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

Chapter 1

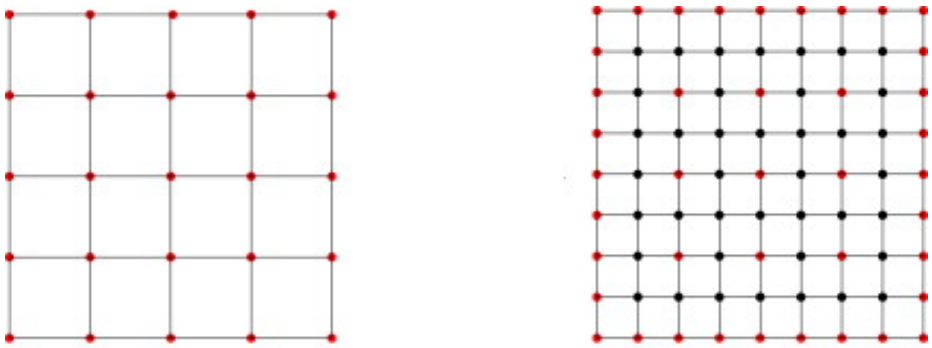
Introduction

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

The motivation for this stems from the fact that the standard Monte Carlo (MC) estimator suffers from slow convergence. Its root mean square error decays $O(N^{-1/2})$ where N is the number of independent samples obtained, and therefore achieving a high accuracy estimate requires a high number of samples. As most SPDEs do not admit analytic solutions, the main approach available for estimating expectations of functionals of SPDEs is Monte Carlo methods. In the context of SPDEs, obtaining samples typically requires discretising the spatial domain via finite difference or finite element methods and then evolving the system to obtain the desired quantity. Accurate estimates require fine discretisations, and as such, the cost of obtaining a sample can become expensive. This, combined with the slow convergence of the standard MC estimator, can make the standard Monte Carlo method prohibitively expensive.

A number of techniques exist that aim to improve on the convergence of the MC estimator. Their goal is to achieve an equivalent variance in their estimate at a reduced computational cost. The MLMC method is one such method. At its heart is an approach of computing a blend of cheaper and more expensive samples, unlike the MC estimator which uses only samples of the same cost. Many inexpensive samples provide the bulk of the statistical signal, while comparitively few more costly samples then correct any residual bias. In the context of SPDEs, this means obtaining samples using meshes at varying *levels* of discretisation. By taking many samples at cheaper, coarser levels and fewer at finer, more expensive levels, the MLMC estimator aims to achieve an estimate of equivalent accuracy at reduced cost compared to the MC estimator.

The practical efficiency of the MLMC method however relies on correlating samples obtained at different levels, in a manner analogous to that of the control variates method [10]. This is because the cost savings achievable is dependent on the extent to which samples across two adjacent levels of accuracy covary. Practically, this entails aligning the randomness used to generate pairs of coarse and fine samples, a process referred to as *coupling*. To provide intuition as to what this means, Figure 1.1 illustrates what the finite difference grid used by two adjacent estimators might look like. At each node in these grids, a random noise is required to be generated. This raises a question: what methods are best suited to aligning the randomness across samples? Further, how large an impact can different coupling methods have?



Coarse Grid Fine Grid

Figure 1.1: 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.

There are three main questions this dissertation aims to answer therefore.

1. **What cost savings can MLMC achieve over standard MC in the context of two parabolic SPDEs?**

2. **What are the effects of different coupling mechanisms?**

3. **Monte Carlo methods are embarrasingly parallelisable. What further cost reductions are therefore achievable in high performance implementations of these methods if we take advantage of the highly parallel nature of MC methods?**

To answer these questions, this dissertation is structured as follows.

Chapter 2 provides the theoretical background and a review of the relevant literature. We begin by formally defining the standard MoION HERE To best present the answers found to these questions, this dissertation is structured as so. In the next section we formally present the requisite topics as they pertain to this investigation. This will cover the main theory of MLMC, a sufficiently relevant overview of SPDEs and a literature review of the current landscape.

SECTION HERE details our numerical implementation. We describe the finite difference schemes used to discretise the SPDEs and, crucially, the specific coupling strategies employed to correlate samples between levels. We then validate our implementation, demonstrating that it achieves the correct theoretical convergence rates for both the stochastic heat equation and the Dean-Kawasaki equation.

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 presents a comprehensive performance analysis. We directly address our first two research questions by comparing the computational cost of our MLMC implementation against a standard MC estimator for a range of target accuracies. Furthermore, we analyse the practical impact of different coupling mechanisms on overall efficiency.

Section SECTION HERE explores the potential for further cost reductions through high-performance computing. We discuss the design of a parallelised MLMC algorithm that leverages modern multi-core architectures and present performance metrics, such as computational speedup, to quantify the benefits of this approach.

Finally, Section SECTION HERE Finally, Chapter 6 concludes the dissertation. We summarise our key findings, revisit the initial research questions, and discuss potential avenues for future investigation.

Chapter 2

Preliminary Concepts and Methods

This section presents the essential thoeretical groundwork upon which the investigations of this dissertation is built. First, we present and develop the theory of the Monte Carlo (MC) and Multilevel Monte Carlo (MLMC) methods, analysing their respective convergence and computational complexities.

Second, we provide a concise introduction to the theory of stochastic partial differential equations (SPDEs). We focus on two parabolic equations which serve as our case studies: the Stochastic Heat Equation and the Dean-Kawasaki equation. We outline their mathematical formulations and briefly discuss the challenges associated with their numerical solutions, this being particularly pertinent to the Dean-Kawasaki equation.

Finally, we conclude the section with a literature rveiew. This review surveys the development and application of the MLMC method within the context of SPDEs, thereby situating the specific questions of this dissertation within the broader research performed in this area.

2.1 Monte Carlo and Multilevel Monte Carlo

The primary objective in many applications involving SPDEs is not to find a single, particular solution, but rather to compute the expected value of a quantity of interest that depends on the solution. For example, we may want to find the average temporature at a specific point in a domain governed by the stochastic heat equation, or the average particle density in a system described by the Dean-Kawasaki equation.

Let P represent such a quantity of interest, which we assume to be a real-valued random variable. Our goal throughout this section is to develop an efficient numerical

method for estimating its expectation, $\mathbb{E}[P]$. Since analytical expressions for $\mathbb{E}[P]$ are rarely available in this context, we must turn to computational methods. The most fundamental of these is defined below.

Definition 1 (Monte Carlo Estimator) *Let $\{P^{(n)}\}$ for $n = 1, \ldots, N$ be a set of N independent and identically distributed samples of a random variable P. The standard Monte Carlo estimator, $\hat{P}_{MC}$, of the expectation $\mathbb{E}[P]$ is the sample mean:*

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

By linearity of the expectation operator, the MC estimator in Definition 1 is unbiased. The accuracy of an estimator is typically estimated via its Mean Squared Error (MSE) and Root Mean Square Error (RMSE), which we define now.

Definition 2 (Mean Squared Error and Root Mean Squared Error) *Let P be a fixed unknown quantity and let $\hat{P}$ be an estimator for P. The **Mean Square Error (MSE)** of the estimator is the expected value of the squared error:*

$$MSE(\hat{P}) = \mathbb{E}[(\hat{P} - P)^2]$$

This can be decomposed into terms representing the estimator's variance and squared bias:

$$MSE(\hat{P}) = \underbrace{\mathbb{V}[\hat{P}]}_{Variance\ of\ estimator} + \underbrace{(\mathbb{E}[\hat{P}] - \mathbb{E}[P])^2}_{Bias^2}$$

*The **Root Mean Square Error (RMSE)** is the square root of the MSE.*

Since the standard MC estimator is unbiased, its bias term is zero. Its MSE is therefore equal to its variance:

$$\text{MSE}(\hat{P}_{MC}) = \mathbb{V}[\hat{P}_{MC}] = \frac{1}{N}\mathbb{V}[P]$$

The framework above assumes we can generate perfect samples of the random variable P. In practice, for complex problems such as SPDEs this is typically impossible. Instead, we must compute numerical approximations which we will denote by P_L, where L represents a level of discretisation. For example, and as will be used later, in a finite difference scheme L could correspond to a mesh with spatial grid spacing $h_L = 2^{-(L+1)}$. A higher L means a finer mesh and a more accurate - but also more computationally expensive - approximation.

This introduces a second source of error. The total error of our estimate is now a combination of the statistical error from the Monte Carlo sampling and the systematic bias from the numerical discretisation. In this context, the MSE of the standard MC estimator which uses N samples of the numerical approximation P_L, which we denote $\hat{P}_{MC,L}$ is therefore the following:

$$\text{MSE}(\hat{P}_{MC,L}) = \underbrace{\frac{1}{N}\mathbb{V}[P_L]}_{\substack{\text{Statistical Error} \\ \text{(Variance)}}} + \underbrace{(\mathbb{E}[P_L] - \mathbb{E}[P])^2}_{\substack{\text{Discretisation Error} \\ \text{(Bias}^2\text{)}}} \tag{2.1}$$

This explicit formula presents the clear dilemma of the MC estimator. To achieve an overall MSE less than ε^2, one must sufficiently reduce the above terms. To make the statistical error small, we require a large number of samples N. To make the discretisation error small, we require a fine discretisation level L such that $P_L \approx P$. A finer level L dramatically increases the computational cost of generating each sample, which we denote by C_L. The total cost of the MC estimator is $N \times C_L$. Thus, we have that to achieve a target MSE, both terms in this product increase.

This is the fundamental challenge that the Multilevel Monte Carlo method is designed to overcome. Instead of estimating the expensive quantity $\mathbb{E}[P_L]$ directly, MLMC rewrites it using a telescoping sum:

$$\mathbb{E}[P_L] = \sum_{\ell=0}^{L} \mathbb{E}[Y_\ell], \quad \text{where} \quad Y_\ell := P_\ell - P_{\ell-1} \quad \text{and} \quad P_{-1} := 0.$$

Each term $\mathbb{E}[Y_\ell]$ is then estimated independently with a standard Monte Carlo estimator.

Definition 3 (Multilevel Monte Carlo Estimator) *Let $Y_\ell = P_\ell - P_{\ell-1}$ be the correction at level ℓ. The **Multilevel Monte Carlo (MLMC) estimator**, $\hat{P}_{\text{MLMC}}$, for the expectation $\mathbb{E}[P_L]$ is:*

$$\hat{P}_{\text{MLMC}} = \sum_{\ell=0}^{L} \hat{Y}_\ell, \quad \textit{where} \quad \hat{Y}_\ell = \frac{1}{N_\ell} \sum_{n=1}^{N_\ell} Y_\ell^{(n)}.$$

By the linearity of expectation, $\hat{P}_{\text{MLMC}}$ is an unbiased estimator for $\mathbb{E}[P_L]$. Since the estimates at each level are independent, its variance is the sum of the individual variances. We define the cost and variance of a single sample at level ℓ as C_ℓ and V_ℓ respectively:

$$
\begin{aligned}
C_\ell &:= \text{Cost}(Y_\ell) \\
V_\ell &:= \mathbb{V}[Y_\ell] = \mathbb{V}[P_\ell - P_{\ell-1}] \\
&= \mathbb{V}[P_\ell] + \mathbb{V}[P_{\ell-1}] - 2\text{Cov}(P_\ell, P_{\ell-1}).
\end{aligned}
$$

The total cost and variance of the MLMC estimator are therefore:

$$
C_{\text{MLMC}} = \sum_{\ell=0}^{L} N_\ell C_\ell, \qquad \mathbb{V}[\hat{P}_{\text{MLMC}}] = \sum_{\ell=0}^{L} \frac{V_\ell}{N_\ell}.
$$

The success of the MLMC method hinges on the behaviour of the level variances, V_ℓ. The key is to ensure that V_ℓ is small, especially for large ℓ where the cost C_ℓ is high. This is achieved by using the same underlying source of randomness to generate the pair $(P_\ell, P_{\ell-1})$. This technique is known as *coupling*. For example, when solving a Stochastic Differential Equation driven by Brownian motion, both the fine (P_ℓ) and coarse $(P_{\ell-1})$ simulations use the same discretised Brownian path ([10], section 5.1 for example).

Because the coarse and fine paths are strongly correlated, functionals P_ℓ and $P_{\ell-1}$ are also strongly correlated. Consequently, the variance of their difference, V_ℓ, is much smaller than the variance of either term individually. As the level ℓ increases, P_ℓ converges to $P_{\ell-1}$, and so we expect that $V_\ell \to 0$.

This property allows for the following trade-off: we can use a small number of samples N_ℓ for the expensive, high-level correction terms (where V_ℓ is small) and compensate by using a large number of samples for the cheap, low-level terms. We can determine the optimal number of samples for each N_ℓ that minimises the total cost in order to achieve a given variance and also ensure we use enough levels such that our discretisation error is also sufficiently small. Both of these ensure that we can achieve a target MSE or RMSE at minimal cost.

For a fixed variance, ε^2, choosing the optimal $\{N_\ell\}_{\ell=0}^{L}$ that minimises the total cost $C_{\text{MLMC}} = \sum N_\ell C_\ell$ is a classic constrained optimisaton problem solveable with Lagrange mutipliers. It yieds optimal number of samples [10]:

$$
N_\ell = \left\lceil \frac{1}{\varepsilon^2_{\text{var}}} \sqrt{\frac{V_\ell}{C_\ell}} \sum_{k=0}^{L} \sqrt{V_k C_k} \right\rceil. \tag{2.2}
$$

The ceiling function $\lceil \cdot \rceil$ ensures the number of samples is an integer. Equation (2.2) states that we should take more samples when the variance per unit cost of a level is high, and less when it is low.

We also have that the bias $(\mathbb{E}[P_\ell] - \mathbb{E}[P]) \to 0$ as $\ell \to \infty$. To ensure that MSE is less than ε^2, by Definition 2 we can impose that $(\mathbb{E}[P_L - P])^2 < \frac{\varepsilon^2}{2}$ and $\mathbb{V}[\hat{P}_{\text{MLMC}}] < \frac{\varepsilon^2}{2}]$.

This leads to the following theorem [10] which makes precise the cost scaling of the MLMC method:

Theorem 1 (MLMC Complexity Theorem) *Let P denote a random variable, and let P_ℓ denote the corresponding level ℓ numerical approximation.*

If there exists independent estimators Y_ℓ based on N_ℓ Monte Carlo samples, each with expected cost C_ℓ and variance V_ℓ, and positive constants $\alpha, \beta, \gamma, c_1, c_2, c_3$ such that $\alpha \geq \frac{1}{2}\min(\beta, \gamma)$ and we have

1. ***Weak Error (Bias) Decay:*** $|\mathbb{E}[P_\ell - P]| \leq c_1 2^{-\alpha\ell}$,
2. ***Unbiased Estimators:*** $\mathbb{E}[Y_\ell] = \begin{cases} E[P_0], & l = 0 \\ E[P_l - P_{l-1}], & l > 0, \end{cases}$
3. ***Variance Decay:*** $V_\ell \leq c_2 2^{-\beta\ell}$,
4. ***Cost Growth:*** $C_\ell \leq c_3 2^{\gamma\ell}$,

then there exists a positive constant c_4 such that for any $\varepsilon < e^{-1}$ there are values L and N_L for which the multilevel estimator

$$Y = \sum_{\ell=0}^{L} Y_\ell,$$

has an MSE with bound

$$MSE \equiv \mathbb{E}\left[(Y - \mathbb{E}[P])^2\right] < \varepsilon^2 \tag{2.3}$$

with a computational complexity C with bound

$$\mathbb{E}[C] \leq \begin{cases} c_4\varepsilon^{-2}, & \beta > \gamma, \\ c_4\varepsilon^{-2}(\log\varepsilon)^2, & \beta = \gamma, \\ c_4\varepsilon^{-2-(\gamma-\beta)/\alpha},] & \beta < \gamma \end{cases} \tag{2.4}$$

To appreciate the significance of the MLMC Complexity Theorem, we first establish the cost of the standard MC method in the above context. To achieve an MSE of $\mathcal{O}(\varepsilon^2)$, both the statistical and discretisation errors must be controlled. Controlling the **bias** to $O(\varepsilon)$ requires using a fine grid with a step size $h_L \propto \varepsilon^{1/\alpha}$. Independently, controlling the statistical error to $O(\varepsilon^2)$ requires $N \propto \varepsilon^{-2}$ samples. The total cost

is the product of the number of samples and the cost per sample, where $C_L \propto h_L^{-\gamma}$. Combining these requirements gives the overall complexity:

$$C_{\text{MC}} \propto N \times C_L \propto \varepsilon^{-2} \times (\varepsilon^{1/\alpha})^{-\gamma} = \varepsilon^{-2-\gamma/\alpha}.$$

In contrast, in the MLMC case $\beta > \gamma$, the dominant computational cost is on coarsest level where the cost per sample C_ℓ is $O(1)$ [10]. Requiring $N = O(\varepsilon^{-2})$ samples provides the dominant cost. This is the optimal case.

When $\beta = \gamma$, the cost contribution from each level, $N_\ell C_\ell \propto \sqrt{V_\ell C_\ell}$, is approximately constant across all levels [10]. The total cost is therefore proportional to the number of levels, L, which must increase as $\mathcal{O}(\log \varepsilon)$ to meet the bias requirement. This results in the total complexity of $\mathcal{O}(\varepsilon^{-2}(\log \varepsilon)^2)$, which remains a vast improvement over the standard method.

In the case $\beta < \gamma$, the cost per level grows with ℓ, meaning the total cost is dominated by the work on the finest level, L. Even in this worst case scenario though, we still arrive at a smaller scaling of cost than the MC estimator.

We conclude this section by highlighting the importance of how β and γ relate to one another in determining the magnitude of cost savings an MLMC implementation can offer over an MC estimator. Determining how MLMC performs relative to MC in SPDE applications will depend heavily on what variance decay we can achieve relative to cost growth in our implementations.

2.2 Stochastic Partial Differential Equations

In their most general sense, an SPDE is a partial differential equation where at least one of the following is random: coefficients, initial boundary conditions, the domain, and the forcing term [15]. In our case, the SPDEs we examine have only a random forcing term. In this section, we present the two SPDEs that will be the use cases for our MLMC implementations and investigations. In section SECTION HERE we will derive and define the finite difference schemes based on these implementations.

2.2.1 The Stochastic Heat Equation

The stochastic heat equation (SHE) is a canonical example of a parabolic SPDE. Formally, it is the standard heat equation perturbed by a stochastic forcing term introducing spatially and temporally uncorrelated fluctuations. For instance, heat diffusing through a metal bar that experiences heat emitting chemical reactions is a typical scenario the SHE describes.

The equation is fully specified by the SPDE itself, initial conditions and a set of bondary conditions [15, 17].

$$\begin{aligned} \frac{\partial u(t,\mathbf{x})}{\partial t} &= \Delta u(t,\mathbf{x}) + \xi(t,\mathbf{x}), \quad \text{for } (t,\mathbf{x}) \in (0,T] \times \Omega \\ u(0,\mathbf{x}) &= u_0(\mathbf{x}), \quad \text{for } \mathbf{x} \in \Omega \\ \mathcal{B}u(t,\mathbf{x}) &= g(t,\mathbf{x}), \quad \text{for } (t,\mathbf{x}) \in (0,T] \times \partial D \end{aligned} \tag{SHE}$$

Where $u(t,\mathbf{x})$ is a real valued function at time $t \in [0,T]$ and spatial position $\mathbf{x}$ within domain $\Omega \subset \mathbb{R}^d$. $u_0(\mathbf{x})$ is the initial state of the field, the operator $\mathcal{B}$ represents the boundary condition on the boundary of domain $\partial\mathcal{B}$, and Δ is the standard Laplacian operator.

The defining component is the stochastic forcing term $\xi(t,\mathbf{x})$ term which denotes space-time white noise. We formally define this as follows ([22], Chapter 1).

Definition 4 (Space-Time White Noise) *A space-time white noise $\xi(t,\mathbf{x})$ on $[0,T]\times D$ is a centered Gaussian process defined by a collection of random variables $\{W(\phi)\}$ indexed by test functions $\phi \in L^2([0,T]\times D)$, with a covariance structure given by:*

$$\mathbb{E}[W(\phi)W(\psi)] = \langle \phi, \psi \rangle_{L^2}$$

This formal defintion has several important consequences. Firstly, it gives rise to the more intuitive heuristic covariance expression $\mathbb{E}\left[\xi(t,\mathbf{x})\xi(s,\mathbf{y}\right] = \delta(t-s)\delta(\mathbf{x}-\mathbf{y})$, implying the noise is perfectly uncorrelated at every point. Secondly, the abstract process can be understood as the distributional derivative of a more tangible (though still highly irregular) object known as a Brownian sheet, i.e. a multidimensional Brownian motion.

Most critically for our purposes, Definition 4 directly informs how we discretise the noise term in a numerical scheme. The integral of noise over a discrete space-time grid cell, $C_j^n = [t_n, t_{n+1}] \times [x_j - \frac{\Delta x}{2}, x_j + \frac{\Delta x}{2}]$, is found by choosing the test function ϕ to be the indicator function of that cell, $\phi_j^n(t,\mathbf{x}) = \mathbf{1}_{C_j^n}(t,\mathbf{x})$.

The random variable representing the integrated noise over this cell is therefore $W(\phi_j^n)$. From Definition 4, we know this is a centred Gaussian random variable whose variance is given by:

$$\mathrm{Var}(W(\phi_j^n)) = \mathbb{E}[W(\phi_j^n)^2] = \langle \phi_j^n, \phi_j^n \rangle_{L^2} \tag{2.5a}$$

$$= \int_0^T \int_D (\mathbf{1}_{C_j^n}(t,\mathbf{x}))^2 \mathrm{d}\mathbf{x}\mathrm{d}t \tag{2.5b}$$

$$= \int_{t_n}^{t_{n+1}} \int_{x_j - \frac{\Delta x}{2}}^{x_j + \frac{\Delta x}{2}} 1 \mathrm{d}\mathbf{x}\mathrm{d}t \tag{2.5c}$$

$$= \mathrm{Area}(C_j^n) = \Delta t \Delta x \tag{2.5d}$$

Since any centred Gaussian random variable with variance δ^2 can be written as σZ where $Z \sim \mathcal{N}(0,1)$. Equations (2.5) lead to a result that will be used in our finite difference implementations:

$$\int_{t_n}^{t_{n+1}} \int_{x_j - \frac{\Delta x}{2}}^{x_j + \frac{\Delta x}{2}} \xi(t,\mathbf{x}) \mathrm{d}x\mathrm{d}t = \sqrt{\Delta t \Delta x} Z_j^n \tag{2.6}$$

where Z_j^n are idependent and identically distirbuted standard normal random variables.

2.2.2 The Dean-Kawasaki Equation

- The Dean-Kawasaki (DK) equation is used to describe the evolution of the density $\rho(\mathbf{x},t)$ of a system of $N >> 1$ weakly interacting particles, having emerged from the field of fluctuating hydrodynamics. For the non-interacting case investigated in this dissertation, the equation is given by [6]:

$$\begin{aligned} \frac{\partial \rho(t,\mathbf{x})}{\partial t} &= \frac{1}{2}\Delta\rho(t,\mathbf{x}) + N^{-1/2}\nabla \cdot (\sqrt{\rho(t,\mathbf{x})}\xi(t,\mathbf{x})), && \text{for } (t,\mathbf{x}) \in (0,T] \times \Omega \quad \text{(DK)} \\ \rho(0,\mathbf{x}) &= \rho_0(\mathbf{x}), && \text{for } \mathbf{x} \in \Omega \\ \rho(t,\mathbf{x}) &\text{ satisfies periodic b.c. on } \partial\Omega, && \text{for } t \in (0,T] \end{aligned}$$

Here, $\rho(t,\mathbf{x})$ is the particle density at time t in a domain $\Omega = \mathbb{T}^d$, the d-dimensional torus. The term $\frac{1}{2}\Delta\rho$ describes typical particle diffusion. The stochastic forcing term $N^{-1/2}\nabla \cdot (\sqrt{\rho}\xi)$ captures the particle density flux, where ξ is space-time white noise. Intuitively, this noise scaled by $\sqrt{\rho(t,\mathbf{x})}$.

The equation is highly singular, and "strong" or "pathwise" solutions i.e. solutions that exist as a function of every given realisation of the noise, do not exist. Instead, the only existing solutions are martingale solutions where ρ corresponds to empirical measures of the underlying particle system:

$$\rho(\mathbf{x},t) \equiv \mu_t^N(\mathbf{x}) := N^{-1}\sum_{i=1}^{N}\delta(x - X_i(t)).$$

Cornalba and Fischer [6] do demonstrate however that statistical properties of fluctuations around the mean-field limit $r\bar{h}o$ can be simulated. Generally, quantities of interest, Q, of the form:

$$Q = \psi\big(N^{1/2}\int(\mu_N^T - \bar{\rho}^T)(\mathbf{x})\phi(\mathbf{x})\mathrm{d}\mathbf{x}\big) \tag{2.7}$$

can be simulated for sufficiently regular test functions ψ, ϕ, provided $Nh^d >> 1$ (i.e. as long as on average there is more than one particle per grid cell). For example, choosing $\psi(z) = z^2$ allows for the computation of the variance of the fluctuations.

2.3 Literature Review

The MLMC method was formally introduced and popularised by Giles for SDE path simulation in 2008 [9], building on earlier foundational work on multilevel integration by Heinrich starting in 1998 [13]. Early works focussed predominantly on application to SDEs, particularly in computational finance. Other research extended the method's application to a wider range of topics, including various classes of SDEs [2, 18], Lévy processes [12], Numerical Linear Algebra [3], and Reliability theory [4].

MLMC for SPDEs is a more recent and active area of research. A search on Scopus [8] of "Multilevel Monte Carlo Stochastic Partial Differential Equations" at the time of writing returns 88 documents. Changing this to "Multilevel Monte Carlo Parabolic Stochastic Partial Differential Equations" refines this down to 18. Of the work examining MLMC for SPDEs, much of the focus has been on elliptic SPDEs (for example [1, 14, 16]).

This dissertation is concerned with parabolic SPDEs, for which the literature provides a smaller but highly relevant set of foundational papers. This review will focus on three key works that inform the central questions of this research: establishing the theoretical basis for MLMC's efficiency, demonstraing its practical validation, and exporing its application to highly singular equations with different coupling strategies.

Barth, Lang, and Schwab [5] analyse provide a foundational analysis of the convegrence and complexity of the MLMC method for a general class of parabolic SPDEs. Using a Galerkin method in space and a Euler-Maruyama scheme in time, they prove that the MLMC estimator significantly reduces the computational work required to

achieve a given accuracy compared to a standard single-level method. Their key result shows that the computational complexity can be reduced from $O(h_L^{-(d+4)})$ for a standard MC method to nearly $O(h_L^{-(d+2)})$ for MLMC, where d is the spatial dimension and h_L is the finest mesh width. Their work provides theoretical underpinning for the cost savings this dissertation seeks to validate for the stochastic heat and Dean-Kawasaki equations. However, their analysis does not explore the practical performance improvements of different noise coupling strategies.

In [11], Giles and Reisinger provide a practical demonstration of MLMC's performance improvements for a class of parabolic SPDEs arising in financial modelling. The authors develop and analyse a Milstein finite difference scheme, proving it converges with first-order accuracy in time and second-order in space. They demonstrate a concrete reduction in computational complexity from $O(\varepsilon^{-7/2})$ for a standard MC approach to the optimal $O(\varepsilon^{-2})$ for their MLMC implementation, validating this gain through numerical experiments. This work serves as a methodological benchmar, demonstrating how to empirically confirm the theoretical performance gains of the MLMC method.

A very revent and highly relevant contribution is the 2024 paper by Cornalba and Fischer, which develops and analyses an MLMC method specifically for the Dean-Kawasaki equation, one o the two case studies in this dissertation. Their work tackles a highly singular SPDE for which standard MLMC convergence proofs fail. By formulating their analysis in terms of the convergence of probability distributions, they prove that MLMC provides a significant computational improvement over standard MC, provided average particle density is sufficiently large. Crucially, they propose and analyse two distinct noise coupling strategies: a "Fourier coupling" and a "Right-Most Nearest Neighbours (NN) coupling". We build directly on this work in this dissertation by also investigating the Dean-KAwasaki equation and proposing an alternative noise coupling strategy.

NEED TO ADD REGARDING HPC MLMC IMPLEMENTATIONS AND SPDES (believe none done)

Chapter 3

Methodologies and Validation

This chapter details the practical methodologies and numerical validation that form the foundation for the investigations in this dissertation. Having presented the theoretical groundwork in the preceding chapter, we now turn to the numerical implementations.

This chapter is organised into 3 main parts. First, we outline the MLMC algorithm that is implemented, presenting the method and algorithm implemented in this dissertation.

Second, we focus on the Stochastic Heat Equation (SHE). We begin by defining the specific SHE problem to be analysed. We then derive the explicit finite difference scheme used for its numerical solution and detail the MLMC implementation, including the proposed noise coupling strategies. To validate this implementation, we apply the method to compute several distinct quantities of interest: Moments of the solution's Fourier modes and its total energy. We demonstrate that our implementation converges to the correct, analytically derived values and that the various convergence, decay and growth rates align with those predicted by theory.

Third, we address the Dean-Kawasaki equation. Following the approach of Cornalba and Fischer [6], we present the finite difference scheme, the specific quantity of interest, and the MLMC implementation. We then validate our implementation by demonstrating its convergence and proceed to examine the practical performance of the two distinct coupling strategies proposed in their work.

3.1 MLMC Algorithm

Having outlined the MLMC theory in Section 2.1, we now present the MLMC algorithm that is implemented in this dissertation. This is based on the MLMC implementation provided by Giles [10]. The practical implementation of the MLMC

method follows a two phase approach. The first phase is dedicated to empirically estimating the parameters of the MLMC Complexity Theorem 1, while the second phase uses these paramaters in an adaptive algorithm to achieve a final result.

Before running the main adaptive algorithm, A fixed, large number of N samples is simulated on a specified number of initial levels. From these simulations we gather empirical estimates of $|\mathbb{E}[Y_\ell]|, V_\ell, C_\ell$. Performing linear regression on the logarithms on these quantities against the index level we obtain estimates for the weak decay rate, variance decay rate, and cost growth rate, α, β and γ respectively.

Those estimates can then be passed to the MLMC algorithm which we detail here.

3.2 Numerical Method for the Stochastic Heat Equation

3.2.1 Problem Specification and Finite Difference Scheme

We consider the one-dimensional SHE on a finite spatial domain with homogenous Dirichlet boundary conditions and a zero initial condition. This represents the simplest setting for a parabolic SPDE.

The problem is formally defined as finding the real-valued function $u(t,x)$ that satisfies:

$$\frac{\partial u(t,x)}{\partial t} - \frac{\partial^2 u(t,x)}{\partial x^2} = \xi(x,t), \qquad \text{for } (t,x) \in (0,T] \times (0,1) \tag{3.1a}$$

$$u(0,x) = 0, \qquad \text{for } x \in [0,1] \tag{3.1b}$$

$$u(t,0) = 0, \quad u(t,1) = 0, \qquad \text{for } t \in (0,T] \tag{3.1c}$$

To solve (3.1), we use a scheme obtained using a finite volume approach [20] where (3.1a) is integrated over small, discreet space-time control volumes, and then the resulting integral terms are then approximated. This scheme is:

$$U_j^{n+1} = U_j^n + \frac{\Delta t}{(\Delta x)^2}(U_{j+1}^n - 2U_j^n + U_{j-1}^n) + \Delta W_j^n, \tag{3.2}$$

$$\text{where} \quad \Delta W_j^n = \sqrt{\frac{\Delta t}{\Delta x}} Z_j^n \quad \text{and} \quad Z_j^n \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0,1).$$

The derivation of this scheme now follows.

First, we define a uniform grid. The spatial domain $[0,1]$ is discretised into J intervals of width $\Delta x = 1/J$, with grid points $x_j = j\Delta x$ for $j = 0, 1, \ldots J$. Similarly,

Algorithm 1 Adaptive Multilevel Monte Carlo Algorithm

Input: Target RMSE $\varepsilon > 0$, initial number of samples $N_0 \geq 2$, α, β and γ estimates.

Output: MLMC estimate $\hat{P}_{\text{MLMC}}$, sample counts N_ℓ, variances V_ℓ, and costs C_ℓ for each level.

Initialize:
Set finest level $L \leftarrow 2$.
Set initial required samples $\Delta N_\ell \leftarrow N_0$ for $\ell = 0, \ldots, L$.
Set current sample counts $N_\ell \leftarrow 0$ for $\ell = 0, \ldots, L$.
Initialize statistical accumulators for sums of Y_ℓ and Y_ℓ^2.

while $\sum_{\ell=0}^{L} \Delta N_\ell > 0$ **do** ▷ Loop until all levels have sufficient samples
▷ *1: Generate Samples*
    for all $\ell = 0, \ldots, L$ **do**
        if $\Delta N_\ell > 0$ **then**
            Simulate ΔN_ℓ new samples of the correction term $Y_\ell = P_\ell - P_{\ell-1}$.
            Update the statistical accumulators (sums of Y_ℓ and Y_ℓ^2).
            Update the current sample count: $N_\ell \leftarrow N_\ell + \Delta N_\ell$.
        end if
    end for

▷ *2: Update Estimates and Optimal Allocation*
    for all $\ell = 0, \ldots, L$ **do**
        Compute the current estimate of the variance $V_\ell \leftarrow \mathbb{V}[Y_\ell]$ from the accumulators.
    end for
    Calculate the optimal number of samples N_ℓ^{opt} for all levels using the current variance estimates V_ℓ and the cost per sample C_ℓ in equation (2.2) with a target statistical variance of $\varepsilon^2/2$.
    Compute the required additional samples: $\Delta N_\ell \leftarrow \max(0, N_\ell^{\text{opt}} - N_\ell)$.

▷ *3: Check for Bias Convergence*
    if $\sum \Delta N_\ell < 0.01 \sum N_\ell$ **then** ▷ Check only if sampling has stabilized
        Compute the mean of the finest correction, $|\hat{Y}_L| = \left| \frac{1}{N_L} \sum_{n=1}^{N_L} Y_L^{(n)} \right|$.
        if $|\hat{Y}_L| \geq \varepsilon/\sqrt{2}$ **then** ▷ If bias is still too large
            $L \leftarrow L + 1$. ▷ Add a new, finer level.
            Initialize accumulators, N_L, and ΔN_L for the new level.
        end if
    end if
end while

Compute Final Estimate: $\hat{P}_{\text{MLMC}} \leftarrow \sum_{\ell=0}^{L} \left(\frac{1}{N_\ell} \sum_{n=1}^{N_\ell} Y_\ell^{(n)} \right)$.

let the time interval $[0, T]$ be discretised into N steps of size $\Delta t = T/N$, with the time points $t_n = n\Delta t$ for $n = 0, 1, \ldots N$. Our discrete approximations of u are denoted $U_j^n \approx u(t_n, x_j)$ with $U_j^0 = 0$, $U_0^n = U_J^n = 0$ capturing our initial and boundary conditions respectively.

We integrate (3.1a) over a control volume $C_j^n = [x_j - \frac{\Delta x}{2}, x_j + \frac{\Delta x}{2}] \times [t_n, t_n + 1]$:

$$\int_{t_n}^{t_{n+1}} \int_{x_j - \frac{\Delta x}{2}}^{x_j + \frac{\Delta x}{2}} \frac{\partial u}{\partial t} \, \mathrm{d}x\mathrm{d}t - \int_{t_n}^{t_{n+1}} \int_{x_j - \frac{\Delta x}{2}}^{x_j + \frac{\Delta x}{2}} \frac{\partial^2 u}{\partial x^2} \, \mathrm{d}x\mathrm{d}t = \int_{t_n}^{t_{n+1}} \int_{x_j - \frac{\Delta x}{2}}^{x_j + \frac{\Delta x}{2}} \xi(t, x) \, \mathrm{d}x\mathrm{d}t \quad (3.3)$$

We then approximate each term in this equation. Focussing first on the time derivative term on the LHS:

$$\int_{t_n}^{t_{n+1}} \int_{x_j - \frac{\Delta x}{2}}^{x_j + \frac{\Delta x}{2}} \frac{\partial u}{\partial t} \, \mathrm{d}x\mathrm{d}t = \int_{x_j - \frac{\Delta x}{2}}^{x_j + \frac{\Delta x}{2}} \left[u(t_{n+1}, x) - u(t_n, x) \right] \mathrm{d}x \quad \text{(Fundamental Theorem of Calculus (FTOC))} \quad (3.4a)$$

$$\approx \Delta x (U_j^{n+1} - U_j^n) \quad \text{(Midpoint Rule)} \quad (3.4b)$$

Similarly, for the spatial derivative term:

$$\int_{t_n}^{t_{n+1}} \int_{x_j - \frac{\Delta x}{2}}^{x_j + \frac{\Delta x}{2}} \frac{\partial^2 u}{\partial x^2} \, \mathrm{d}x\mathrm{d}t = \quad (3.5a)$$

$$\int_{t_n}^{t_{n+1}} \left[\frac{\partial u}{\partial x}\left(t, x_j + \frac{\Delta x}{2}\right) - \frac{\partial u}{\partial x}\left(t, x_j - \frac{\Delta x}{2}\right) \right] \mathrm{d}t \quad \text{(FTOC)} \quad (3.5b)$$

$$\approx \frac{\Delta t}{\Delta x} \left[U_{j+1}^n - 2U_j^n + U_{j-1}^n \right] \quad \text{(Midpoint Rule and Central Differences)} \quad (3.5c)$$

Finally, the forcing term on the RHS, via equation (2.6) is equal to

$$\int_{t_n}^{t_{n+1}} \int_{x_j - \frac{\Delta x}{2}}^{x_j + \frac{\Delta x}{2}} \xi(t, x) \, \mathrm{d}x\mathrm{d}t = \sqrt{\Delta x \Delta t} Z_j^n \quad (3.6)$$

where Z_j^n are indepenent and identically distributed standard normal random variables. Finally, we substitute the discrete approximations for each of the three terms back into the integral equation (3.3). This yields the following relation:

$$\Delta x (U_j^{n+1} - U_j^n) = \frac{\Delta t}{(\Delta x)^2} \Delta x (U_{j+1}^n - 2U_j^n + U_{j-1}^n) + \sqrt{\Delta t \Delta x} Z_j^n$$

Rearranging yields the scheme shown in equation (3.2). This scheme is used to propagate the solution forward in time. It is known to be conditionally stable, requiring the Courant-Friedrichs-Lewy (CFL) condition, $\frac{\Delta t}{(\Delta x)^2} \leq \frac{1}{2}$, to be satisfied for convergence, similar to the corresponding deterministic heat equation [20]. A proof of this is given in the Appendix (CITE APPENDIX HERE).

3.2.2 Stochastic Heat Equation - MLMC Implementation

We now describe how scheme (3.2) is implemented for our MLMC algorithm 1. This applies for a generic quantity of interest (QoI) P_L. The QoIs tested for this investigation are outlined in section SECTION HERE.

For each level ℓ, we divide the spatial domain into $n_\ell = 2^{l+1}$ subdivisions resulting in a spatial step size $\Delta x_\ell = 1\ n_\ell$. We fix CFL number $\lambda = \frac{\Delta t}{(\Delta x)^2} = 0.25$. This yields a timestep of $\Delta t_\ell = \lambda(\Delta x_\ell)^2$. This means that for any two consecutive levels the refinement ratios are related via

$$\Delta x_{\ell-1} = 2\Delta x_\ell, \qquad \Delta t_{\ell-1} = 4\Delta t_\ell \tag{3.7}$$

For noise coupling, we investigate 3 strategies: Right-Most Nearest Neighbours (NN) used in [6], Central Coupling (CC) and a Finite Element (FE) based coupling. We outline each of these now. We follow the convention used in [10], in coupling describing the $\ell - 1$ level as the *fine* level and the ℓ level as the *coarse* level.

Right-Most Nearest Neighbours Coupling For the fine level's noise generation, at each interior fine grid point, j, and each fine time step, n, an independent noise increment is generated:

$$\Delta W_{j,f}^n = \sqrt{\frac{\Delta t_f}{\Delta x_f}} Z_j^n, \qquad \text{where } Z_j^n \overset{\text{i.i.d}}{\sim} \mathcal{N}(0,1) \tag{3.8}$$

The coarse level's noise at a grid point, k, corresponding to fine grid point $2j$, over a coarse time step, m, is constructed by aggregating the underlying fine noise. We sum the noise from the corresponding fine grid point and its immediate right-hand Neighbour $2j+1$. This sum is accumulated over the 4 corresponding fine timesteps and then rescaled.

$$\Delta W_{k,c}^m = \frac{1}{2} \sum_{n=4m}^{4m+3} \left(\Delta W_{2j,f}^n + W_{2j+1,f}^n \right) \tag{3.9}$$

The rescaling factor $\frac{1}{2}$ is essential to ensure that the coarse noise $\Delta W_{k,c}^m$ has the correct statistical variance of $\frac{\Delta t_c}{\Delta x_c}$ required for coarse grid simulation.

We note that this method discards the final interior noise increment from the fine grid point. The statistical impact of this is unveiled in analysis of this system.

Central Coupling As an alternative to the asymmetric NN method, we propose a centred coupling strategy. This aims to create a more symmetric correlation between the fine and coarse grids by defining the fundamental source of randomness on a "half-cell" refinement of the spatial grid. From this common source of randomness, the noise increments for both the fine and coarse grids are constructed.

We divide each internal fine grid cell $[x_j - \frac{\Delta x_f}{2}, x_j + \frac{\Delta x_f}{2}] \times [t_n, t_n + \Delta t_f]$ into two half-cells of width $\frac{\Delta x_f}{2}$ For each of these half-cells and for each fine time step n, we generate an independent, fundamental noise increment ζ. Let $\zeta_{j,L}^n$ and $\zeta_{j,R}^n$ be the half-cell noises on the left and right halves of the j-th fine grid point during the n-th time step. These are i.i.d Gaussian random variables with variance equal to the area of the half-cell:

$$\zeta_{j,L}^n, \zeta_{j,R}^n \overset{\text{i.i.d}}{\sim} \mathcal{N}(0, \frac{\Delta x_f \Delta t_f}{2}) \tag{3.10}$$

The noises for the fine finite difference scheme are then constructed by aggregating these fundamental half-cell noises.

$$\Delta W_{j,f}^n = \frac{1}{\Delta x_f}(\zeta_{j,L}^n + \zeta_{j,R}^n) \tag{3.11}$$

where the $\frac{1}{\Delta x_f}$ coefficient ensures $\Delta W_{j,f}^n$ has the correct variance.

Similarly to the NN coupling, the coarse noise is constructed as an aggregation of the underlying fine noises accummulated over the four fine timesteps that constitute a single coarse time step, equivalent to using the exact underlying 16 half-cell noises.

$$\Delta W_{k,c}^n = \frac{1}{\Delta x_c} \sum_{n=4m}^{4m+3} \left(\Delta W_{2j,f}^n + \Delta W_{2j+1}^n\right) = \frac{1}{\Delta x_c} \sum_{n=4m}^{4m+3} \left(\zeta_{2j,L}^n + \zeta_{2j,R}^n + \zeta_{2j+1,L}^n + \zeta_{2j+1,R}^n\right) \tag{3.12}$$

The scaling factors $\frac{1}{\Delta x_f}$ and $\frac{1}{\Delta x_c}$ ensure the resulting noises have the correct variances required on each grid. By further discretising the spatial grid, this method aims perfectly align the noises used for each grid point in both the coarse and fine grids, unlike the NN method.

Finite Element Coupling Method The third coupling strategy is derived from a Galerkin Finite Element Method (FEM) [21] spatial discretisation of the SHE. We transform the infinite-dimensional SPDE into a finite-dimensional system of SDEs, which in turn defines a strcture of the discrete noise and coupling between levels.

We begin by formulating a weak form of the SHE. We multiply (3.1a) by a sufficiently smooth test function $\phi(x)$ and integrate over the spatial domain $D = [0, 1]$. Applying integration by parts yields:

$$(\mathrm{d}u, \phi) + (u_x, \phi_x)\mathrm{d}t = (\mathrm{d}W(t), \phi) \tag{3.13}$$

where $(\cdot, \cdot)$ denotes the L^2 inner product and $dW(t) = \xi(t, x)\mathrm{d}t$ represents Brownian motion.

The Galerkin method seeks an approximate solution, $U(t, x)$ within a finite-dimensional subspace spanned by a set of basis functions [21]. For this problem, we use the standard piecewise linear "hat" basis functions $\phi_j(x)$, defined on a uniform grid with spacing h.

$$\phi_j(x) = \max(0, 1 - |x - x_j|/h) \tag{3.14}$$

The approximate solution is written as

$$U(t, x) = \sum_{j=1}^{J-1} U_j(t)\phi_j(x). \tag{3.15}$$

Here, the $U_j(t)$ are the time-dependent coefficients represents the solution's value at the spatial nodes x_j. By requiring the weak form to hold for every basis function in this space, we obtain a system of SDEs for the vector of coefficients $\mathbf{U}(t)$.

The system of SDEs can be written in matrix form as:

$$M\mathrm{d}\mathbf{U}(t) + K\mathbf{U}(t)\mathrm{d}t = \mathrm{d}W(t) \tag{3.16}$$

where M is a tri-diagonal matrix with elements $M_{ij} = (\phi_i, \phi_j)$. For the hat basis, this gives

$$M_{i,i} = \frac{2}{3}h, \qquad M_{i,i\pm1} = \frac{1}{6}h. \tag{3.17}$$

K is a tridiagonal matrix with entries $K_{ij} = (\phi_i'), \phi_j'$. This gives

$$K_{i,i} = \frac{2}{h}, \qquad K_{i,i\pm1} = -\frac{1}{h} \tag{3.18}$$

$\mathrm{d}W(t)$ is a vector of normal increments. To determine their variance and covariance, we can derive the following covariance function:

$$\mathbb{E}[(f, \mathrm{d}W)(g, \mathrm{d}W)] = (f, g)\mathrm{d}t \tag{3.19}$$

for arbitrary spatial functions f and g. This follows from Definition 4 by considering test functions ϕ_f and ϕ_g:

$$\begin{aligned}\phi_f(s, x) &= f(x)\mathbf{1}_{[t,t+\mathrm{d}t]}(s)\\ \phi_g(s, x) &= g(x)\mathbf{1}_{[t,t+\mathrm{d}t]}(s)\end{aligned}$$

where $\mathbf{1}_{[t,t+\mathrm{d}t]}(s)$ is an indicator function which is 1 for $s \in [t, t+\mathrm{d}t]$, 0 otherwise. Th abstract random variable $W(\phi_f)$ from Definition 4 now represents the noise tested against the spatial function $f(x)$

$$\begin{aligned}\mathbb{E}[(f, \mathrm{d}W)(g, \mathrm{d}W)] &= \mathbb{E}[W(\phi_f)W(\phi_g)]\\ &= \langle \phi_f, \phi_g \rangle_{L^2}\\ &= \int_0^T \int_D \phi_f(s, \mathbf{x})\phi_g(s, \mathbf{x})\, d\mathbf{x}ds\\ &= \int_0^T \int_D \left(f(\mathbf{x})\mathbf{1}_{[t,t+\mathrm{d}t]}(s)\right)\left(g(\mathbf{x})\mathbf{1}_{[t,t+\mathrm{d}t]}(s)\right)\, d\mathbf{x}ds\\ &= \left(\int_D f(\mathbf{x})g(\mathbf{x})\, d\mathbf{x}\right) \times \left(\int_0^T (\mathbf{1}_{[t,t+\mathrm{d}t]}(s))^2\, ds\right)\\ &= (f, g)\, \mathrm{d}t\end{aligned}$$

The vector of normal increments have the following expectations:

$$\begin{aligned}\mathbb{E}[\mathrm{d}W_i \mathrm{d}W_j] = \mathbb{E}[(\mathrm{d}W, \phi_i)(\mathrm{d}W, \phi_j)] = (\phi_i, \phi_j)\mathrm{d}t\\ \mathbb{E}[\mathrm{d}W_i^2] = \frac{2}{3}h\mathrm{d}t, \qquad \mathbb{E}[\mathrm{d}W_i \mathrm{d}W_{i\pm1}] = \frac{1}{6}h\mathrm{d}t\end{aligned}$$

To obtain a fully discrete scheme, we apply the explicit Euler-Maruyama method in time. A common simplification known as mass lumping is used, where M is used, where the mass matrix M is replaced by the diagonal matrix hI. From equation (3.16), this yields the scheme:

$$h\mathbf{U}^{n+1} = h\mathbf{U}^n - K\mathbf{U}^n\Delta t + \Delta W^n \tag{3.20}$$

where the discrete noise vector ΔW^n has the covariance structure $M\Delta t$.

The coupling between a fine level ℓ and a coarse level $\ell - 1$ is derived from the relationship between the FEM basis functions. A coarse grid basis function, ϕ_k^c, can be expressed as a linear combination of the fine grid basis functions:

$$\phi_k^c(x) = \frac{1}{2}\phi_{2k-1}^f(x) + \phi_{2k}^f(x) + \frac{1}{2}\phi_{2k+1}^f(x).$$

This provide a natural way to construct the coarse noise from the fine noise. The coarse noise increment at a coarse node k, $\Delta W_{k,c}^m$, is constructed by applying the same linear weighting to the fine noise increments over the corresponding time steps:

$$\Delta W_{k,c}^m = \frac{1}{2}\sum_{n=4m}^{4m+3}\left(\frac{1}{2}\Delta W_{2k-1,f}^n + \Delta W_{2k,f}^n + \frac{1}{2}\Delta W_{2k+1,f}^n\right).$$

3.2.3 Quantities of interest for the Stochastic Heat Equation

To validate our numerical schemes and analyse the performance of different MLMC coupling strategies, we compute expectations of two distinct quantities of interest (QoI) derived from the solution of the SHE problem (3.1). The first examines the solution's behaviour in the frequency domain through its Fourier modes, while the second is a measure of the total energy of the system.

3.2.3.1 Fourier Modes

For the problem (3.1), the solution can be expanded in a Fourier sine series. This decomposition is justified as the sine functions are the eigenfunctions of the Laplacian operator with the given boundary conditions, and they also form a basis for decomposing the stochastic noise term via the Karhunen-Loéve theorem ()[7], Section 4.1).

$$u(t,x) = \sum_{k=1}^{\infty} u_k(t)\phi_k(x) \qquad \phi_k(x) = \sqrt{2}\sin(k\pi x) \tag{3.21}$$

where $u_k(t)$ is the k-th Fourier mode at time t and phi_k our orthonormal sine basis functions. $u_k(t)$ is given by the L^2 inner product of u with ϕ_k:

$$u_k(t) = \int_0^1 u(t,x)\phi_k(x)\mathrm{d}x \tag{3.22}$$

We derive first an analytic expression for u_k, and then the first and second moments of u_k for validating our MLMC implementation is correct. We will also derive the analytic error we expect to observe and the expected variance decay.

Analytic Moments of Fourier Modes To obtain analytic solutions to the Fourier modes, we first express each term in the SHE governing equation as series expansions (an approach justified in [19], Chapter 5).

$$u_t(t,x) = \sum_{k=1}^{\infty} v_k(t)\phi_k(x) \tag{3.23}$$

$$u_{xx}(t,x) = \sum_{k=1}^{\infty} w_k(t)\phi_k(x)\xi(x,t) = \sum_{k=1}^{\infty} \dot{B}_k(t)\phi_k(x) \tag{3.24}$$

where $\dot{B}(t)$ represents one-dimensional white noise, and v_k and w_k are Fourier modes. Substituting these into (3.1a):

$$u_t - u_{xx} - \xi = 0 \tag{3.25}$$

$$\sum_{k=1}^{\infty} v_k\phi_k - \sum_{k=1}^{\infty} w_k\phi_k - \sum_{k=1}^{\infty} \dot{B}_k\phi_k = 0 \tag{3.26}$$

$$\sum_{k=1}^{\infty} \left(v_k - w_k - \dot{B}_k\right)\phi_k = 0 \tag{3.27}$$

$$\tag{3.28}$$

System Energy A fundamental property of the solution is its total energy, which is defined as the squared L^2-norm of the solution $u(t,x)$ at a final time T. The continuous form of this quantity if given by:

$$P_{\text{energy}} = \int_0^1 (u(T,x))^2 \mathrm{d}x \tag{3.29}$$

We

Bibliography

[1] Assyr Abdulle, Andrea Barth, and Christoph Schwab. Multilevel monte carlo methods for stochastic elliptic multiscale pdes. *Multiscale Modeling & Simulation*, 11(4):1033–1070, 2013.

[2] Assyr Abdulle and Adrian Blumenthal. Stabilized multilevel monte carlo method for stiff stochastic differential equations. *Journal of Computational Physics*, 251:445–460, 2013.

[3] Juan A Acebrón. A probabilistic linear solver based on a multilevel monte carlo method. *Journal of Scientific Computing*, 82(3):65, 2020.

[4] Louis JM Aslett, Tigran Nagapetyan, and Sebastian J Vollmer. Multilevel monte carlo for reliability theory. *Reliability Engineering & System Safety*, 165:188–196, 2017.

[5] Andrea Barth, Annika Lang, and Christoph Schwab. Multilevel monte carlo method for parabolic stochastic partial differential equations. *BIT Numerical Mathematics*, 53(1):3–27, 2013.

[6] Federico Cornalba and Julian Fischer. Multilevel monte carlo methods for the dean–kawasaki equation from fluctuating hydrodynamics. *SIAM Journal on Numerical Analysis*, 63(1):262–287, 2025.

[7] Giuseppe Da Prato and Jerzy Zabczyk. *Stochastic equations in infinite dimensions*, volume 152. Cambridge university press, 2014.

[8] Elsevier. Scopus. https://www.scopus.com, Accessed: August 11, 2025.

[9] Michael B Giles. Multilevel monte carlo path simulation. *Operations research*, 56(3):607–617, 2008.

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

[11] Michael B Giles and Christoph Reisinger. Stochastic finite differences and multilevel monte carlo for a class of spdes in finance. *SIAM journal on financial mathematics*, 3(1):572–592, 2012.

[12] Michael B Giles and Yuan Xia. Multilevel monte carlo for exponential lévy models. *Finance and Stochastics*, 21(4):995–1026, 2017.

[13] Stefan Heinrich. Monte carlo complexity of global solution of integral equations. *Journal of complexity*, 14(2):151–175, 1998.

[14] Ralf Kornhuber, Christoph Schwab, and Maren-Wanda Wolf. Multilevel monte carlo finite element methods for stochastic elliptic variational inequalities. *SIAM Journal on Numerical Analysis*, 52(3):1243–1268, 2014.

[15] Sergey V Lototsky, Boris L Rozovsky, et al. *Stochastic partial differential equations*, volume 11. Springer, 2017.

[16] Yan Luo and Zhu Wang. A multilevel monte carlo ensemble scheme for random parabolic pdes. *SIAM Journal on Scientific Computing*, 41(1):A622–A642, 2019.

[17] Étienne Pardoux et al. *Stochastic partial differential equations: An introduction*. Springer, 2021.

[18] Chang-han Rhee and Peter W Glynn. Unbiased estimation with square root convergence for sde models. *Operations Research*, 63(5):1026–1043, 2015.

[19] Walter A Strauss. *Partial differential equations: An introduction*. John Wiley & Sons, 2007.

[20] Endre Süli. Numerical solutions to partial differential equations, September 2024.

[21] Endre Süli. Lecture notes on finite element methods for partial differential equations, January 2025.

[22] John B Walsh. An introduction to stochastic partial differential equations. In *École d'Été de Probabilités de Saint Flour XIV-1984*, pages 265–439. Springer, 2006.