}	δ^{-1}	$\delta^{-5/2}$	δ^{-2}	$\delta^{-1/2}$	δ^{-4}	δ^{-2}	δ^{-2}	δ^{-3}	δ^{-2}	δ^{-1}
2	δ^{-4}	δ^{-2}	δ^{-2}	δ^{-3}	δ^{-2}	δ^{-1}	δ^{-6}	δ^{-4}	δ^{-4}	δ^{-4}	δ^{-2}	δ^{-2}
3	δ^{-5}	δ^{-3}	δ^{-3}	$\delta^{-7/2}$	δ^{-2}	$\delta^{-3/2}$	δ^{-8}	δ^{-6}	δ^{-6}	δ^{-5}	δ^{-3}	δ^{-3}

TABLE 1. Theoretical upper bounds of δ-costs for three different methods in the case of a log-normal field with two covariance functions.

b:theocomp

4.1. **1D problems.** Let us start by solving the 1D problem of (4.1) in D = (0,1) and $f \equiv 1$ with boundary conditions $u(0) = u(1) = 0$. The quantity of interest $\mathcal{Q}(u)$ is chosen as $||u||_{L^2(D)}$ and $|u|_{H^1(D)}$, respectively. We first numerically verify the assumptions in **Theorem 3.1**, and estimate the values of the parameters α and β to ensure the decay of the variance of $Q_{h_\ell} - Q_{h_{\ell-1}}$ for each level. Then we study the efficiency of the improved MLMC method in terms of the accuracy and the computational cost. For brevity, we denote $Q_\ell := Q_{h_\ell}$ in the following.

4.1.1. *Gaussian covariance function.* There exist several techniques to produce samples of the coefficient $a(x, \omega)$, including the circulant embedding [6, 8, 9] and the truncated KL-expansion. Here we employ the circulant embedding technique. Given a log-normal field with Gaussian covariance function (4.2), for $\lambda = 0.3, \sigma^2 = 1$, Figure 1 shows the behaviour of the variance and the expected value of Q_ℓ and $Q_\ell - Q_{\ell-1}$ when $\mathcal{Q}(u) = ||u||_{L^2(D)}$ and $\alpha \approx 1.9998, \beta \approx 3.6100$.

FIGURE 1. 1D problem. Plots of the variance (left) and the expected value (right) for Q_ℓ and $Q_\ell - Q_{\ell-1}$ in the case of a log-normal field with Gaussian covariance function when $\mathcal{Q}(u) = ||u||_{L^2(D)}$. tab:Varian

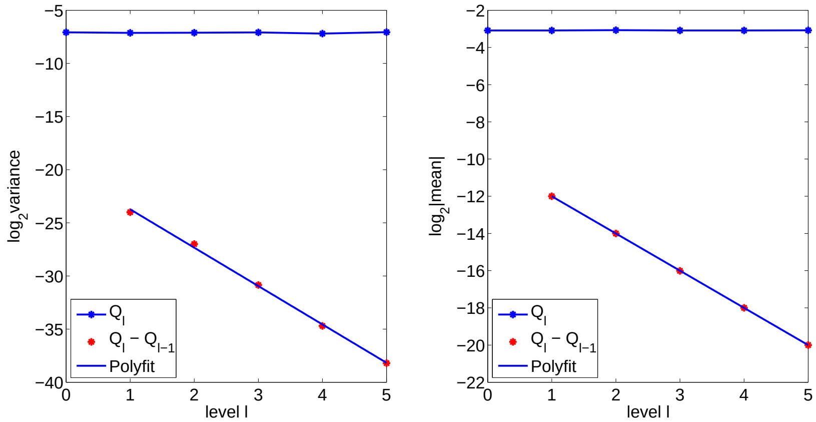
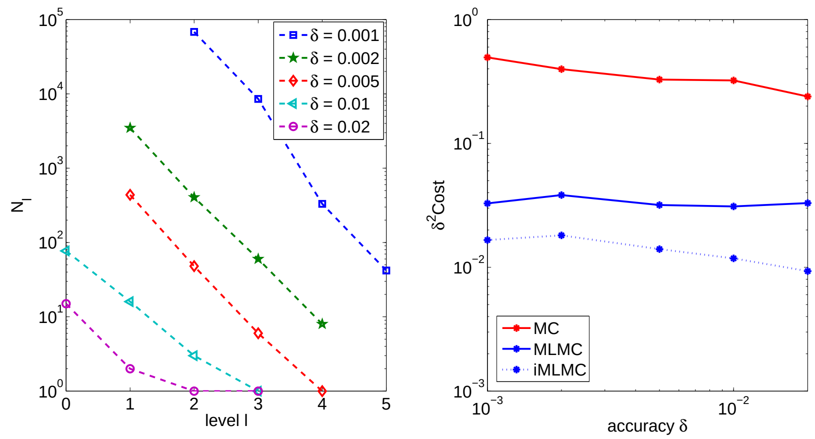


FIGURE 2. 1D problem. Left: Number of samples N_ℓ on each level in the improved MLMC method for different RMSEs δ; Right: Cost scaled by δ^2 of three methods in the case of a log-normal field for Gaussian covariance function with $\lambda = 0.3, \sigma^2 = 1$ when $\mathcal{Q}(u) = ||u||_{L^2(D)}$. tab:Gaussi



Numerical experiments are performed with the setting of the above parameters. We investigate the improved MLMC method with the fixed total number of mesh levels $L = 4$ and the fixed mesh refinement ratio $s = 2$. To achieve a RMSE of δ, after traversing to find the optimal strategy, the sampling numbers N_ℓ used on each level are shown in the left plot of Figure 2. The right plot gives a comparison of the costs among the MC method, the classical MLMC method and the improved MLMC method, which exhibits a significant advantage of the improved MLMC method. Note that the finest mesh size in our experiment is $h = 1/2^{12}$ and hence $\mathbb{E}[||u_h||_{L^2(D)}] \approx 0.1177$. The cost on the vertical axis of the plot is calculated as $N_0 + \sum_{\ell=1}^{L} N_\ell \frac{T_\ell + T_{\ell-1}}{T_0}$, where T_ℓ is CPU time of computing numerical solutions when the mesh size is h_ℓ.

In Table 2, we record the computational costs for δ-accuracy of the MC method, the classical MLMC method and the improved MLMC method in the case of a log-normal field with Gaussian covariance function with $\lambda = 0.3, \sigma^2 = 1$ when $\mathcal{Q}(u) = ||u||_{L^2(D)}$. It is easy to see that the computational cost of the improved MLMC method is almost half of that of the classical MLMC method without sacrifice of the accuracy.

ab:Cost_L2

TABLE 2. 1D problem. δ^2-Cost and actual error to achieve the accuracy δ for three methods in the case of a log-normal field with Gaussian covariance function for $\lambda = 0.3, \sigma^2 = 1$ when $\mathcal{Q}(u) = ||u||_{L^2(D)}$.

	δ^2-Cost			Actual Error		
δ	MC	MLMC	iMLMC	MC	MLMC	iMLMC
0.02	0.2391	0.0330	0.0093	1.90e-2	4.60e-3	9.30e-3
0.01	0.3227	0.0310	0.0118	9.50e-3	5.90e-3	4.10e-3
0.005	0.3275	0.0318	0.0140	5.00e-3	1.40e-3	2.55e-4
0.002	0.3979	0.0383	0.0181	2.00e-3	1.70e-3	3.55e-4
0.001	0.4947	0.0328	0.0166	1.00e-3	3.25e-4	4.00e-4

4.1.2. *Exponential covariance function.* Now let us take a look at the case of a log-normal field with exponential covariance function (4.3) for $\lambda = 0.3, \sigma^2 = 1$. In Figure 3, we can find the behaviour of the variance and the expected value of Q_ℓ and $Q_\ell - Q_{\ell-1}$ when $\mathcal{Q}(u) =$

$||u||_{L^2(D)}$ and $\alpha \approx 1.0068, \beta \approx 1.9981$. Note that the finest mesh size is still chosen as $h = 1/2^{12}$, which gives $\mathbb{E}[||u_h||_{L^2(D)}] \approx 0.1254$. From Figure 4 and Table 3, we can come to the same conclusion that the improved MLMC method outperforms over the MC method and the classical MLMC method.

FIGURE 3. 1D problem. Plots of the variance (left) and the expected value (right) for Q_ℓ and $Q_\ell - Q_{\ell-1}$ in the case of a log-normal field with exponential covariance function when $\mathcal{Q}(u) = ||u||_{L^2(D)}$. tab:Varian

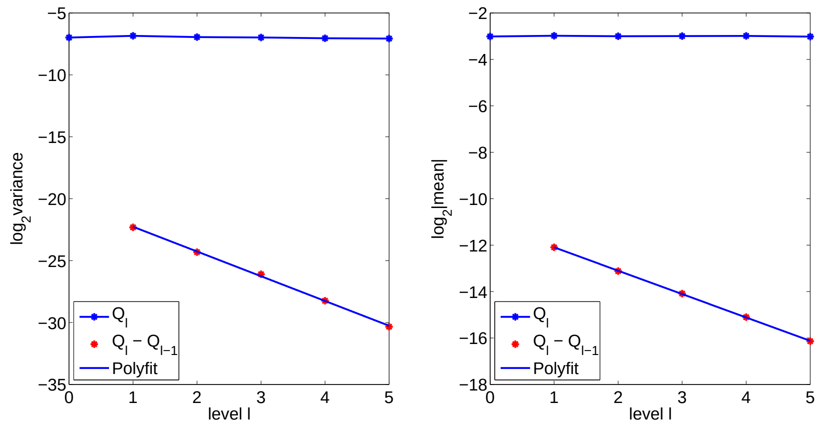
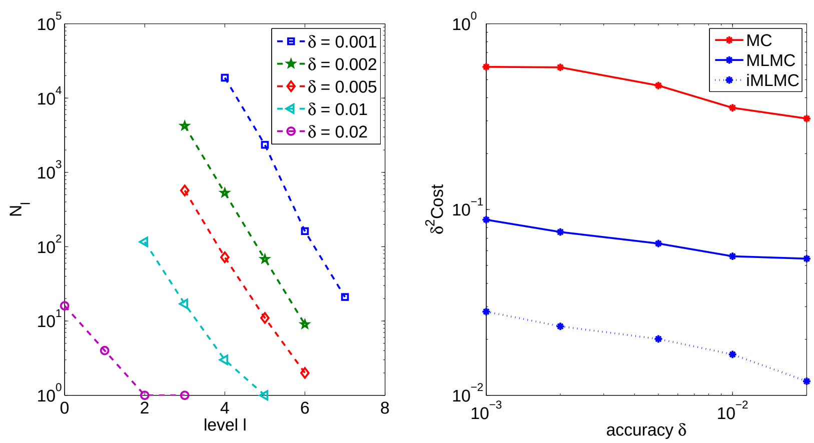


FIGURE 4. 1D problem. Left: Number of samples N_ℓ on each level in the improved MLMC method for different RMSEs δ; Right: Cost scaled by δ^2 of three methods in the case of a log-normal field with exponential covariance function for $\lambda = 0.3, \sigma^2 = 1$ when $\mathcal{Q}(u) = ||u||_{L^2(D)}$. tab:exp_L2



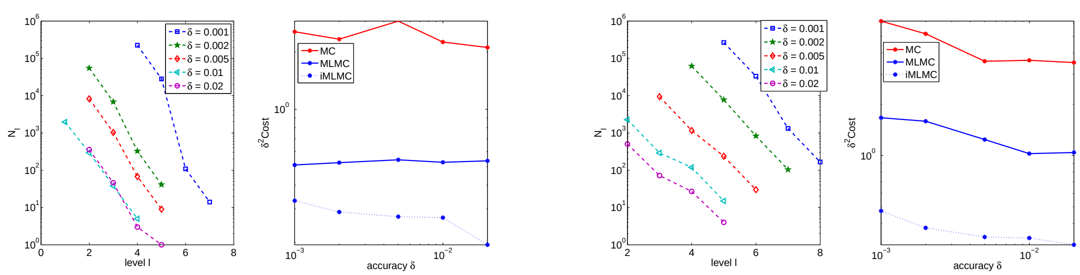
Furthermore, we consider the case when the quantity of interest $\mathcal{Q}(u) = |u|_{H^1(D)}$. Given a log-normal field with Gaussian covariance function and exponential covariance function ost_exp_L2

TABLE 3. 1D problem. δ^2-Cost and actual error to achieve the accuracy δ for three methods in the case of a log-normal field with exponential covariance function for $\lambda = 0.3, \sigma^2 = 1$ when $\mathcal{Q}(u) = ||u||_{L^2(D)}$.

	δ^2-Cost			Actual Error		
δ	MC	MLMC	iMLMC	MC	MLMC	iMLMC
0.02	0.3090	0.0544	0.0119	1.98e-02	9.20e-3	1.07e-2
0.01	0.3528	0.0560	0.0166	9.70e-3	8.10e-3	5.00e-4
0.005	0.4648	0.0656	0.0201	4.90e-3	2.00e-3	1.00e-4
0.002	0.5825	0.0757	0.0235	2.00e-3	1.10e-3	2.64e-4
0.001	0.5861	0.0881	0.0282	1.00e-3	7.05e-4	4.11e-4

FIGURE 5. 1D problem. Left: Number of samples N_ℓ on each level in the improved MLMC method for different RMSEs δ; Right: Cost scaled by δ^2 of three methods in the case of a log-normal field with two covariance functions for $\lambda = 0.3, \sigma^2 = 1$ when $\mathcal{Q}(u) = |u|_{H^1(D)}$.

all_1D_H1



(a) Gaussian covaraince function (b) Exponential covaraince function

for $\lambda = 0.3, \sigma^2 = 1$, the corresponding numerical results are shown in Figure 5 and Table 4. For the same accuracy, the improved MLMC method reduces more than half of the computational cost compared with the classical MLMC method.

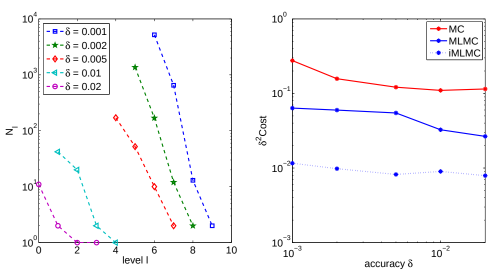
Finally we change the value of λ from 0.3 into 0.03. In this case, the random field is rougher, while the improved MLMC method still plays a very good role in improving the computatioanl efficiency. For the detailed results, please refer to Figure 6 and Table 5 - Table 6.

ll_1D_H1_1

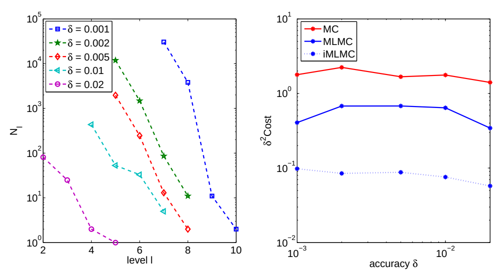
TABLE 4. 1D problem. δ^2-Cost and actual error to achieve the accuracy δ for three methods in the case of a log-normal field with two covariance functions for $\lambda = 0.3, \sigma^2 = 1$ when $\mathcal{Q}(u) = |u|_{H^1(D)}$.

	Gaussian covaraince function						Exponential covariance function					
	δ^2-Cost			Actual Error			δ^2-Cost			Actual Error		
δ	MC	MLMC	iMLMC	MC	MLMC	iMLMC	MC	MLMC	iMLMC	MC	MLMC	iMLMC
0.02	2.1133	0.5364	0.1943	1.99e-2	4.50e-3	6.00e-4	3.5125	1.0412	0.2992	1.97e-2	9.60e-3	1.80e-3
0.01	2.2568	0.5269	0.2698	1.00e-2	6.00e-3	1.20e-3	3.6207	1.0258	0.3278	9.70e-3	7.30e-3	6.30e-3
0.005	2.9076	0.5429	0.2730	5.00e-3	1.70e-3	3.21e-4	3.5763	1.2408	0.3322	5.00e-3	3.30e-3	1.80e-3
0.002	2.3364	0.5247	0.2888	2.00e-3	1.70e-3	4.45e-4	5.1800	1.5892	0.3761	2.00e-3	1.30e-3	9.43e-4
0.001	2.5563	0.5101	0.3312	1.00e-3	4.42e-4	3.01e-4	6.1520	1.6653	0.4729	1.00e-3	5.42e-4	4.67e-4

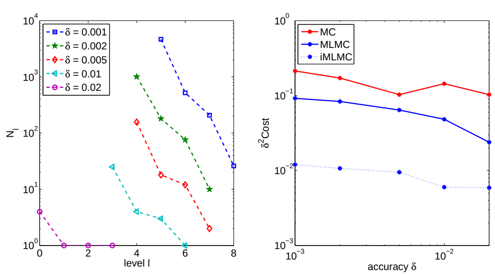
FIGURE 6. 1D problem. Left: Number of samples N_ℓ on each level in the improved MLMC method for different RMSEs δ; Right: Cost scaled by δ^2 of three methods in the case of a log-normal field with two covariance functions for $\lambda = 0.03, \sigma^2 = 1$ when $\mathcal{Q}(u) = ||u||_{L^2(D)}$ and $\mathcal{Q}(u) = |u|_{H^1(D)}$, respectively. all_1D_03



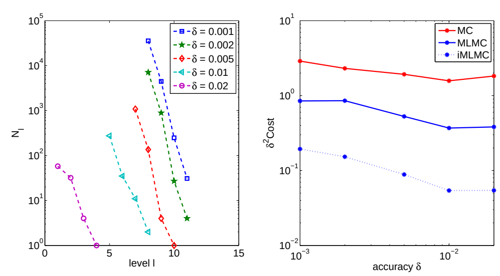
(a) Gaussian function when $\mathcal{Q}(u) = ||u||_{L^2(D)}$



(b) Gaussian function when $\mathcal{Q}(u) = |u|_{H^1(D)}$



(c) Exponential function when $\mathcal{Q}(u) = ||u||_{L^2(D)}$



(d) Exponential function when $\mathcal{Q}(u) = |u|_{H^1(D)}$

:all_1D_03

TABLE 5. 1D problem. δ^2-Cost and actual error to achieve the accuracy δ for three methods in the case of a log-normal field with covariance functions for $\lambda = 0.03, \sigma^2 = 1$ when $\mathcal{Q}(u) = ||u||_{L^2(D)}$.

	Gaussian covaraince function						Exponential covariance function					
	δ^2-Cost			Actual Error			δ^2-Cost			Actual Error		
δ	MC	MLMC	iMLMC	MC	MLMC	iMLMC	MC	MLMC	iMLMC	MC	MLMC	iMLMC
0.02	0.1148	0.0266	0.0079	1.64e-2	1.81e-2	3.40e-3	0.1030	0.0239	0.0059	1.66e-2	1.09e-2	1.45e-2
0.01	0.1100	0.0326	0.0090	9.20e-3	9.70e-3	4.70e-3	0.1442	0.0483	0.0060	9.80e-3	4.70e-4	2.60e-3
0.005	0.1213	0.0548	0.0082	4.50e-3	4.90e-3	2.40e-3	0.1030	0.0642	0.0095	4.90e-3	2.76e-4	2.80e-3
0.002	0.1577	0.0598	0.0098	2.00e-3	1.90e-3	1.00e-4	0.1718	0.0836	0.0107	2.00e-3	5.65e-4	1.60e-3
0.001	0.2758	0.0637	0.0116	1.00e-3	5.23e-4	1.93e-5	0.2132	0.0902	0.0120	1.00e-3	8.00e-4	8.00e-4

Cost_H1_03

TABLE 6. 1D problem. δ^2-Cost and actual error to achieve the accuracy δ for three methods in the case of a log-normal field with two covariance functions for $\lambda = 0.03, \sigma^2 = 1$ when $Q(u) = |u|_{H^1(D)}$.

	Gaussian covariance function						Exponential covariance function					
	δ^2-Cost			Actual Error			δ^2-Cost			Actual Error		
δ	MC	MLMC	iMLMC	MC	MLMC	iMLMC	MC	MLMC	iMLMC	MC	MLMC	iMLMC
0.02	1.4057	0.3435	0.0575	2.00e-2	1.79e-2	1.67e-2	1.8232	0.3824	0.0543	1.99e-2	1.65e-2	1.81e-2
0.01	1.7667	0.6427	0.0757	1.00e-2	4.30e-3	2.80e-3	1.5760	0.3692	0.0541	1.00e-2	5.10e-3	8.00e-3
0.005	1.6741	0.6793	0.0877	5.00e-3	3.30e-3	1.10e-3	1.9237	0.5279	0.0886	5.00e-3	4.00e-3	2.70e-3
0.002	2.2366	0.6781	0.0847	2.00e-3	1.90e-3	7.42e-4	2.3104	0.8550	0.1528	2.00e-3	1.20e-3	2.00e-3
0.001	1.7856	0.4068	0.0979	1.00e-3	1.00e-3	8.20e-5	2.8895	0.8494	0.1939	1.00e-3	4.17e-4	7.32e-4

4.2. **2D problems.** In this subsection, we solve the 2D problem of (4.1) in $D = (0,1)^2$ with $f \equiv 1$. The coefficient $a(x,\omega)$ is chosen as a log-normal random field such that $\log(a)$ has Gaussian covariance function (4.2) or exponential covariance function (4.3) with $\|\cdot\|$ being the 2-norm (i.e., $\|x\| := (x^T x)^{1/2}$). Again, we use the circulant embedding technique to generate samples for the random coefficient $a(x,\omega)$. In addition, it is worthwhile to mention that for one-dimensional problems we use the catch-up method to solve the linear systems of equations, while for the two-dimensional problems we directly use the sparse direct solver provided in Matlab through the standard backslash operation to solve the linear systems of equations for each sample.

4.2.1. *Gaussian covariance function.* Given a log-normal field with Gaussian covariance function (4.2) with $\lambda = 0.3, \sigma^2 = 1$, Figure 7 shows the behaviour of the variance and the expected value of Q_l and $Q_l - Q_{l-1}$ when $Q(u) = ||u||_{L^2(D)}$ and $\alpha \approx 1.5700, \beta \approx 2.9110$. The left plot of Figure 8 is related to the implementation of the improved MLMC method. When the fixed total number of mesh levels $L = 4$, and the fixed mesh refinement ratio $s = 2$, to achieve a RMSE of δ, after traversing to find the optimal strategy, we get the sampling numbers used on each level with $h_0 = 1/2$. The right plot of Figure 8 compares the cost among the MC method, the classical MLMC method and the improved MLMC method. Note that the finest grid in this experiment is $h = 1/2^8$, which gives $\mathbb{E}[||u_h||_{L^2(D)}] \approx 0.0459$. It clearly shows that the improved MLMC method does much better than the classical MLMC method.

In Table 7, we can see the δ^2-Cost and actual error to achieve the accuracy δ of the MC method, the classical MLMC method and the improved MLMC method in the case of a log-normal field with Gaussian covariance function for $\lambda = 0.3, \sigma^2 = 1$ when $Q(u) = ||u||_{L^2(D)}$. Numerical results show that the ratio of the costs between the classical MLMC method and the improved MLMC method with 5 levels is more than 2.

4.2.2. *Exponential covariance function.* Finally, we study the performance of the three methods in the case of a log-normal field with exponential covariance function (4.3) for $\lambda =$

FIGURE 7. 2D problem. Plots of the variance (left) and the expected value (right) of Q_ℓ and $Q_\ell - Q_{\ell-1}$ in the case of a log-normal field with Gaussian covariance function when $\mathcal{Q}(u) = ||u||_{L^2(D)}$.

tab:Var2D_

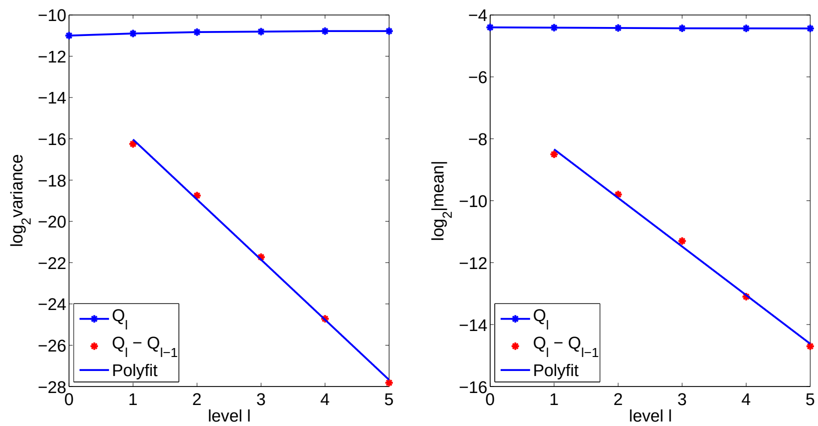
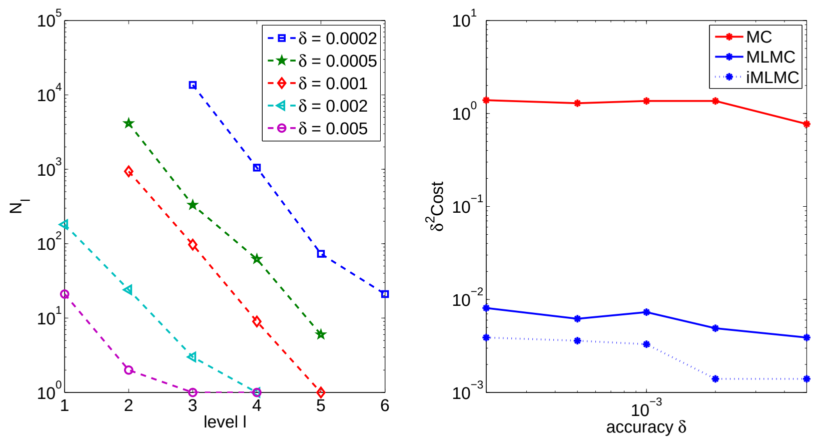


FIGURE 8. 2D problem. Left: Number of samples N_ℓ on each level in the improved MLMC method for different RMSEs δ; Right: Cost scaled by δ^2 of three methods in the case of a log-normal field with Gaussian covariance functions for $\lambda = 0.3, \sigma^2 = 1$ when $\mathcal{Q}(u) = ||u||_{L^2(D)}$.

tab:NNl2D_



$0.3, \sigma^2 = 1$. Plots of the variance and the expected value of Q_l and $Q_l - Q_{l-1}$ are shown in Figure 9, when $Q(u) = ||u||_{L^2(D)}$ and $\alpha \approx 0.9500, \beta \approx 1.9710$.

Note that the finest grid in our experiment is still $h = 1/2^8$, which gives $\mathbb{E}[||u||_{L^2(D)}] \approx 0.0462$. We show the detailed data comparison of the computational cost of the three methods in Figure 10 and Table 8, from which we can see that the computational efficiency of the improved MLMC method is more than doubled compared to the classical MLMC method.

Cost_L2_2D

TABLE 7. 2D problem. δ^2-Cost and actual error to achieve the accuracy δ for three methods in the case of a log-normal field with Gaussian covariance function for $\lambda = 0.3, \sigma^2 = 1$ when $\mathcal{Q}(u) = ||u||_{L^2(D)}$.

	δ^2-Cost			Actual Error		
δ	MC	MLMC	iMLMC	MC	MLMC	iMLMC
0.005	0.7712	0.0039	0.0014	4.90e-3	3.99e-4	8.76e-4
0.002	1.3631	0.0049	0.0014	2.00e-3	8.69e-4	1.00e-4
0.001	1.3646	0.0073	0.0033	1.00e-3	5.84e-4	5.04e-4
0.0005	1.2886	0.0062	0.0036	5.00e-4	3.43e-4	6.61e-5
0.0002	1.3910	0.0081	0.0039	2.00e-4	6.33e-5	1.74e-5

When the the quantity of interest $Q(u) = |u|_{H^1(D)}$ see Figure 11 and Table 9. From the numerical results, we conclude that the computational efficiency has been at least doubled by the improved MLMC method.

FIGURE 9. 2D problem. Plots of the variance (left) and the expected value (right) of Q_ℓ and $Q_\ell - Q_{\ell-1}$ in the case of a log-normal field with exponential covariance function when $\mathcal{Q}(u) = ||u||_{L^2(D)}$.

tab:Var2D_

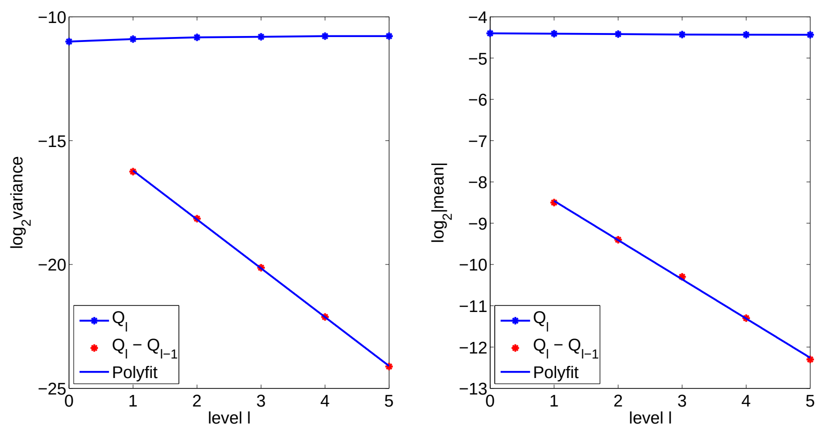


FIGURE 10. 2D problem. Left: Number of samples N_ℓ on each level in the improved MLMC method for different RMSEs δ; Right: Cost scaled by δ^2 of three methods in the case of a log-normal field with exponential covariance functions for $\lambda = 0.3, \sigma^2 = 1$ when $\mathcal{Q}(u) = ||u||_{L^2(D)}$.

tab:NNl2D_

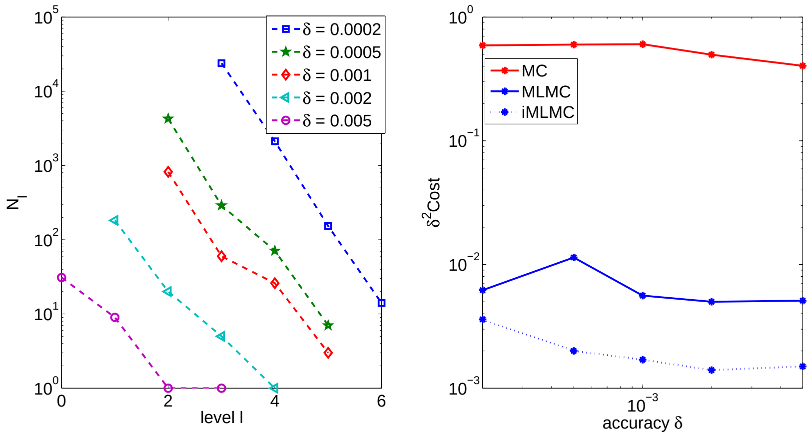


TABLE 8. 2D problem. δ^2-Cost and actual error to achieve the accuracy δ for three methods in the case of a log-normal field with exponential covariance function for $\lambda = 0.3, \sigma^2 = 1$ when $\mathcal{Q}(u) = ||u||_{L^2(D)}$.

_exp_L2_2D

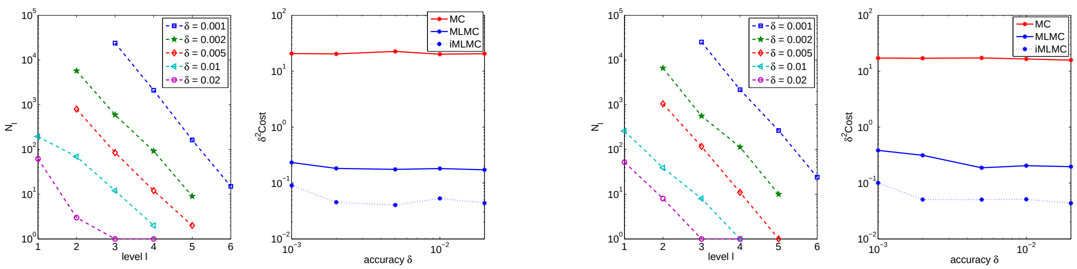
	δ^2-Cost			Actual Error		
δ	MC	MLMC	iMLMC	MC	MLMC	iMLMC
0.005	0.4036	0.0051	0.0015	5.00e-3	3.20e-3	4.99e-4
0.002	0.4964	0.0050	0.0014	2.00e-3	1.50e-3	7.00e-4
0.001	0.6034	0.0056	0.0017	1.00e-3	9.20e-4	4.00e-4
0.0005	0.5993	0.0062	0.0020	5.00e-4	2.45e-4	1.90e-4
0.0002	0.5907	0.0114	0.0036	2.00e-4	1.33e-4	1.03e-4

5. CONCLUSIONS

Solving elliptic PDEs with random coefficients is a challenging problem that arises in many applications physics and engineering sciences. In the literatures, the classical MLMC methods have been discussed to improve the computational efficiency of solving this type of problems by computing the correction through multilevel grids to reduce the variance. In

FIGURE 11. 2D problem. Left: Number of samples N_ℓ on each level in the improved MLMC method for different RMSEs δ; Right: Cost scaled by δ^2 of three methods in the case of a log-normal field with two covariance functions for $\lambda = 0.3, \sigma^2 = 1$ when $\mathcal{Q}(u) = |u|_{H^1(D)}$.

all_2D_H1



(a) Gaussian covaraince function (b) Exponential covaraince function

TABLE 9. 2D problem. δ^2-Cost and actual error to achieve the accuracy δ for three methods in the case of a log-normal field with two covariance functions for $\lambda = 0.3, \sigma^2 = 1$ when $Q(u) = |u|_{H^1(D)}$.

Cost_2D_H1

	Gaussian covariance function						Exponential covariance function					
	δ^2-Cost			Actual Error			δ^2-Cost			Actual Error		
δ	MC	MLMC	iMLMC	MC	MLMC	iMLMC	MC	MLMC	iMLMC	MC	MLMC	iMLMC
0.02	20.5640	0.1724	0.0438	2.00e-2	4.2e-3	3.00e-4	15.7402	0.1960	0.0437	1.90e-2	1.53e-2	5.30e-3
0.01	20.1233	0.1815	0.0531	1.00e-2	3.20e-3	3.50e-3	16.5474	0.2041	0.0512	1.00e-2	4.40e-3	4.50e-3
0.005	22.5470	0.1753	0.0404	5.00e-3	4.50e-3	4.00e-4	17.2536	0.1871	0.0506	5.00e-3	4.80e-3	4.50e-3
0.002	20.3584	0.1827	0.0454	2.00e-3	1.80e-3	1.80e-3	17.0074	0.3132	0.0507	2.00e-3	1.30e-3	1.10e-3
0.001	20.6859	0.2328	0.0904	1.00e-3	5.01e-4	4.40e-4	17.1396	0.3898	0.1007	9.82e-4	9.88e-4	7.00e-4

this paper, we proposed an improved MLMC method to further reduce the computational complexity of the classical MLMC method. Under mild conditions, we proves that our improved MLMC method is optimized on the basis of the classical MLMC method, which allows us to achieve the enhancements of computational efficiency. We also presented numerical examples for both 1D and 2D elliptic PDEs with random coefficients to demonstrate the accuracy and efficiency of the proposed method. There are two directions we want to explore in our future work. On one hand, we intend to apply the improved MLMC method to solve other stochastic PDEs arising from uncertainty quantification, such as the Helmholtz equation with random media and time-dependent stochastic PDEs. On the other hand, we will investigate some techniques, such as the sparse matrix method [11, 12, 13, 14] to further reduce the computational time in the implementation of the proposed method.

Acknowledgement

J. Chen is supported by NSFC grant 11971021. R. Du was supported by NSFC grant 11501399. The research of Z. Zhang is supported by Hong Kong RGC grant (Projects 17300318 and 17307921), National Natural Science Foundation of China (Project 12171406), and Seed Funding Programme for Basic Research (HKU).

References

1. M. B. Giles, Multilevel Monte Carlo path simulation. Oper. Res. 256, 981-986(2008).
2. M. B. Giles, Multilevel Monte Carlo methods. Acta Numerica(2015), pp. 259-328.
3. A. L. Teckentrup, R. Scheichl, M. B. Giles, and E. Ullmann, Further analysis of multilevel Monte Carlo methods for elliptic PDEs with random coefficients, Numer. Math. 125(2013), 569-600.
4. M. B. Giles, Multilevel Monte Carlo methods, Monte Carlo and Quasi-Monte Carlo Methods, Springer, 2012, pp. 83-103.
5. J. Charrier, R. Scheichl, and A. L. Teckentrup, Finite element error analysis of elliptic PDEs with random coefficients and its application to multilevel Monte Carlo methods, SIAM J. Numer. Anal. 51 (2013), 322-352.
6. I. G. Graham, F. Y. Kuo, D. Nuyens, R. Scheichl, I. H. Sloan, Quasi-Monte Carlo methods for elliptic PDEs with random coefficients and applications, J. Comput. Phys., 230(2011), pp. 3668-3694.

7. K. A. Cliffe, M. B. Giles, R. Scheichl, and A. L. Teckentrup, Multilevel Monte Carlo methods and applications to elliptic PDEs with random coefficients, Comput. Visual Sci. 14(2011), 3-15.
8. C. R. Dietrich, G. N. Newsam, Efficient generation of conditional simulations by chebyshev matrix polynomial approximations to the symmetric square root of the covariance matrix. Math. Geol. 27(2), 207-228(1995).
9. C. R. Dietrich, G. N. Newsam, Fast and exact simulation of stationary Gaussian processes through circulant embedding of the covariance matrix, SIAM J. Sci. Comput,. 18(1997), pp. 1088-1107.
10. S. Barnett, Matrices, Methods and Applications, in Oxford Applied Mathematics and Computing Sciences Series, Clarendon Press, Oxford, 1990.
11. J. Alberty, C. Carstensen, and S. A. Funken. Remarks around 50 lines of Matlab: short finite element implementation. Numerical Algorithms, 20:117-137, 1999.
12. T. Davis, Creating sparse finite element matrices in MATLAB. http://blogs.mathworks.com/loren/2007/03/01/creating sparse finite element matrices in matlab/, 2007.
13. D. Dunavant. High degree efficient symmetrical Gaussian quadratrue rules for the triangle. Internat. J. Numer. Methods Engrg., 21(6):1129-1148, 1985
14. J. R. Gilbert, C. Moler, and R. Schreiber. Sparse matrices in MATLAB: design and implementation. SIAM J. Matrix Anal. Appl., 13(1):333-356, 1992
15. A. Barth, C. Zollinger, N. Multi-level Monte Carlo finite element method for elliptic PDE's with stochastic coefficients. Numer. Math. Online First(2011)
16. J. Charrier, R. Scheichl, A. L. Teckentrup. Finite element error analysis of elliptic PDEs with random coefficients and its application to multilevel Monte Carlo methods. Tech. rep. University of Bath (2011). BICS Preprint 02/11
17. Ghanem, Roger G., and Pol D. Spanos. Stochastic finite elements: a spectral approach. Courier Corporation, 2003.
18. Xiu, D. and Karniadakis, G.E., 2003. Modeling uncertainty in flow simulations via generalized polynomial chaos. Journal of computational physics, 187(1), pp.137-167.

19. Babuska, I., Tempone, R. and Zouraris, G.E., 2004. Galerkin finite element approximations of stochastic elliptic partial differential equations. SIAM Journal on Numerical Analysis, 42(2), pp.800-825.

20. Hou, T.Y., Luo, W., Rozovskii, B. and Zhou, H.M., 2006. Wiener chaos expansions and numerical solutions of randomly forced equations of fluid mechanics. Journal of computational physics, 216(2), pp.687-706.

21. Najm, H.N., 2009. Uncertainty quantification and polynomial chaos techniques in computational fluid dynamics. Annual review of fluid mechanics, 41, pp.35-52.

22. Cheng, M., Hou, T.Y. and Zhang, Z., 2013. A dynamically bi-orthogonal method for time-dependent stochastic partial differential equations I: Derivation and algorithms. Journal of Computational Physics, 242, pp.843-868.

23. Doostan, A. and Owhadi, H., 2011. A non-adapted sparse approximation of PDEs with stochastic inputs. Journal of Computational Physics, 230(8), pp.3015-3034.

24. Bieri, M., Andreev, R. and Schwab, C., 2010. Sparse tensor discretization of elliptic SPDEs. SIAM Journal on Scientific Computing, 31(6), pp.4281-4304.

25. R. Ghanem, D. Higdon and H. Owhadi, Springer Handbook of Uncertainty Quantification, 2016.

Mathematical center for interdisciplinary research and School of Mathematical Sciences, Soochow University, Suzhou, 215006, China; School of Mathematical Sciences, University of Science and Technology of China, Hefei, Anhui 230026, China; Suzhou Institute for Advanced Research, University of Science and Technology of China, Suzhou, Jiangsu 215123, China

Email address: jingrunchen@ustc.edu.cn

Mathematical center for interdisciplinary research and School of Mathematical Sciences, Soochow University, Suzhou, 215006, China

Email address: durui@suda.edu.cn

Mathematical center for interdisciplinary research and School of Mathematical Sciences, Soochow University, Suzhou, 215006, China

Email address: 15848497244@163.com

School of Mathematics, Sun Yat-Sen University, Guangzhou, 510275, China

Email address: linling27@mail.sysu.edu.cn

Department of Mathematics, The University of Hong Kong, Pokfulam, Hong Kong SAR, China

Email address: zhangzw@hku.hk

School of Data Science and Department of Mathematics, City University of Hong Kong, Tat Chee Ave, Kowloon, Hong Kong SAR, China

Email address: xiang.zhou@cityu.edu.hk