Multilevel Monte Carlo for Stochastic PDEs

Inti-Raymi Carhuancho Mantripp
Mansfield College
University of Oxford

A dissertation submitted for the degree of
MSc in Mathematical Modelling & Scientific Computing
Trinity 2025

Contents

List of Figures

0.1 Introduction

The purpose of this dissertation is to investigate the application of the Multilevel Monte Carlo (MLMC) method as a technique for reducing computational cost in estimating expectations of random variables for specific classes of stochastic partial differential equations (SPDEs).

The central question is:

> *By how much can MLMC reduce the computational cost (wall-clock time) needed to achieve a target root-mean-square error (RMSE) compared with the standard Monte Carlo (MC) estimator built on the same numerical scheme?*

We begin this dissertation by motivating this investigation of alternatives to the standard Monte Carlo method when solving SPDEs, before continuing on to outline more specifically the goals and structure of this dissertation.

Monte Carlo Estimation

Given a random quantity P, the standard Monte Carlo estimator of its expectation $\mathbb{E}[P]$ is simply the average of N independent samples of P:

$$\mathbb{E}[P] \approx \hat{P}_{MC} = \frac{1}{N} \sum_{n=1}^{N} P^{(n)}.$$

Although conceptually simple, the efficiency and feasibility of the Monte Carlo estimator depends heavily on the cost of obtaining a single sample. For straightforward numerical integration, Monte Carlo sampling is inexpensive. To estimate an integral over the unit cube, such as

$$\int_{[0,1]^d} f(x)\, dx = \mathbb{E}[f(X)], \quad X \sim \text{Uniform}(0,1]^d,$$

each sample requires only evaluating the function f at a randomly drawn point $X^{(n)}$, incurring an $O(1)$ computational cost per sample.

However, for stochastic systems described by differential equations, obtaining a sample can be significantly more costly. Consider the case where $P = f(X_T)$, with X_t governed by a stochastic differential equation (SDE):

$$\mathrm{d}X_t = \mu(X_t)\,\mathrm{d}t + \sigma(X_t)\,\mathrm{d}W_t, \quad X_0 = x_0$$

where dW is standard Brownian motion. Obtaining samples is typically approached by discretising the interval $[0, T]$ into small increments of length Δt and generating a sequence of approximations $\{X_{t_0}, X_{t_1}, \ldots, X_{t_N}\}$, with $t_n = n\Delta t$. A common method for doing this is the Euler-Maruyama scheme:

$$X_{t_{n+1}} = X_{t_n} + \mu(X_{t_n})\Delta t + \sigma(X_{t_n})\Delta W_{t_n}, \quad n = 0, 1, \ldots, N-1, \tag{1}$$

with increments $\Delta W_{t_n} \sim \mathcal{N}(0, \Delta t)$. As samples are generated by discrete approximations, there is also error introduced through losing a continuous approximations. Thus, we require our increments Δt to be sufficiently small (fine). Generating each sample thus requires stepping sequentially through a one-dimensional time grid. As the computational cost per sample scales linearly with the number of time steps, the cost per sample grows as $O(\Delta t^{-1})$.

Now consider if P is a functional of the quantity $u(x, t)$ which is governed by a stochastic partial differential equation (SPDE), a spatially extended analogue of an SDE, with stochastic forcing distributed over space and/or time. SPDEs will be discussed in more detail in Section SECTION HERE. It is sufficient to state now though that in these problems, the unknown variable $u(x, t)$ is a random field evolving over both spatial and possibly temporal domains. In a manner analogous to the Euler-Maruyama scheme for SDEs, the numerical solution of SPDEs is typically done using finite-difference or finite-element methods. This means that each Monte Carlo sample corresponds to solving a high-dimensional system of equations repeatedly at each time step. Consequently, the computational cost per sample typically scales as $O(h^{-\gamma})$, where h denotes the spatial discretisation scale and $\gamma \geq d$ with d the spatial dimension. Achieving accurate numerical solutions requires fine discretisation, driving the computational cost dramatically upward.

Suffice to say, the cost for obtaining a single sample of P when dealing with SPDEs is typically very high, making the standard Monte Carlo estimator an unattractive option when estimating expectations of random variables arising from SPDEs.

Compounding this issue further, the standard Monte Carlo estimator's convergence is inherently slow. The variance of the MC estimator is given by

$$\mathbb{V}[\hat{P}_{MC}] = \frac{\mathbb{V}[P]}{N}.$$

This implies the standard error decreases only as $O(N^{-1/2})$. Hence, achieving an accuracy in $\hat{P}_{MC}$ of ε requires $N = O(\varepsilon^{-2})$ samples. The situation for SPDEs therefore

is that each sample can be expensive to obtain, and achieving a high accuracy (e.g. $\varepsilon = 10^{-5}$) requires a *substantially* large number of samples.

Variance Reduction Techniques

The above motivates the use of *variance reduction techniques*, methods designed to achieve a faster variance decay for an equivalent cost. A classical example is the method of control variates, in which an auxiliary random quantity C with known expectation $\mathbb{E}[C]$ is introduced. A modified Monte Carlo estimator is then defined as

$$\hat{P}_{CV} = \frac{1}{N}\sum_{n=1}^{N}\left[P^{(n)} - \beta(C^{(n)} - \mathbb{E}[C])\right].$$

where β is a free parameter. The variance of this estimator is

$$\begin{aligned}\mathbb{V}[\hat{P}_{CV}] &= \frac{1}{N}\mathbb{V}[P - \beta(C - \mathbb{E}[C])] \\ &= \frac{1}{N}\left(\mathbb{V}[P] + \beta^2\mathbb{V}[C] - 2\beta\mathrm{Cov}(P, C)\right).\end{aligned}$$

This expression is quadratic in β, and we can determine its minimiser β^* by differentiating with respect to β and setting the result to zero. This yields

$$\beta^* = \frac{\mathrm{Cov}(P, C)}{\mathbb{V}[C]}.$$

Therefore, the minimal variance is equal to

$$\mathbb{V}[\hat{P}_{CV}] = \frac{1}{N}\left(1 - \rho_{P,C}^2\right), \quad \text{where } \rho_{P,C} = \frac{\mathrm{Cov(P, C)}}{\sqrt{\mathbb{V}[P]\mathbb{V}[C]}}$$

is the correlation between P and C. The purpose of this derivation is to demonstrate the control variates mechanism for achieving variance reduction: **through introducing correlated auxiliary random variables with the target variable P, we can reduce the overall variance of our estimator compared to the standard Monte Carlo estimator $\hat{P}_{MC}$ and thus achieve an equally accurate estimate at a reduced cost**.

How is this correlation achieved? Typically by using the same underlying random noises for a pair of $P^{(n)}$ and $C^{(n)}$ samples. Such samples are then said to be *coupled*.

We emphasise this point regarding how correlation leads to variance reduction in the control variates method because it is also central to how the Multilevel Monte

Carlo method achieves variance reduction. Determining how to achieve this correlation forms a key part of this dissertation's investigation in the practical application of MLMC to SPDEs.

Multilevel Monte Carlo and SPDEs

The Multilevel Monte Carlo (MLMC) method extends the idea of control variates. Rather than a single auxiliary variable, MLMC construct a hierarchy of increasingly accurate (but more costly) estimators $P_0, P_1, \ldots, P_\ell, \ldots, P_L$, each associated with different discretisation parameters. In the context of SDEs, this corresponds to finer and finer time intervals Δt. In that of SPDEs, larger ℓ corresponds to finer meshes. Instead of directly estimating the closest approximation P_L using the standard Monte Carlo estimator, MLMC exploits the following telescoping sum:

$$\mathbb{E}[P_L] = \mathbb{E}[P_0] + \sum_{\ell=1}^{L} \mathbb{E}[P_\ell - P_{\ell-1}] \tag{2}$$

where each $\mathbb{E}[P_\ell - P_{\ell-1}]$ is described as a *level*. Equation (2) can be thought of as an initial poor estimate obtained using the coarsest estimator P_0 which is then refined by subsequent levels obtained using estimators of greater accuracy.

The MLMC estimator is then multiple Monte Carlo estimates of each level in (2) (hence *Multilevel* Monte Carlo):

$$\hat{P}_{MLMC} = \frac{1}{N_0} \sum_{n=1}^{N_0} P_0^{(n)} + \sum_{\ell=1}^{L} \frac{1}{N_\ell} \sum_{n=1}^{N_\ell} \left(P_\ell^{(n,\ell)} - P_{\ell-1}^{(n,\ell)} \right) \tag{3}$$

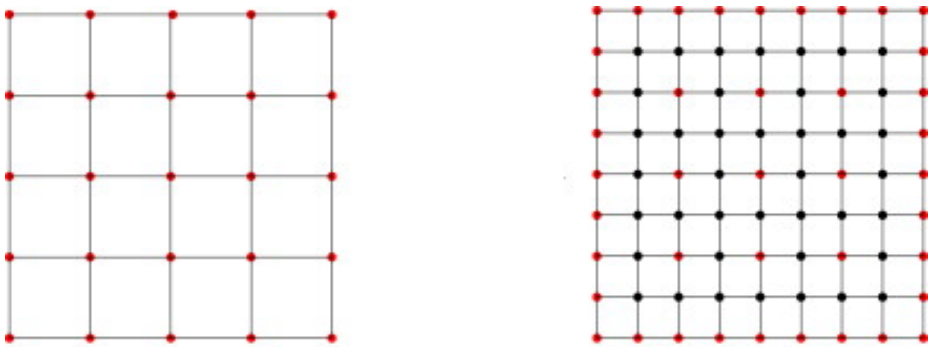
where ℓ in the superscript (ℓ, n) serves to indicate that independent samples are used at each ℓ. With a single level this becomes reminiscent of the control variate estimator with $\beta = 1$, except that $\mathbb{E}[P_0]$ is not known.

As with control variates, by ensuring each pair $P_\ell^{(n,\ell)}, P_{\ell-1}^{(n,\ell)}$ use the same random noises we can achieve strong correlation and thus variance reduction. As quantities P_ℓ where ℓ is small correspond to coarser discretisation, they are cheap to compute, while larger ℓ samples are more accurate but more expensive. MLMC, by initially forming a coarse estimate and then refining it, aims to achieve a target accuracy ε in estimator $\hat{P}_{MLMC}$ at a lower cost than the standard Monte Carlo estimator $\hat{P}_{MC}$. Further details of the MLMC method will be given in section SECTION HERE.

While theoretical advantages of MLMC exist for a variety of stochastic systems, their effectiveness for parabolic SPDEs remains an open question. A literature review of the current landscape is provided in section SECTION HERE. This question of to

what extent MLMC can reduce cost for parabolic SPDEs is the central question of this dissertation.

As has hopefully been demonstrated though, this question is in part dependent on the extent to which we couple samples in the same level of the MLMC estimator, (3). What does this mean in practice? Figure 1 shows an illustrative finite difference coarse grid and a fine grid used to obtain samples of $P_{\ell-1}$ and P_ℓ respectively. At each vertex of both grids, a random noise is required to generate a sample. What approaches can we use to ensure that these random noises are correlated? This forms a key part in the investigation of this dissertation.



Coarse Grid Fine Grid

Figure 1: An illustrative coarse and fine finite difference grid used to obtain samples of $P_{\ell-1}$ and P_ℓ respectively. Red dots indicate vertices of the coarse grid. These are overlayed on the fine grid also, indicating common points both grids share and that both would evaluate. Black dots indicate vertices of the fine grid that are not common to the coarse grid.

Dissertation Structure and Goals

Having motivated the investigation of MLMC for SPDEs and illustrating what this entails, we state the goals of this dissertation.

- **To investigate and quantify the cost reductions achievable with MLMC for parabolic SPDEs.** This will be done by implementing MLMC across several classes of parabolic SPDEs and comparing costs to the baseline method, chosen to be the standard Monte Carlo estimator.
- **Investigate different coupling strategies for parabolic SPDEs.** This goal is heavily related to the first, as the extent to which coupling is achieved is a determinant in the cost reduction achievable. We will propose and investigate several coupling strategies.

- **Assess further performance improvements achievable through high-performance implementations**. Monte Carlo methods are described as "embarrassingly parallel", lending themselves to implementations on graphics processing units (GPUs). Having assessed the cost reduction of MLMC compared to baseline Monte Carlo methods, we will determine what performance improvements are realisable when taking advantage of the highly parallelisable nature of Monte Carlo methods.

The structure of this dissertation will be as follows. In the next section we will give a theoretical background relevant topics in this dissertation. This will cover the main theory of MLMC, a sufficiently relevant overview of SPDEs and a literature review of the current landscape.

Section SECTION HERE will then present an MLMC implementation for the stochastic heat equation. We validate this implementation through a series of methods, varying the quantity of interest being derived, that our MLMC implementation converges to this and that its rate of convergence is appropriate. We then further validate our MLMC implementation of the Dean-Kawasaki equation, replicating the results of the original paper and our implementation.

Section SECTION HERE then presents results for timings. These are

Section SECTION HERE will present the high performance implementations.

We finally conclude, outlining further avenues of investigation and telling everyone great we are.

0.2 Prerequisites

In this section we will outline the relevant prerequisites required to justify and present the investigations performed in this dissertation. These encompass outlining the Multilevel Monte Carlo (MLMC) method in further depth than that given in Section 0.1, presenting the SPDEs investigated and the finite difference schemes derived for them, and a review of the relevant literature. [1]

0.2.1 Multilevel Monte Carlo method

Let P be a random variable defined implicitly through the solution of a stochastic differential equation, and let P_L denote its numerical approximation obtained using a discretisation of level L, such that $\mathbb{E}[P] \approx \mathbb{E}[P_L]$. We define additional estimators P_ℓ for less accurate levels $\ell \in \{0, 1, ..., L\}$. For example, in later finite difference schemes we will use spatial step size $h_\ell = 2^{-(\ell+1)}$.

We express $\mathbb{E}[P_L]$ via the following telescopic sum

$$\mathbb{E}[P_L] = \sum_{\ell=0}^{L} \mathbb{E}\left[P_\ell - P_{\ell-1}\right] \quad \text{where } P_{-1} \equiv 0.$$

We define the following unbiased estimator for $\mathbb{E}[P_L]$ to be the MLMC estimator:

Definition 1 (MLMC Estimator) *Given $N_\ell \in \mathbb{N}$ samples on each level $\ell = 0, \ldots, L$, the MLMC estimator for $\mathbb{E}[P_L]$ is*

$$\hat{P}_{\text{MLMC}} \;=\; \sum_{\ell=0}^{L} \frac{1}{N_\ell} \sum_{n=1}^{N_\ell} \big(P_\ell^{(n,\ell)} - P_{\ell-1}^{(n,\ell)}\big),$$

with the convention $P_{-1}^{(n,-1)} \equiv 0$ and where $P_\ell^{(n,\ell)}, P_{\ell-1}^{(n,\ell)}$ are coupled coarse/fine samples sharing the same random inputs.

For each level ℓ we also define the following quantities, following the notation used in [1]:

$$Y_\ell := \frac{1}{N_\ell} \sum_{n=1}^{N_\ell} \big(P_\ell^{(n,\ell)} - P_{\ell-1}^{(n,\ell)}\big) \quad V_\ell := \mathbb{V}[P_\ell - P_{\ell-1}]$$

Denoting by C_ℓ the expected cost of a single sample for $P_\ell - P_{\ell-1}$, we obtain an overall cost for the MLMC estimator of $\sum_{\ell=0}^{L} C_\ell N_\ell$ and an overall variance of $\sum_{\ell=0}^{L} N_\ell^{-1} V_\ell$.

We wish to determine N_ℓ that minimises the total cost for a fixed variance ε^2. Using a largange multiplier μ^2 gives $N_\ell = \mu\sqrt{V_\ell/C_\ell}$. Via the variance constraint we obtain $\mu = \varepsilon^{-2}\sum_{\ell=0}^{L}\sqrt{V_\ell C_\ell}$. Thus, the overall cost is

$$C_{MLMC} = \varepsilon^{-2}\Big(\sum_{\ell=0}^{L}\sqrt{V_\ell C_\ell}\Big)^2 \tag{4}$$

The dominant cost is determined by whether the product $V_\ell C_\ell$ increases or decreases with ℓ. If the product increases with ℓ, the dominant cost comes from $V_L C_L$ such that $C_{MLMC} \approx \varepsilon^{-2}V_L C_L$. Conversely, if the product decreases with ℓ then $C_{MLMC} \approx \varepsilon^{-2}V_0 C_0$ [1]. In contrast, the Monte Carlo estimator has total cost $\varepsilon^{-\not{\vdash}}\mathbb{V}_{\not\vdash}\mathbb{C}_{\mathbb{L}}$

We also define the following unbiased Monte Carlo estimator for $\mathbb{E}[P_L]$ also

Definition 2 (MC Estimator) *Given $N_L \in \mathbb{N}$ independent samples obtained at level L, the MC estimator for $\mathbb{E}[P_L]$ is*

$$\hat{P}_{MC} = \frac{1}{N_L}\sum_{n=1}^{N_L} P_L^{(n)}$$

The cost of the MC estimator therefore is $N_L C_L$

Bibliography

[1] Michael B Giles. Multilevel monte carlo methods. *Acta numerica*, 24:259–328, 2015.