Multi Level Monte Carlo Methods for Atmospheric Dispersion Modelling

submitted by

Sarah Elizabeth Cook

for the degree of Master of Philosophy

of the

University of Bath

Department of Mathematical Sciences

September 2013

COPYRIGHT

Signature of Author . . . .

Sarah Elizabeth Cook

Summary

The Met Office uses the NAME dispersion model to solve stochastic differential equations (SDEs) for predicting the transport and spread of atmospheric pollutants. Time stepping methods for this SDE dominate the computation time. In particular the slow convergence of the Monte Carlo Method imposes limitations on the accuracy with which predictions can be made on operational timescales.

We review the theory of both the Standard and Multi Level Monte Carlo Methods, and in particular the complexity theorems discussed in [9] in a more general context. We then argue how it can potentially give rise to significant gains for this problem in atmospheric dispersion modelling.

To verify these theoretical arguments numerically, we consider two model problems; a simplified problem which corresponds to homogeneous turbulence and is used by the Met Office for long term predictions, as well as a full non-linear model problem close to that used by the Met Office for atmospheric dispersion modelling.

For both model problems we performed numerical tests in which we observed significant speed-up as a result of the implementation of the Multi Level Monte Carlo Method. The numerically observed convergence rates are also confirmed by a full theoretical analysis for the simplified model problem.

Several open questions, such as the correct treatment of reflective boundary conditions and the Multi Level coarsening factor, are also addressed. We present interesting preliminary numerical results which will be useful for extending the method to more realistic scenarios and hopefully allow it to be used in an operational setting in the future.

Contents

Chapter 1

Introduction

1.1 Motivation

Predicting the spread of airborne pollutants is important in a variety of applications; in particular it allows for the informed formulation of emergency responses to events such as industrial fires, volcanic eruptions or even nuclear disasters. Consequently, it is vital that these predictions can be made quickly and accurately in order for efficient and effective responses to be implemented.

The eruption of Eyjafjallajkull, in Iceland in 2010 [1][2][3], resulted in an extensive ash cloud that caused major disruption for European air travel. The facility to model the dispersion of such airborne pollutants was crucial to managing the situation effectively and thus, in this case, to minimising the cost to airlines as well as the disruption to passengers. As such, being able to model atmospheric dispersion quickly and accurately has substantial economic, as well as wider, social benefits.

The Fukushima Daiichi nuclear disaster in 2011 is another case in point for the necessity of such models. The meltdown of the nuclear reactors resulted in airborne radioactive material which posed an immediate threat to the local population as well as those further afield. The fast and accurate prediction of the dispersion of this material was thus of prime concern in order to take action in the interest of public safety.

1.2 Background of Problem

The Met Office's current atmospheric dispersion model, NAME (Numerical Atmospheric Modelling Environment) [4][5][6] was created in response to the Chernobyl disaster in 1986 [7][8], to ensure that in the event of such an incident occurring again, relevant data can be provided and all necessary precautions can be taken.

NAME uses random walk techniques to model turbulence in the atmosphere, namely Monte Carlo simulation; however Monte Carlo methods introduce stochastic errors. These stochastic errors are proportional to the inverse square root of the number of model particles, thus in order to minimise this error we must take a large number of these sample particles. As a result obtaining an approximation with significant accuracy is expensive and, in particular, slow to run.

In the above mentioned applications, for example, time constraints are such that the Met Office must often use simplified physics/models or compromise on accuracy in order to be able to model the dispersion within the required, generally short, time period.

Recent applications of Multi Level Monte Carlo Methods (MLMC) in the context of computational finance and the resultant reduction in computation times are given in [9]. We endeavour to recreate these gains in the context of atmospheric dispersion modelling, with the aim to enable the Met Office to use more complex, realistic parameters and work to a higher degree of accuracy whilst retaining the low computational cost required to obtain predictions on operational timescales.

1.3   Model Problem

In the case of atmospheric dispersion modelling it is the turbulent dispersion that we wish to model by random walk techniques and resolved large scale flow is treated deterministically. To do this NAME tracks thousands of independent model particles through the atmosphere to obtain approximate statistics of the dispersion.

In d-dimensions, at time t, we take a typical particle with trajectory $\mathbf{X}(t) \in \mathbb{R}^d$, mean velocity field $\mathbf{v}(\mathbf{X}(t), t) \in \mathbb{R}^d$ and the deviation of the velocity from the mean due to turbulence given by $\mathbf{U}(t) \in \mathbb{R}^d$. The time evolution of $\mathbf{U}(t)$ and $\mathbf{X}(t)$ can be described by the stochastic differential equations (SDEs) below,

$$\begin{aligned}
\mathrm{d}\mathbf{U}(t) &= \mathbf{a}(\mathbf{U}(t), \mathbf{X}(t), t)\mathrm{d}t + \mathbf{b}(\mathbf{X}(t), t)\mathrm{d}\mathbf{W}(t), \\
\mathrm{d}\mathbf{X}(t) &= \big[\mathbf{v}(\mathbf{X}(t), t) + \mathbf{U}(\mathbf{X}(t), t)\big]\mathrm{d}t,
\end{aligned} \tag{1.1}$$

here $\mathbf{W}(t)$ is a d-dimensional Wiener process, both $\mathbf{X}(t)$ and $\mathbf{U}(t)$ are d-dimensional random processes and $\mathbf{a}(\mathbf{U}(t), \mathbf{X}(t), t) \in \mathbb{R}^d$ and $\mathbf{b}(\mathbf{X}(t), t) \in \mathbb{R}^{d \times d}$ are problem dependent, possibly non-linear functions which may or may not depend on $\mathbf{X}(t)$ and $\mathbf{U}(t)$, [11][12].

When the coefficients $\mathbf{a}$ and $\mathbf{b}$ are complicated (i.e. non-constant), the SDE cannot be easily solved and instead we turn to numerical methods to approximate the solution.

Such methods involve computing an approximate solution to our problem at discrete time points, using the solution at previous time steps to approximate the solution at the next time step, so-called *time-stepping methods*.

There are a variety of different time-stepping methods, for example Forward Euler, Symplectic Euler, Crank-Nicolson and the Leapfrog Method, each of which has its own convergence and stability properties as well as cost per time step. For this project we use both the Forward Euler and the Symplectic Euler method, with time step size h. The Symplectic Euler method is a modification of the Forward Euler method and is used for solving Hamiltonian systems of equations; we note, however, that our system of equations in (1.1) has a form close to that of a Hamiltonian system and so we apply the Symplectic Euler method in an extended way to this problem, [10]. Indeed for our general model the Symplectic Euler method, which is the method currently implemented in NAME, is given by,

$$
\begin{aligned}
\mathbf{U}_{n+1} &= \mathbf{U}_n + \mathbf{a}(\mathbf{U}_n, \mathbf{X}_n, t_n)h + \mathbf{b}(\mathbf{X}_n, t_n)\Delta\mathbf{W}_n, \\
\\
\mathbf{X}_{n+1} &= \mathbf{X}_n + \big[\mathbf{v}(\mathbf{X}_n, t_n) + \mathbf{U}_{n+1}\big]h,
\end{aligned}
\tag{1.2}
$$

where $t_n = nh$ is the time at our n-th time step, with $T = Mh$ the final time, at which we are interested in the solution of the SDE and M the number of time steps. Here $\mathbf{U}_n \approx \mathbf{U}(t_n)$ is our approximation of $\mathbf{U}$ at the n-th time step and similarly $\mathbf{X}_n \approx \mathbf{X}(t_n)$ is our approximation of $\mathbf{X}$ at the n-th time step and, finally, $\Delta\mathbf{W}_n$ is a d-vector comprised of independent normally distributed random variables with mean 0 and variance h, which correspond to the increments in each of the d directions for the n-th step in the random walk. We note that NAME also supports non-Gaussian turbulence, but for this project we focus only on Gaussian turbulence.

For the Forward Euler Method we compute $\mathbf{X}_{n+1}$ using the values for $\mathbf{X}_n$ and $\mathbf{U}_n$ computed at the previous time step, however for the Symplectic Euler, we instead compute $\mathbf{X}_{n+1}$, using the newly computed $\mathbf{U}_{n+1}$ and $\mathbf{X}_n$ computed at the previous time step. The two methods can be compared directly, in the context of their implementation in the algorithm, in the following table.

Forward Euler	Symplectic Euler
Loop over M time steps	Loop over M time steps
$\quad\mathbf{U}_{n+1} = \mathbf{U}_n + \mathbf{a}(\mathbf{U}_n, \mathbf{X}_n, t_n)h + \mathbf{b}(\mathbf{X}_n, t_n)\Delta\mathbf{W}_n,$	$\quad\mathbf{U}_{n+1} = \mathbf{U}_n + \mathbf{a}(\mathbf{U}_n, \mathbf{X}_n, t_n)h + \mathbf{b}(\mathbf{X}_n, t_n)\Delta\mathbf{W}_n,$
$\quad\mathbf{X}_{n+1} = \mathbf{X}_n + \big[\mathbf{v}(\mathbf{X}_n, t_n) + \mathbf{U}_n\big]h.$	$\quad\mathbf{X}_{n+1} = \mathbf{X}_n + \big[\mathbf{v}(\mathbf{X}_n, t_n) + \mathbf{U}_{n+1}\big]h.$
End loop	End loop

The Forward and Symplectic Euler methods have the same cost and for small time step sizes they differ little from each other, however the Symplectic Euler is better at preserving the underlying Hamiltonian dynamics. For our initial tests we used the Forward Euler Method and later we implemented the Symplectic Euler method, in line with that used in the Met Office's model NAME.

Once we have computed our Euler approximation to the solution, we are generally interested in some functional of this solution, say $Q(\mathbf{U}(T), \mathbf{X}(T))$, we call this our quantity of interest (QoI). In particular, for the Met Office's purposes, this is usually the probability of a particle being in a particular subsection of the domain at end time T. Our Euler approximation of this quantity of interest, Q_M, will then be the functional evaluated with our approximate solution of the SDE, i.e. $Q(\mathbf{U}_M, \mathbf{X}_M)$.

1.4   Aims

The aim of this thesis is to show that the gains seen in terms of computational complexity in [9] can be replicated in the context of atmospheric dispersion modelling. Thus we hope to demonstrate that the application of Multi Level Monte Carlo Methods to this problem has the potential to allow the Met Office to use more complex, realistic parameters and work to a higher degree of accuracy whilst retaining the low computational cost required to obtain predictions on operational timescales.

1.5   Achievements

We have identified that for a simplified 1-dimensional model problem the application of Multi Level Monte Carlo gives significant gains over the Standard Method which is currently used by the Met Office. We have also shown that gains are possible for a more realistic test case, opening up a number of interesting research questions that follow on from this thesis.

In particular we have written original code in MATLAB for both a simplified and a more realistic test case, allowing us to directly compare the results of the Standard and Multi Level Monte Carlo methods for these two model problems.

The simplified problem corresponds to that which is used by the Met Office to make predictions on longer time scales; studying this simpler problem allowed us to complete a full theoretical analysis, to which we could then compare our numerical results. From this theoretical analysis we were able to identify the potential for improvement from the implementation of the Multi Level Method; in particular it enabled us to apply the theory described in [9] to calculate the computational cost of the Multi Level Method

7

for this simplified problem and compare it to that of the Standard Method.

When our quantity of interest was chosen to be the probability of a particle being in a particular "box" at a certain time T, for the simplified model problem, we were able to prove that the computational cost of the Multi Level Method is $\mathcal{O}(\varepsilon^{-2}(\log \varepsilon)^2)$ compared to $\mathcal{O}(\varepsilon^{-3})$ for the Standard, where ε is a specified error tolerance. We were also able to verify these results numerically using the MATLAB code we had written.

Further to this, studying a different quantity of interest, the mean particle position, we saw that the method proved to perform even better, giving an order of magnitude improvement over the Standard, with the Multi Level Method's computational cost, in this case, being $\mathcal{O}(\varepsilon^{-2})$.

In addition to this we also showed numerically that similar gains are possible for a more realistic model, in which case it was not feasible to prove this analytically. Finally, we modified the realistic model in order to represent as closely as possible the behaviour of real life airborne pollutant particles in line with that used at the Met Office. To do this we reflected any particles that reached the ground in order to prevent them going beneath the ground. This reflection, however, gave way to some unusual results and the Multi Level Method appeared not to be as successful in this case. These unusual results raise several interesting questions, the investigation of which would constitute the next steps in the continuing research of this topic.

It is important to note that the error tolerances considered here are those which are based on practical use and as such are relatively large compared to those generally small error tolerances considered in mathematical papers. In fact for this problem, had we considered smaller error tolerances the results would likely have been very impressive, however for this project the aim has been to achieve improvements for accuracies in line with those that the Met Office require for the practical prediction of the dispersion of airborne pollutants; with this in mind the results in this thesis are in fact very positive.

In conclusion, this thesis shows that the Multi Level Monte Carlo method has good potential to provide improved performance over the Standard in this application to atmospheric dispersion modelling, with further work required to identify the exact magnitude of these improvements as well as the optimal parameter choices.

1.6   Structure

This thesis begins by giving a background of Monte Carlo Methods and their application to atmospheric dispersion modelling. We study the statistical errors associated with this method and the limitations controlling these errors imposes on its operational use. From this we establish the need for an improvement to the Standard Monte Carlo

Method, in particular for predicting the spread of airborne pollutants.

We go on to study the Multi Level Monte Carlo Method, the resultant statistical errors and how an optimal method can be used to control these errors. We conclude the chapter by contrasting computational complexity theorems for both the Standard and Multi Level Monte Carlo Methods, identifying the potential for significant improvement over the Standard Method, provided certain conditions are satisfied.

To see how well the Multi Level Method performs we first look at a simplified, yet relevant, 1-dimensional model problem, which describes homogeneous turbulence. The main purpose of this simplified model is to allow us to use tools of mathematical analysis which would prove difficult, if not impossible for a more complicated case. The analysis allows us to look at how the well the Multi Level Method performs for this model problem which we then compare to numerical results.

For this simplified problem we perform numerical tests studying different quantities of interest, varying the time spans over which we run the model as well as using different accuracy tolerances and output cells in which we are interested in the concentration of particles, with a view to understanding how the performance of the method changes as we vary these parameters.

Having established the potential for improvement over the Standard Method for the simplified case, we then look at a more realistic test case. We carefully study the SDE to be solved and address the problems that arise as a result of the more complicated, non-linear model as well as the stability constraints imposed by the time-stepping method.

We then go on to introduce reflection of the particles at the boundary layers to better mimic the behaviour of real life airborne pollutant particles. We look at numerical tests carried out for both the full model without reflection at the boundaries and that with reflection at the boundaries. From these we are able to draw conclusions about the significance of the gains made by applying Multi Level Monte Carlo Methods to this problem in atmospheric dispersion modelling.

Finally we identify the next steps in the full implementation of this method, including further testing and improvements to the way in which the method is currently applied.

Chapter 2

Monte Carlo

In this section we discuss the Standard Monte Carlo Method, in particular its application to atmospheric dispersion modelling in line with its implementation in the Met Office's current model NAME. We study the statistical error introduced by this method of approximation as well as its computational complexity and the limitations these impose on the widespread use of the method.

We then move on to give a background of the Multi Level Monte Carlo Method, detailing the method itself and the reasons behind the gains we see as a result of its implementation. Using the theorem proved in [9], we study again the computational complexity, allowing us to identify the cases in which we expect to see significant improvements in terms of cost from the application of Multi Level Monte Carlo.

2.1 Standard Monte Carlo

2.1.1 Background

Although statistical sampling, the predecessor of the Monte Carlo Method, has been around for centuries, the samples were previously carried out by means of painstaking "by-hand" samples. Thus it was the advent of modern computing power which gave way for the idea to be applied to a much broader range of problems, allowing it to become what is still a very powerful tool in modern science.

The birth of Monte Carlo methods is widely attributed to the paper by Metropolis and Ulam in 1949, [13][14][15], in which they used statistical sampling methods to evaluate integro-differential equations for various physical problems, and in which they named this process the Monte Carlo Method. This paper is considered the main driving force in paving the way for the growth of this now widely used method.

Historically Statistical Sampling or Monte Carlo Methods are known for their use

in evaluating integrals of non-smooth functions, which pose problems for traditional quadrature methods of numerical integration. One of the key advantages of using Monte Carlo methods, however, is that the cost of the method, unlike other quadrature methods, does not grow exponentially with dimension, meaning the method is particularly competitive for higher dimensional problems.

2.1.2 Method

The general idea of Monte Carlo methods is that we have some quantity of interest, that we cannot compute explicitly and thus for which we require an approximation. We write our quantity of interest as the expected value of some random variable. To approximate this quantity numerically we take a number of samples of this random variable and compute the mean. The law of large numbers then says that for a sufficiently large number of samples this mean will approximate the expected value [15][16]. To illustrate this we consider the following example.

Say we wish to compute the volume of Ω, where Ω is the d-dimension sphere defined by,

$$\Omega := \{\mathbf{x} \in \mathbb{R}^d : \|\mathbf{x}\| \leq \tfrac{1}{2}\}.$$

Then we define our quantity of interest as the expected value of the indicator function on Ω, i.e.

$$Q(\mathbf{x}) = \chi_\Omega(\mathbf{x}) := \begin{cases} 1 & \text{if } \mathbf{x} \in \Omega \\ 0 & \text{otherwise.} \end{cases}$$

We consider a sample of N, uniformly distributed random variables, $\mathbf{x}_i \in \left[-\tfrac{1}{2}, \tfrac{1}{2}\right]^d$, whose probability density function (pdf) is given by,

$$p(\mathbf{x}) = \begin{cases} 1 & \text{if } \mathbf{x} \in \left[-\tfrac{1}{2}, \tfrac{1}{2}\right]^d \\ 0 & \text{otherwise.} \end{cases}$$

By the definition of the expected value of a random variable we have,

$$\mathbb{E}[Q(\mathbf{x})] = \int_{\mathbb{R}^d} p(\mathbf{x})Q(\mathbf{x}) \, d\mathbf{x} = \int_{\left[-\frac{1}{2}, \frac{1}{2}\right]^d} Q(\mathbf{x}) \, d\mathbf{x} = \int_\Omega d\mathbf{x} = V(\Omega).$$

Then as we noted above, the law of large numbers says that for a sufficiently large number of samples, N, we can approximate the expected value by the mean, i.e.

$$\overline{Q} := \frac{1}{N} \sum_{i=1}^{N} Q(\mathbf{x}_i) \approx \mathbb{E}[Q] = \int_{\Omega} \mathrm{d}\mathbf{x} = V(\Omega).$$

Thus $\overline{Q}$ gives us an approximation of the volume of the sphere, Ω.

For $d = 1$ and for a sufficiently smooth function, this method would be easily outperformed by any standard quadrature method, however, for higher dimensional problems, the cost of standard (tensor-product) quadrature methods grows exponentially with dimension whereas the cost of Monte Carlo methods remain almost entirely unchanged [15].

This independence of dimension gives the competitive edge to Monte Carlo methods as well as their ability to cope with non-smooth and discontinuous functions, for which Gaussian quadrature will not work, and as a result Monte Carlo methods are favoured for a vast range of applications.

For the application to atmospheric dispersion modelling our random variable will be our Euler approximation, Q_M, to the quantity of interest Q, recalling that M is number of Euler time steps. So we have,

$$\mathbb{E}[Q] \approx \mathbb{E}[Q_M] \approx \frac{1}{N} \sum_{i=1}^{N} Q_M^{(i)}, \tag{2.1}$$

where $Q_M^{(i)}$ are independent samples of Q_M and so we have that the expected value of the quantity of interest, Q, is approximately equal to the expected value of the Euler approximation of the quantity of interest, Q_M, which can be estimated by the mean of N samples of the random variable Q_M, for sufficiently large N.

We are replacing the expected value with a **finite** number of samples and as such we introduce an error, to which we refer as the *sampling error*. This sampling error thus depends on the number of samples we take, so, in order to reduce this sampling error, we must take more samples.

According to the Central Limit Theorem the relative error of Monte Carlo methods scales with $1/\sqrt{N}$, [15], thus the number of samples required to reduce the sampling error by a factor of N is of order N^2. In other words, in order to make our approximation 10 times more accurate we must take 100 times more samples. This relatively slow convergence results in the high computational cost associated with the method and, although computing power continues to improve, Monte Carlo methods are still prohibitively expensive particularly for those problems with high computational costs per sample.

The independence of dimension, however, remains a crucial motivating factor for

the continued use of these methods, since they allow for complex high dimensional problems to be modelled with costs similar to those of 1 or 2 dimensional problems.

It is for these reasons that research into modifications of Monte Carlo methods, such as the Multi Level Monte Carlo Metho, is currently thriving with ambitions to create methods which allow for more accurate predictions to be made on ever shorter operational timescales.

2.1.3   Estimator

As discussed in Section 1.3 we solve our SDE using the Forward or Symplectic Euler method, with M time steps of size h, giving us **one** of our Monte Carlo samples, which we denote by,

$$Q_M^{(i)} = Q\left(\mathbf{U}_M^{(i)}, \mathbf{X}_M^{(i)}\right).$$

We then repeat this process for $i = 1, \ldots, N$, with N being the number of Monte Carlo samples. Taking the mean of the N samples gives us an estimator for our quantity of interest which, in the Met Office's case, would be the probability of a particle landing in a specified cell of the output grid at time T.

More precisely, we define the Monte Carlo estimator, $\hat{Q}_{M,N}$, of our quantity of interest, Q, to be,

$$\hat{Q}_{M,N} := \frac{1}{N} \sum_{i=1}^{N} Q_M^{(i)}, \tag{2.2}$$

where $Q_M^{(i)} = Q(\mathbf{U}_M^{(i)}, \mathbf{X}_M^{(i)})$ is the ith sample of the Euler approximation of Q_M from N independent samples, where M refers to the number of time steps in the Euler discretisation with time step size h.

2.1.4   Mean Square Error

Both the Euler and Monte Carlo methods introduce errors, the Monte Carlo method introduces a *sampling error* resulting from replacing the expected value with a finite number of samples, as discussed in Section 2.1.2 and the Euler method introduces an error resulting from evaluating the SDE only at discrete time points, to which we refer as the *discretisation error* or *bias error*.

The discretisation error will depend on the size of the time step, h; indeed the root mean square error (RMSE) is known to be $\mathcal{O}(h)$ as given in [17]. Thus taking a large leap forward in time will naturally give an inaccurate solution of the SDE at the next time point, and these errors will propagate with each step forward in time. Taking

small time steps will give a more accurate solution, but this means we must compute the updates of the system of equations in (1.2) for a large number of points, $M = T/h$, resulting in a high computational cost. Our problem becomes that of ensuring that we obtain a solution which is sufficiently accurate whilst the number of time steps remains such that the computation time is feasible to be repeated for each of the generally large number of Monte Carlo samples.

In general, for applications of the method we would like to obtain an approximation which is accurate to some specified tolerance, which we denote by ε. With this in mind, we study the mean square error (MSE). We define the expected mean square error as the expected value of the square of the difference between our estimator and the expected value of the quantity of interest, i.e.

$$\mathbb{E}\left[\left(\hat{Q}_{M,N} - \mathbb{E}[Q]\right)^2\right], \tag{2.3}$$

where $\hat{Q}_{M,N}$ is our approximation to $\mathbb{E}[Q]$, as described in equation (2.2).

To obtain an approximation which is accurate to our tolerance ε, we require that the mean square error is bounded by ε^2.

The formula for the mean square error given in (2.3) can be expanded to demonstrate the contribution from the sampling error and that from the discretisation error, see Appendix A.1. Thus we have that our mean square error is, [21],

$$\mathbb{E}\left[\left(\hat{Q}_{M,N} - \mathbb{E}[Q]\right)^2\right] = \underbrace{N^{-1}\mathbb{V}[Q_M]}_{\text{sampling error}} + \underbrace{\left(\mathbb{E}[Q_M - Q]\right)^2}_{\text{discretisation error}}. \tag{2.4}$$

We need to control both of these sources in order to satisfy the desired bound of ε^2 on the mean square error. To do so it is sufficient, although possibly not optimal, to require that each term is bounded by $\varepsilon^2/2$.

We note that the discretisation error is dependent on the time step size h only, whereas the sampling error depends also on the number of Monte Carlo samples.

2.1.5 Complexity Theorem

The following complexity theorem, proved in [22], allows us to examine, based on the properties of our model problem, how expensive the Standard Monte Carlo algorithm is, and will allow for us to compare this against the cost of the Multi Level algorithm.

Theorem 2.1.1. *Assuming that,*

(a) $|\mathbb{E}[Q_M - Q]| = \mathcal{O}(h^\alpha)$ *- Discretisation Error*

14

(b) $Cost(Q_M^{(i)}) = \mathcal{O}(h^{-\gamma})$ - *Cost per sample*

Then there exists time step h and number of samples N such that the total cost to obtain mean square error,

$$\mathbb{E}\left[\left(\hat{Q}_{M,N} - \mathbb{E}[Q]\right)^2\right] \leq \varepsilon^2,$$

is,

$$
\begin{aligned}
Cost(\hat{Q}_{M,N}) \quad &= \# \ of \ Samples \ \times \ Cost \ per \ Sample \\[6pt]
&= \mathcal{O}(\varepsilon^{-2-\gamma/\alpha}).
\end{aligned}
$$

(2.5)

Proof. The result is obtained by choosing $N^{-1} = \mathcal{O}(\varepsilon^2)$ and $h = \mathcal{O}(\varepsilon^{1/\alpha})$. Indeed, the requirement that $N^{-1} = \mathcal{O}(\varepsilon^2)$ results from splitting the error between the sampling and discretisation error and requiring that the sampling error be less than $\varepsilon^2/2$. The sampling error was shown to be $N^{-1}\mathbb{V}[Q_M]$, see Section 2.1.4, so we require,

$$
\begin{aligned}
N^{-1}\mathbb{V}[Q_M] \quad &\leq \quad \tfrac{1}{2}\varepsilon^2 \\[6pt]
\implies N \quad &\geq \quad 2\mathbb{V}[Q_M]\varepsilon^{-2} \\[6pt]
\implies N \quad &= \quad \mathcal{O}(\varepsilon^{-2}), \qquad \text{since } \mathbb{V}[Q_M] \text{ is independent} \\
&\qquad\qquad\qquad\qquad\quad \text{of the number of samples.}
\end{aligned}
$$

Similarly the requirement that $h = \mathcal{O}(\varepsilon^{1/\alpha})$ results from the bound on the discretisation error term, i.e. requiring that,

$$\left(\mathbb{E}[Q_M - Q]\right)^2 \leq \frac{1}{2}\varepsilon^2$$

Condition (a) gives $\mathbb{E}[Q_M - Q] = \mathcal{O}(h^\alpha)$, so the bound on the discretisation error can be written as,

$$
\begin{aligned}
\mathcal{O}(h^\alpha) = \mathbb{E}[Q_M - Q] \quad &\leq \quad \frac{1}{\sqrt{2}}\varepsilon \\[6pt]
\implies h^\alpha \quad &\lesssim \quad \frac{1}{D\sqrt{2}}\varepsilon, \qquad \text{for some constant } D \\[6pt]
\implies h \quad &= \quad \mathcal{O}(\varepsilon^{1/\alpha}), \quad \text{since } \frac{1}{D\sqrt{2}} \text{ is constant.}
\end{aligned}
$$

$\square$

15

The complexity theorem illustrates that for any time stepping method, even those with a good rate of convergence for the discretisation error or a low cost per sample, we are limited by the high cost of the Monte Carlo method resulting from the large number of samples required, meaning that the cost will always be greater than $\mathcal{O}(\varepsilon^{-2})$.

For both the Forward and Symplectic Euler Methods, for example, we have that the cost per sample is proportional to the number of time steps i.e. $\mathcal{O}(M) = \mathcal{O}(h^{-1})$, so $\gamma = 1$ in the complexity theorem, meaning that our cost is of order $\mathcal{O}(\varepsilon^{-2-1/\alpha})$.

In order to see how expensive the method will be it is important that we establish the rate of convergence of the discretisation error. If, for example, the discretisation error is $\mathcal{O}(h)$, this would mean that the cost of the Standard Monte Carlo would be $\mathcal{O}(\varepsilon^{-3})$.

In this case, for the Standard Method, in order to gain one decimal place of accuracy, the cost becomes 1000 times more expensive. Thus we see just how quickly the Standard Monte Carlo method becomes prohibitively expensive, hence we look to the Multi Level Monte Carlo Method with the aim to attain the same mean square error but with a reduced computational cost.

2.2   Multi Level Monte Carlo

2.2.1   Background

The idea behind Multi Level Monte Carlo is that we obtain approximations on a number of different "levels", each of which is computed using a finer time step size than the previous level. The idea being that we get the accuracy of the finest time step, but focus the bulk of the computations on the coarser, cheaper levels by only computing a small number of samples on the finer, more expensive levels.

We achieve this by exploiting the linearity of the expectation value, which allows us to compute a large number of samples with a coarse time step, h_0, which are cheap to compute, for example $h_0 = T$ or $h_0 = T/2$ (recalling that T is the end time at which we wish to solve the SDE). This gives a "rough" estimate for our quantity of interest. Then we look at corrections with respect to successively smaller time steps, we call these the "levels" in line with the more general notation in Giles' paper [9].

On each level, we look at the difference between estimates on that level and that on the previous level, i.e. the difference between the estimate with the finer time step and that with the coarser time step, where the coarser time step is a factor of s, larger than the previous level, i.e, $h_{\ell-1} = sh_\ell$, for $s \in \mathbb{N}\backslash\{1\}$. The index ℓ refers to the level, $\ell = 0, \ldots L$, where h_L is our finest time step. This difference is effectively the correction of the error made by using the larger Euler time step to an error corresponding to the

smaller time step, and thus by computing successive corrections we obtain a final estimate which has bias corresponding to a discretisation with the smallest time step h_L.

The advantage over Standard Monte Carlo with the same time step size h_L lies with the fact that we compute fewer samples on the finer levels and thus incur a smaller computational cost on these more expensive levels. The method for determining how many samples are required depends on the variance of these differences and the gains result from the smaller variance of the differences on the finer levels; consequently the success of the method is dependent on the rate of decay of this variance with respect to h.

2.2.2 Method

More concisely, we have the levels $\ell = 0, \ldots L$, and $\{h_\ell : \ell = 0, \ldots, L\}$ a decreasing sequence, where h_0 represents the coarsest time step and h_L the finest. We also assume $h_{\ell-1} = s h_\ell$ for $s \in \mathbb{N} \backslash \{1\}$. We take the number of points in the Euler discretisation to be $M_\ell = T/h_\ell$ (assuming for simplicity that T/h_ℓ is an integer for all ℓ).

We define $Y_\ell = Q_{M_\ell} - Q_{M_{\ell-1}}$ as the difference between the estimate of our quantity of interest, Q, with time step h_ℓ and that with the larger time step $h_{\ell-1}$ computed on the same Brownian path. Then the linearity of the expected value allows us to formulate the following, [21]:

$$
\begin{aligned}
\mathbb{E}[Q_{M_L}] \quad &= \mathbb{E}[Q_{M_0}] - \mathbb{E}[Q_{M_0}] + \mathbb{E}[Q_{M_1}] - \mathbb{E}[Q_{M_1}] + \ldots \\
&\qquad\qquad \cdots + \mathbb{E}[Q_{M_{L-1}}] - \mathbb{E}[Q_{M_{L-1}}] + \mathbb{E}[Q_{M_L}] \\
&= \mathbb{E}[Q_{M_0}] + \sum_{\ell=1}^{L} \mathbb{E}[Q_{M_\ell} - Q_{M_{\ell-1}}] \\
&= \sum_{\ell=0}^{L} \mathbb{E}[Y_\ell],
\end{aligned}
\tag{2.6}
$$

where $Y_0 = Q_{M_0}$. In particular each of the $Y_\ell = Q_{M_\ell} - Q_{M_{\ell-1}}$ terms is estimated independently, with the same Brownian increments used on one level of the algorithm to compute both Q_{M_ℓ} and $Q_{M_{\ell-1}}$. It is important to note that the linearity of the expected values means that estimating these Y_ℓ's independently does not change the total expectation value.

To clarify how on one level of the algorithm the same Brownian increments are used for estimating both Q_{M_ℓ} and $Q_{M_{\ell-1}}$, we consider a 1-dimensional example. We have a Wiener process which for our time-stepping method in 1-dimension is given by a vector, $\Delta\mathbf{W}^\ell$, of normally distributed random variables with mean 0 and variance h_ℓ. The elements of the vector $\Delta\mathbf{W}^\ell$ are the increments in the random walk and correspond to the finer time step, h_ℓ.

Since we know that, given two independent normally distributed random variables,

$$X \sim N(\mu_X, \sigma_X^2), \qquad Y \sim N(\mu_Y, \sigma_Y^2),$$

their sum $X + Y$ is also a normal random variable, whose distribution is given by,

$$X + Y \sim N(\mu_X + \mu_Y, \sigma_X^2 + \sigma_Y^2).$$

Then to obtain the Brownian increment corresponding to the coarser time step, $h_{\ell-1}$, we simply take the sum of the two Brownian increments on the finer level. These will simply be $\Delta W_j^\ell \sim N(0, h_\ell)$ and $\Delta W_{j+1}^\ell \sim N(0, h_\ell)$ and thus the increment for the coarser level, $\Delta W_j^{\ell-1}$, can be written as,

$$\Delta W_j^{\ell-1} = \Delta W_{2j-1}^\ell + \Delta W_{2j}^\ell \quad \text{for } j = 1, \dots, M_{\ell-1},$$

so $\Delta W_j^{\ell-1} \sim N(0, 2h_\ell) = N(0, h_{\ell-1})$.

The Multi Level method allows us to obtain an estimate with accuracy equivalent to that of the Standard Monte Carlo with time step h_L by calculating the corrections on the levels up to level L, rather than by direct calculation, without invoking additional error terms.

Although intuitively this computation of the differences would seem to incur greater costs, we gain by observing that we require fewer samples on these correction levels in order to reduce the sampling error.

2.2.3   Estimator

Our Multi Level Monte Carlo estimator, $\hat{Q}_{M_L, \{N_\ell\}}^{(ML)}$, whose finest level is $h_L = T/M_L$, is simply the sum of the estimators on each level, given by,

$$\hat{Q}_{M_L, \{N_\ell\}}^{(ML)} := \sum_{\ell=0}^{L} \hat{Y}_{\ell, N_\ell}, \tag{2.7}$$

with $\hat{Y}_{\ell, N_\ell}$ an estimator for $\mathbb{E}[Y_\ell]$, defined as,

18

$$\hat{Y}_{\ell,N_\ell} = \frac{1}{N_\ell}\sum_{i=1}^{N_\ell} Y_\ell^{(i)},$$

with $Y_0 = Q_{M_0}$ and $Y_\ell = Q_{M_\ell} - Q_{M_{\ell-1}}$, for each $\ell = 1,\ldots,L$, where each Y_ℓ is estimated independently with N_ℓ samples on each level.

2.2.4 Mean Square Error

For the Multi Level Monte Carlo we again have contributions to the error from both the sampling error and from the discretisation error. To quantify this we look at the expected mean square error of the Multi Level estimator, $\hat{Q}_{M_L,\{N_\ell\}}^{(ML)}$, which is defined as,

$$\mathbb{E}\left[\left(\hat{Q}_{M_L,\{N_\ell\}}^{(ML)} - \mathbb{E}[Q]\right)^2\right]. \tag{2.8}$$

As with the Standard MSE we can expand this, see Appendix A.2, to illustrate the two sources of error,

$$\mathbb{E}\left[\left(\hat{Q}_{M_L,\{N_\ell\}}^{(ML)} - \mathbb{E}[Q]\right)^2\right] \;=\; \underbrace{\mathbb{V}\left[\hat{Q}_{M_L,\{N_\ell\}}^{(ML)}\right]}_{\text{sampling error}} + \underbrace{\left(\mathbb{E}[Q_{M_L} - Q]\right)^2}_{\text{discretisation error}}. \tag{2.9}$$

Our approximations, $\hat{Y}_{\ell,N_\ell}$, to the expected value $\mathbb{E}[Y_\ell]$ are estimated independently, using different random numbers for each of the estimators $\hat{Y}_{\ell,N_\ell}$, $\ell = 0,\ldots,L$, this independence allows us to write the variance of the Multi Level estimator as,

$$\mathbb{V}\left[\hat{Q}_{M_L,\{N_\ell\}}^{(ML)}\right] = \sum_{\ell=0}^{L}\mathbb{V}[\hat{Y}_{\ell,N_\ell}] = \sum_{\ell=0}^{L} N_\ell^{-1}\mathbb{V}[Y_\ell], \tag{2.10}$$

i.e. the sum of the of the variances of differences on each level scaled by the number of samples on that level, thus we can write the mean square error of the Multi Level estimator as,

$$\mathbb{E}\left[\left(\hat{Q}_{M_L,\{N_\ell\}}^{(ML)} - \mathbb{E}[Q]\right)^2\right] \;=\; \underbrace{\sum_{\ell=0}^{L} N_\ell^{-1}\mathbb{V}[Y_\ell]}_{\text{sampling error}} + \underbrace{\left(\mathbb{E}[Q_{M_L} - Q]\right)^2}_{\text{discretisation error}}. \tag{2.11}$$

As with the one level method this gives a contribution from the sampling error and that from the discretisation error. The contribution of the discretisation error remains unchanged from the Standard Monte Carlo, and the contribution is that from the finest

level of the Multi Level method. What changes, therefore, is the sampling error of the estimator $\hat{Q}_{M_L, \{N_\ell\}}^{(ML)}$. Since on each level we use independent random numbers, the sampling error is given by the sum of the sampling error on each level, i.e. the sum of the variance of the differences on each level scaled by the number of samples on each level.

Since the sampling error is dependent on the variance, $\mathbb{V}[Y_\ell]$, of the differences, Y_ℓ, the decay rate of this variance is crucial to the success of the Multi Level method. As with the Standard Monte Carlo we split the the error between the discretisation error and the sampling error, requiring that each be bounded by $\varepsilon^2/2$ and again we note that this may not be the optimal split.

So we need each of the terms in the sum $\sum_{\ell=0}^{L} N_\ell^{-1} \mathbb{V}[Y_\ell]$ to be small enough such that that sum is bounded by $\varepsilon^2/2$. As such we require that the variance of the differences, Y_ℓ, tends to zero as we decrease the size of our time step so that fewer samples are required on the finer, more expensive levels in order to bound the sampling error term. In other words we require,

$$\mathbb{V}[Y_\ell] = \mathbb{V}[Q_{M_\ell} - Q_{M_{\ell-1}}] \to 0 \text{ as } h_\ell \to 0.$$

In addition we require that $\mathbb{V}[Y_\ell] < \mathbb{V}[Q_{M_\ell}]$, i.e. the variance of the difference is less than the variance of the quantity itself. If, instead, we had $\mathbb{V}[Y_\ell] > \mathbb{V}[Q_{M_\ell}]$ then naturally the Standard Monte Carlo would be the cheaper method.

As a result the rate at which this variance decays is vital to how much we can gain from the application of Multi Level Monte Carlo. In particular the rate of decay will determine how the computational cost is distributed across the different levels, with the cost either being focussed on the coarser level, in which case we see the most gain, or the cost will be spread evenly across all levels, or finally, in the case of the slowest convergence of the variance of the differences, the cost will be concentrated on the finer more expensive levels, in which case we will see the least gain over the Standard Monte Carlo.

From [9] we have a formula for the optimal number of samples, N_ℓ, required on each level such that the cost is minimised for a constant sampling error,

$$N_\ell = \left\lceil 2\varepsilon^{-2} \sqrt{V_\ell^{(0)} h_\ell} \left(\sum_{\ell=0}^{L} \sqrt{V_\ell^{(0)}/h_\ell} \right) \right\rceil \tag{2.12}$$

here $V_\ell^{(0)} \approx \mathbb{V}[Y_\ell]$ is an estimator of the variance of the differences, $Y_\ell = Q_{M_\ell} - Q_{M_{\ell-1}}$, computed with an initial number of samples $N_\ell^{(0)}$, on each level ℓ, using the formula,

$$V_\ell^{(0)} = \frac{1}{N_\ell^{(0)}} \left(\sum_{i=1}^{N_\ell^{(0)}} \left(Y_\ell^{(i)} \right)^2 \right) - \left(E_\ell^{(0)} \right)^2 \tag{2.13}$$

with,

$$E_\ell^{(0)} = \frac{1}{N_\ell^{(0)}} \sum_{i=1}^{N_\ell^{(0)}} \left(Y_\ell^{(i)} \right) \tag{2.14}$$

where $E_\ell^{(0)} = \mathbb{E}[Y_\ell]$ is the observed expected value of Y_ℓ and $N_\ell^{(0)}$ is arbitrary but chosen sufficiently large such that $V_\ell^{(0)}$ is a good approximation of the variance, $\mathbb{V}[Y_\ell]$.

An optimal algorithm which incorporates the computation of these initial samples into the calculation of the final solution is given in [9] and discussed further in Section 3.5.1.

2.2.5 Complexity Theorem

The following is the complexity theorem, proved in [9], given for the specific application to atmospheric dispersion modelling. The theorem gives us conditions on the numerical regime for which the Multi Level algorithm gives an improvement over the Standard Monte Carlo.

Theorem 2.2.1. *Let* $Q : \mathbb{R}^2 \to \mathbb{R}$ *denote a functional of* $U(T)$ *and* $X(T)$ *where* U *and* X *are the solutions of the SDE,*

$$\mathrm{d}U = a(U, X, t)\mathrm{d}t + b(X, t)\mathrm{d}W(t)$$
$$\tag{2.15}$$
$$\mathrm{d}X = [v + U]\mathrm{d}t$$

for a given Brownian path $W(t)$*, where* v *is the mean velocity field. Let* Q_{M_ℓ} *denote the corresponding approximation using a numerical discretisation with timestep* $h_\ell = M^{-\ell}T$*.*

If there exist independent estimators $\hat{Y}_\ell$ *based on* N_ℓ *Monte Carlo samples, and positive constants* $\alpha \geq \frac{1}{2}$*,* β*, such that*

(i) $\mathbb{E}[Q_{M_\ell} - Q] = \mathcal{O}(h_\ell^\alpha)$

(ii) $\mathbb{E}[Y_\ell] = \begin{cases} \mathbb{E}[Q_{M_0}] & \ell = 0 \\ \mathbb{E}[Q_{M_\ell} - Q_{M_{\ell-1}}] & \ell > 0 \end{cases}$

(iii) $\mathbb{V}[Y_\ell] = \mathcal{O}(N_\ell^{-1} h_\ell^\beta)$

(iv) C_ℓ, the computational complexity (or cost) of $\hat{Y}_\ell$, is such that $C_\ell = \mathcal{O}(N_\ell h_\ell^{-1})$,

then for any $\varepsilon < e^{-1}$, there are values L and N_ℓ for which the multi level estimator

$$\hat{Q}_{M_L}^{(ML)} = \sum_{\ell=0}^{L} \hat{Y}_\ell$$

has a MSE with bound

$$MSE \equiv \mathbb{E}\left[\left(\hat{Q}_{M_L}^{(ML)} - \mathbb{E}[Q]\right)^2\right] < \varepsilon^2$$

with a computational complexity (or cost) C with bound,

$$C = \begin{cases} \mathcal{O}(\varepsilon^{-2}), & \beta > 1 \\ \mathcal{O}(\varepsilon^{-2}(\log \varepsilon)^2), & \beta = 1 \\ \mathcal{O}(\varepsilon^{-2-(1-\beta)/\alpha}), & 0 < \beta < 1 \end{cases}$$

For $\beta > 1$ we see the most gain in computational cost for the Multi Level Method over the standard; in this case the bulk of the computational cost is focussed on the coarser levels. If instead we have that $\beta = 1$, then the cost is distributed evenly across all levels and we still see gains over the Standard method although these are likely to be less significant than for $\beta > 1$. Finally for $0 < \beta < 1$, the slow convergence of the variance of the differences means that the computational cost is focussed on the finest, most expensive levels as the slower decay rate of the variance will in general mean a larger variance on the finer levels which will require more samples to reduce the sampling error. How much we gain, in this case, from the implementation of Multi Level Monte Carlo, will depend on the value of α.

The complexity theorem outlines good potential for improvement over the Standard Monte Carlo. We consider again the example studied for the Standard Monte Carlo complexity theorem in Section 2.1.5, wherein we assumed that the time stepping method had discretisation error which converged with $\mathcal{O}(h)$, i.e. $\alpha = 1$. In this case we had that the computational complexity for Standard Method was $\mathcal{O}(\varepsilon^{-3})$.

In contrast, for the Multi Level algorithm, assuming we had a problem for which the variance converged with $\mathcal{O}(h)$, i.e. $\beta = 1$ in the complexity theorem, we would see that the cost would be $\mathcal{O}(\varepsilon^{-2}(\log \varepsilon)^2)$, which already demonstrates an improvement over the Standard.

However if we were to consider a problem in which the variance converged with $\mathcal{O}(h^2)$, we could see this cost improved to $\mathcal{O}(\varepsilon^{-2})$. In this case we have $\beta = 2$ in the complexity theorem, and as such to gain one decimal place of accuracy, the cost of the

method is only 100 times more expensive, compared to 1000 times more expensive for the Standard Monte Carlo.

It is therefore crucial for our application to atmospheric dispersion modelling that we study these rates of convergence in order to identify the magnitude of the potential improvements in terms of computational cost, however this indicates that even if our variance decays linearly there is potential for a significant improvement over the Standard Monte Carlo Method. The decay rates of the variance are studied in Section 3.5 for a simplified problem and Section 4.5 for a realistic model problem.

Chapter 3

Simplified Problem

3.1 Simplified Model

To establish whether the gains seen in [9], in the context of mathematical finance, have the potential to be replicated in the application to atmospheric dispersion modelling, we first look at a simplified, but nevertheless relevant, 1-dimensional model problem.

The simplified model we study describes homogeneous turbulence and is used by the Met Office to make predictions on longer timescales, since its reduced computational cost allows for longer term model problems to be run on operational timescales.

Another source of motivation for first studying this simplified problem is that it allows us to apply various tools of mathematical analysis to the problem which would otherwise prove challenging to apply to a full model with non-linear coefficients.

Our simplified model problem is given by,

$$
\begin{aligned}
\mathrm{d}U(t) &= -\frac{U(t)}{\tau}\mathrm{d}t + \sqrt{\frac{2\sigma_U^2}{\tau}}\mathrm{d}W(t) \\
\mathrm{d}X(t) &= U(t)\mathrm{d}t,
\end{aligned}
\tag{3.1}
$$

with σ_U^2 and τ both constants, where σ_U^2 corresponds to the velocity variance and τ is the time scale over which the velocity decorrelates, in other words, the autocorrelation of the velocity of the particles between time t and $t + \delta t$ decays with $\exp(-\frac{\delta t}{\tau})$ [18].

In this model we have set $v(X, t) = 0$, so our mean velocity is zero, i.e. there is no mean wind acting on our particle, simply the turbulence. Since we have a 1-dimensional model, $X(t)$ is simply the vertical height of the particle above the ground at time t and $U(t)$ is the vertical deviation of the velocity from the mean at time t.

The Forward Euler discretisation for this simplified model is given by,

$$U_{n+1} = (1 - \frac{h}{\tau})U_n + \sqrt{\frac{2\sigma_U^2 h}{\tau}}\Delta W_n$$

$$(3.2)$$

$$X_{n+1} = X_n + U_n h,$$

again $U_n \approx U(t_n)$ is our Euler approximation to U at the n-th time step, $X_n \approx X(t_n)$ the approximation of X at the n-th time step and $\Delta W_n \sim N(0,1)$ are independent and give the increments in the random walk.

3.2   Quantity of Interest

As we have previously discussed, our quantity of interest is generally the expected value of some functional of the solution of the SDE. In the application to atmospheric dispersion modelling the Met Office produces concentration fields in order to illustrate the spread of a pollutant. The concentration fields are produced by approximating the probability of a particle being in a particular subsection or "box" in the atmosphere at some time T.
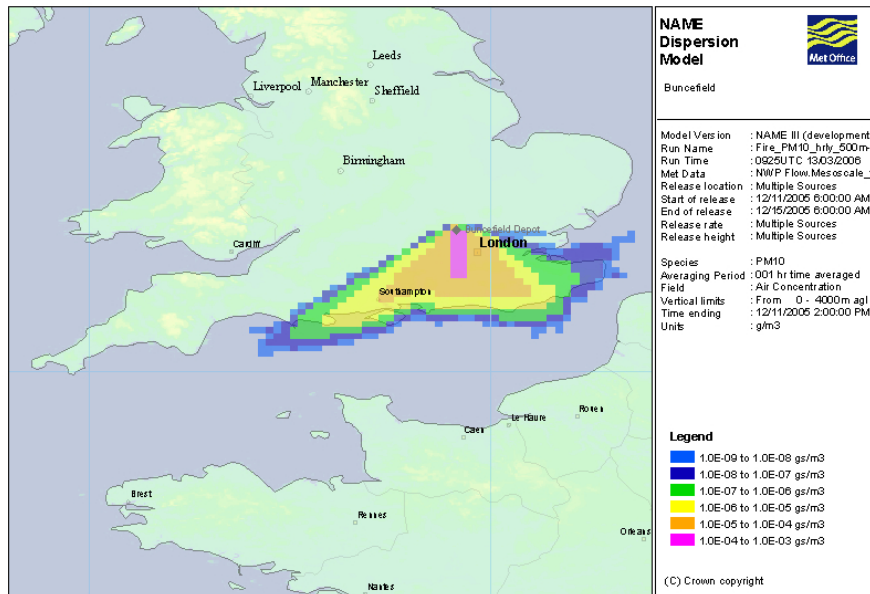

Figure 3-1: Buncefield NAME Concentration Field Plot, Copyright Met Office, used with permission, [23]

Figure 3-1 shows an example of a typical concentration field, where the varying

colours correspond to the differing concentration of particles of the atmospheric pollutant. In particular here the plot shows the predicted spread of smoke resulting from the series of explosions that occurred at the Buncefield site in Hertfordshire on 11 December 2005.

As well as running tests for the simplified model with our quantity of interest the probability of a particle landing in a particular "box", we began by using the mean particle position, $X(T)$. These tests serve as an interesting comparison and may give scope for further improvements on the method by adapting the functional we use.

3.2.1 Indicator Function

A simple way to approximate the probability of a particle being in a particular "box", which, in 1-dimension, we refer to as the interval $[a, b]$, is to evaluate the indicator function of the final position for each of our Monte Carlo samples. Then for a sufficiently large number of samples, N, the average will approximate the probability of a particle landing in that interval. The indicator function on $[a, b]$ is defined as,

$$\chi_{a,b}^{(i)} = \begin{cases} 1 & \text{if } X_M^{(i)}(T) \in [a, b] \\ 0 & \text{otherwise.} \end{cases}$$

This gives 1 if the i-th particle, i.e. the i-th sample of X_M, is inside our specified interval at time T and 0 otherwise. This is the functional that the Met Office currently use in NAME.

So the estimator for the quantity of interest, i.e. the probability, is of the form,

$$\hat{Q}_{[a,b]} = \frac{1}{N} \sum_{i=1}^{N} \chi_{a,b}^{(i)}, \tag{3.3}$$

which gives us an approximation to the probability of a particle being in $[a, b]$ at time T.

The Multi Level complexity theorem relies on having a bound on the variance of the differences, $Y_\ell = Q_{M_\ell} - Q_{M_{\ell-1}}$, i.e. we require $\mathbb{V}[Q_{M_\ell} - Q_{M_{\ell-1}}] = \mathcal{O}(h^\beta)$ for some β. In order to attain this bound on the variance it is sufficient to require Lipschitz continuity of $Q_{M_\ell}(X(T))$, which does not hold for the indicator function, as defined above, since it is not continuous and therefore not Lipschitz continuous. However for this simplified problem this bound can be obtained without Lipschitz continuity.

A continuous indicator function would however be beneficial and in particular could give faster variance reduction and as such preliminary results for a "smoothed" indicator function are given in Section 3.6

3.3 Analytic Results

As discussed previously, one of the advantages of studying this simplified, linear, 1-dimensional problem, is that it allows us to apply analytical tools which would prove more difficult or outright impossible for a more complicated test case. This analysis allows us to derive information about the distribution of our quantity of interest, thus enabling us to describe how well the method will perform for this model problem which we can then verify with numerical tests.

We first derive formulas for the expected value and variance of the solution X_M. Since in the linear case X_M is Gaussian, it is possible to deduce the entire probability density function from this, which allows us to compute information about the expected value and variance of some functional of our solution, in particular the indicator function. This will also enable us to prove the order of the discretisation error and so give us α in the complexity theorem.

We begin by recalling that the Euler Method for this simplified problem is given by,

$$U_{n+1} = (1 - \frac{h}{\tau})U_n + \sqrt{\frac{2\sigma_U^2 h}{\tau}}\Delta W_{n+1},$$

$$X_{n+1} = X_n + U_n h.$$

(3.4)

We can prove by induction, see Appendix A.4, that the solution to the Euler scheme can be written as,

$$U_M = (U_0 + \sigma_U^*\Delta W_0)\left(1 - \frac{h}{\tau}\right)^M + \sqrt{\frac{2\sigma_U^2 h}{\tau}}\sum_{k=1}^{M}\left(1 - \frac{h}{\tau}\right)^{M-k}\Delta W_k$$

$$X_M = X_0 + h\sum_{n=0}^{M-1}U_n,$$

(3.5)

here the $U_0 + \sigma_U^*\Delta W_0$ term gives the initial velocity, with $\sigma_U^*\Delta W_0$ included in line with the results given in [18]. Expanding the sum for X_M thus gives,

$$X_M = \underbrace{X_0 + U_0 h \sum_{n=0}^{M-1} \left(1 - \frac{h}{\tau}\right)^n +}_{\text{Deterministic Part}}$$

$$\underbrace{+ \left(\sigma_U^* h \sum_{n=0}^{M-1} \left(1 - \frac{h}{\tau}\right)^n\right) \Delta W_0 + h \sqrt{\frac{2\sigma_U^2 h}{\tau}} \sum_{n=0}^{M-1} \sum_{k=1}^{n} \left(1 - \frac{h}{\tau}\right)^{n-k} \Delta W_k}_{\text{Stochastic Part}}.$$

$$(3.6)$$

Here we are assuming that our velocity at time $t = 0$ already has some turbulent component, $\sigma_U^* \Delta W_0$, so our initial velocity will be $U_0 + \sigma_U^* \Delta W_0$. For the analytical results that follow we have assumed, that $\sigma_U^* = \sigma_U$, to be consistent with the results in [18], however in our numerical tests we have taken $\sigma_U^* = 0$, so the particles simply have initial velocity U_0. In future work it would be interesting to repeat this derivation for $\sigma_U^* = 0$ and indeed to repeat the numerical tests with $\sigma_U^* \neq 0$.

We see from equation (3.6) that X_M is a sum of normally distributed random variables, thus X_M is also a normally distributed random variable, with mean given by the sum of the means and variance the sum of the variances. As a result we are able to obtain approximations to both $\mathbb{E}[X_M]$ and $\mathbb{V}[X_M]$ which we can compute to any required order of accuracy in $\frac{h}{\tau}$. For our analysis, however, we generally only require first order accuracy.

This gives us sufficient information about the distribution of X_M to be able to determine the probability density function (pdf) of X_M and as a result we are able to compute the expected value and variance of some functional of X_M, again to the required order of accuracy in $\frac{h}{\tau}$, in particular we look at the indicator function on $[a, b]$, $\chi_{a,b}(X_M)$.

We begin by looking at the expected value of X_M, which we do so by expanding the sums and taking various Taylor expansions of our quantities, see Appendix A.5, from this we deduce that the expected value can be written as,

$$\mathbb{E}[X_M] = \mu_0 + \frac{h}{\tau}.\delta\mu + \mathcal{O}\left(\left(\frac{h}{\tau}\right)^2\right) \tag{3.7}$$

where, constants μ_0 and $\delta\mu$ are given by,

$$\mu_0 \quad = \quad X_0 + U_0\tau\left(1 - \exp\left[\frac{-T}{\tau}\right]\right)$$

$$\delta\mu \quad = \quad \frac{U_0 T}{2}\exp\left[\frac{-T}{\tau}\right]$$

Similarly for the variance, expanding the sums and taking Taylor expansions gives,

$$\mathbb{V}[X_M] = \sigma_0^2 + \delta\sigma^2.\frac{h}{\tau} + \mathcal{O}\left(\left(\frac{h}{\tau}\right)^2\right) \tag{3.8}$$

where constants σ_0^2 and $\delta\sigma^2$ are given by,

$$\sigma_0^2 \quad = \quad 2\sigma_U^2\tau^2\left(\frac{T}{\tau} - 1 + \exp\left[\frac{-T}{\tau}\right]\right)$$

$$\delta\sigma^2 \quad = \quad \sigma_U^2\tau^2\left(\frac{1}{2} - \frac{T}{\tau}\exp\left[\frac{-T}{\tau}\right] - \frac{1}{2}\exp\left[\frac{-2T}{\tau}\right]\right)$$

Since we know that X_M is the sum of normally distributed random variables and thus itself a normally distributed random variable we know information about its probability density function. A normally distributed random variable, $Y \sim N(\mu, \sigma^2)$ has a pdf given by,

$$P(Y) = \frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(-\frac{(Y-\mu)^2}{2\sigma^2}\right)$$

From formulas (3.7) and (3.8) we have the mean, μ, and variance, σ^2, which we can now substitute into the form for the pdf of a normally distributed random variable in order to attain the distribution of our random variable X_M. So the pdf of X_M will be given by,

$$P(X_M) = \frac{1}{\sqrt{2\pi(\sigma_0^2 + \delta\sigma^2.\frac{h}{\tau})}}\exp\left(-\frac{(X_M - \mu_0 - \frac{h}{\tau}.\delta\mu)^2}{2(\sigma_0^2 + \delta\sigma^2.\frac{h}{\tau})}\right)$$

Substituting in the formulas for μ_0, $\delta\mu$, σ_0 and $\delta\sigma^2$ as described above and Taylor-expanding in h/τ, see Appendix A.6, allows us to write the pdf as,

$$P(X_M) = \frac{1}{\sqrt{2\pi\sigma_0^2}}\exp\left(-\frac{(X_M - \mu_0)^2}{2\sigma_0^2}\right).\left(1 + (q_0 + q_1 X_M + q_2 X_M^2)\frac{h}{\tau}\right) + \mathcal{O}\left(\left(\frac{h}{\tau}\right)^2\right) \tag{3.9}$$

29

where the constants q_i, $i = 0, 1, 2$ are given by,

$$q_0 = \frac{2\mu_0\delta\mu}{2\sigma_0^2} - \frac{\mu_0^2}{2\sigma_0^2}\cdot\frac{\delta\sigma^2}{\sigma_0^2} = -\frac{1}{\sigma_0^2}\left(-\mu_0\delta\mu + \mu_0^2\frac{\delta\sigma^2}{2\sigma_0^2}\right)$$

$$q_1 = -\frac{\delta\mu}{\sigma_0^2} + \frac{\mu_0\delta\sigma^2}{\sigma_0^4} = -\frac{1}{\sigma_0^2}\left(\delta\mu - \mu_0\frac{\delta\sigma^2}{\sigma_0^2}\right)$$

$$q_2 = -\frac{\delta\sigma^2}{2\sigma_0^4}$$

The expected value, $\mathbb{E}[Y]$, of a random variable, Y, is defined as the integral with the density $P(Y)$ over the probability space, Ω, i.e.

$$\mathbb{E}[Y] := \int_\Omega P(Y)Y\,\mathrm{d}Y.$$

To this end, some functional, f, of a random variable, has expected value given by,

$$\mathbb{E}\Big[f(Y)\Big] = \int_\Omega P(Y)f(Y)\,\mathrm{d}Y.$$

So, returning to our approximation X_M, to the solution of the SDE (3.1), if we consider some functional of the solution, for example the indicator function, $\chi_{a,b} : \mathbb{R} \to \mathbb{R}$, then the expected value of this functional evaluated on our solution will be given by,

$$\mathbb{E}\Big[\chi_{a,b}(X_M)\Big] = \int_\Omega P(X_M)\chi_{a,b}(X_M)\,\mathrm{d}X_M.$$

where $P(X_M)$ is as described in equation (3.9).

Writing out the pdf explicitly gives us the following form for the expected value,

$$\mathbb{E}\Big[\chi_{a,b}(X_M)\Big] = I_0(\mu_0, \sigma_0; a, b) + \left(\sum_{k=0}^{2} q_k I_k(\mu_0, \sigma_0; a, b)\right)\cdot\frac{h}{\tau} + \mathcal{O}\left(\left(\frac{h}{\tau}\right)^2\right) \qquad (3.10)$$

where the integrals $I_i(\mu_0, \sigma_0; a, b)$, $i = 0, 1, 2$, with respect to some random variable Y, over the interval $[a, b]$ are defined as,

$$I_0(\mu_0, \sigma_0; a, b) \equiv \int_a^b N(\mu_0, \sigma_0^2; Y) \mathrm{d}Y$$

$$I_1(\mu_0, \sigma_0; a, b) \equiv \int_a^b N(\mu_0, \sigma_0^2; Y).Y \, \mathrm{d}Y \qquad (3.11)$$

$$I_2(\mu_0, \sigma_0; a, b) \equiv \int_a^b N(\mu_0, \sigma_0^2; Y).Y^2 \, \mathrm{d}Y$$

formulas for which are given in the Appendix in Section A.6 alongside the full workings.

Thus we have shown that for our random variable, X_M, we are able to compute the expected value and variance of X_M to some specified order in $\frac{h}{\tau}$ and that from this we are able to compute the probability density function, $P(X_M)$. Finally with this pdf we are then able to deduce a formula for the expected value of some functional of X_M. We would also like to be able to have a formula for the variance of this functional of X_M.

By the definition and a few simple calculations, see Appendix A.6, the variance can be written as,

$$\mathbb{V}\Big[\chi_{a,b}(X_M)\Big] = \mathbb{E}[\chi_{a,b}(X_M)]\Big(1 - \mathbb{E}[\chi_{a,b}(X_M)]\Big).$$

We now have a formula for the expected value and the variance of our quantity of interest $\chi_{a,b}(X_M)$. These results allow us to deduce the value of α in the complexity theorem for this simplified problem with the functional $\chi_{a,b}(X_M)$, in other words we can derive the rate at which the discretisation error decays with respect to the time step size h, which we will later verify numerically.

The discretisation error for our functional is defined as,

$$\mathbb{E}\Big[\chi_{a.b}(X_M) - \chi_{a,b}(X(T))\Big],$$

so by the definition of the expected value this is,

$$\mathbb{E}\Big[\chi_{a.b}(X_M) - \chi_{a,b}(X(T))\Big] = \mathbb{E}\Big[\chi_{a.b}(X_M)\Big] - \mathbb{E}\Big[\chi_{a,b}(X(T))\Big]$$

$$= \int P(X_M)\chi_{a.b}(X_M) \, \mathrm{d}X_M -$$

$$- \int P_{h=0}(X(T))\chi_{a,b}(X(T)) \, \mathrm{d}X(T),$$

31

where $P_{h=0}(X(T))$ is simply the probability density function $P(X_M)$ as defined in equation (3.9) in the continuum limit as $h \to 0$. So we can use the formula for $\mathbb{E}[\chi_{a,b}]$ from equation (3.10) which in the continuum limit for $X(T)$ will be given by,

$$\mathbb{E}\Big[\chi_{a,b}(X(T))\Big] = I_0(\mu_0, \sigma_0; a, b). \tag{3.12}$$

Substituting in the values for $\mathbb{E}\Big[\chi_{a.b}(X_M)\Big]$ and $\mathbb{E}\Big[\chi_{a,b}(X(T))\Big]$ gives,

$$\mathbb{E}\Big[\chi_{a.b}(X_M) - \chi_{a,b}(X(T))\Big] = I_0(\mu_0, \sigma_0; a, b) + \left(\sum_{k=0}^{2} q_k I_k(\mu_0, \sigma_0; a, b)\right).\frac{h}{\tau}$$

$$-I_0(\mu_0, \sigma_0; a, b)$$

$$= \left(\sum_{k=0}^{2} q_k I_k(\mu_0, \sigma_0; a, b)\right).\frac{h}{\tau} + \mathcal{O}\left(\left(\frac{h}{\tau}\right)^2\right)$$

So the zero-th order terms cancel, leaving us with terms of order $\frac{h}{\tau}$, so our discretisation error is,

$$\mathbb{E}\Big[\chi_{a.b}(X_M) - \chi_{a,b}(X(T))\Big] = \mathcal{O}\Big(\frac{h}{\tau}\Big). \tag{3.13}$$

Thus we have shown that for our quantity of interest the indicator function $\chi_{a.b}$, $\alpha = 1$ in the complexity theorem. Although the above analytical derivation is for $\sigma_U^* = \sigma_U$, the same result, as far as the dependence on $\frac{h}{\tau}$ is concerned, holds for general σ_U^*.

Further to this it can also be proved, [24], that for our quantity of interest $\chi_{a,b}$, the indicator function on $[a, b]$, the variance of the differences on consecutive levels, i.e. $\mathbb{V}[\chi_{a,b}(X_{M_\ell}) - \chi_{a,b}(X_{M_{\ell-1}})]$ is of order h and thus that $\beta = 1$ in the complexity theorem. The proof of this result is not given in this thesis however numerical results in Section 3.5 support this conclusion.

Since we now know both α and β, and we also know that the cost per sample is $\mathcal{O}(h^{-1})$, i.e. $\gamma = 1$, we may now apply the complexity theorem for this problem and see that for the Standard Monte Carlo the cost is $\mathcal{O}(\varepsilon^{-3})$, whereas for the Multi Level we are in the case where the cost is $\mathcal{O}(\varepsilon^{-2}(\log \varepsilon)^2)$ and the cost is spread evenly across all levels. These results are verified numerically in Section 3.5.

Further to this, [24], although analytical results are not presented in this thesis, we have proved that, when our quantity of interest is instead the mean particle position, $\beta = 2$ due to the first order rate of strong convergence, [17]. In which case the Multi

Level Method performs even better, with the cost of $\mathcal{O}(\varepsilon^{-2})$, compared to $\mathcal{O}(\varepsilon^{-3})$ for the Standard. Numerical results that support this conclusion are presented in Section 3.5.

3.4   Stability Analysis

We have now established that there is potential for significant gains from the application of Multi Level Monte Carlo to this problem, however in order to ensure that our approximations behave well we must also take steps to ensure that the method is stable.

It is well known that for numerical methods we are concerned not only with the accuracy of the method, but also with the stability, with the combination of the two guaranteeing the convergence of the method. The stability of the method depends on the time step size that we take, so in this section we look to obtain an upper bound on our time step size h, such that the method remains stable.

In the case of the Standard Monte Carlo Method generally we require a small time step size in order to attain the prescribed accuracy, this small time step size is often sufficient to guarantee stability also; however in the Multi Level method we must be more careful.

For the Multi Level method we are interested in computing solutions with coarser time steps in order to exploit their cheapness and are thus not concerned with the accuracy of the results on these coarser levels as this is refined on the finer levels. We do however require that the method is stable on all levels, so that the only error sources are those bias and sampling errors over which we have control.

If the method is unstable the variance of the differences $Y_\ell = Q_{M_\ell} - Q_{M_{\ell-1}}$ diverges and as such we will require a large number of samples in order to reduce the sampling error, meaning that the Multi Level method will no longer be competitive. Therefore we must ensure that our coarsest time step remains within the stability region for our numerical method.

For our simplified problem, given in (3.1), we must therefore consider the stability of the Euler Method; however we have not yet carried out this stability analysis for the full SDE. Instead, for the moment, we consider only the stability of the deterministic part, in other words the ODE given by,

$$\frac{\mathrm{d}U}{\mathrm{d}t} = -\frac{U}{\tau} \tag{3.14}$$

under the assumption that the stochastic term makes a minimal, in fact possibly stabilising, contribution to the method's stability. The stability analysis for the full SDE

33

would be an interesting topic for future work in this application.

The solution of the ODE given in (3.14) is easily shown to be,

$$U(t) = U_0 \exp\left(-\frac{t}{\tau}\right),$$

where $U_0 = U(0)$. We observe that the solution decays exponentially and thus our numerical solution must also, i.e.

$$U_n(t_n) \to 0 \quad \text{as} \quad t_n \to \infty. \tag{3.15}$$

The Forward Euler Method for this ODE is given by,

$$U_{n+1} = U_n - \frac{h}{\tau}U_n$$

$$= \left(1 - \frac{h}{\tau}\right)U_n,$$

and by induction we can prove that U_n can be written as,

$$U_n = \left(1 - \frac{h}{\tau}\right)^n U_0. \tag{3.16}$$

The condition on our numerical solution given in equation (3.15) can equivalently be written as $U_n \to 0$ as $n \to \infty$, which using our formula for U_n in equation (3.16) implies,

$$U_n = \left(1 - \frac{h}{\tau}\right)^n U_0 \to 0 \quad \text{as} \quad n \to \infty \iff \left|1 - \frac{h}{\tau}\right| < 1$$

$$\iff \frac{h}{\tau} < 2 \qquad \text{since } \frac{h}{\tau} \text{ always positive}$$

$$\implies h < 2\tau,$$

so, given τ, we now have a condition on our time step size which ensures that our method is stable, [25][26].

For the numerical tests on the simplified case we took τ to be 1, and our time step size was always less than $2\tau = 2$, consequently we always remained inside the stability region for the method and having verified by numerical tests that the additional stochastic term induces no further instability in the method, this very lax constraint on the coarsest time step posed no real restrictions to the application of Multi Level Monte Carlo. So for the simplified problem our choice of time step size, h, was concerned only

34

with the accuracy of the method; however in the realistic, full model problem the parameter τ varies with X and in Section 4.3 we examine the function $\tau(X)$ and the stability constraint this imposes on the coarsest time step for the Multi Level in order to ensure that the method is stable on each level.

3.5   Numerics

In this section we start by outlining the algorithm for both the Standard and the Multi Level Monte Carlo Methods. We then look at the results for the simplified problem, as studied in Section 3.1, examining the performance of the Multi Level Monte Carlo method for various parameter choices.

3.5.1   Algorithm for Standard Monte Carlo

Here we give a general overview of the form of the algorithm used. At present the algorithm is not as efficient as it could be, but can be easily optimised, see [9], in order to improve efficiency for operational use. These changes were not implemented at this stage for the sake of easy debugging and manipulation.

The Standard Monte Carlo algorithm used to produce the results in the following sections consists of four main stages,

1. Initial Run

2. Set-up Phase

3. Calculation Phase

4. Data Processing Phase

Initial Run

In an operational environment the initial run would be built into the algorithm, however currently the initial run is computed manually. This stage allows us to determine a reasonable approximation to the expected value and variance of our particular quantity of interest for several different time step sizes. We then use these quantities to determine the time step size required to bound the bias term and compute the number of Monte Carlo samples necessary to bound the sampling error.

For this problem we are interested in computing the probability of a particle landing in a specified interval at time T. It is possible that this quantity will be small, for example at the edge of a plume, where there are naturally fewer particles, so the

probability of a particle being there at time T is low. If our required accuracy is larger than this quantity our numerical solution will naturally be inaccurate and so we instead look at the relative error. As such we require that our mean square error bound, as detailed in Section 2.1.4, is scaled by the size of our quantity of interest, i.e. by $|\mathbb{E}[Q]|$ (our initial run gives us an approximate value for $\mathbb{E}[Q]$). Thus our mean square error bound becomes,

$$\mathbb{E}\left[\left(\hat{Q}_{M_L,N} - \mathbb{E}[Q]\right)^2\right] \leq \left(\varepsilon|\mathbb{E}[Q]|\right)^2$$

As before we split the error between the discretisation error and the sampling error and so choose our time step size h_L such that the discretisation error is bounded by $\varepsilon|\mathbb{E}[Q]|$, i.e.

$$\mathbb{E}[Q_{M_L} - Q] \leq \frac{\varepsilon|\mathbb{E}[Q]|}{\sqrt{2}}$$

Similarly we choose the number of samples, N, such that it is bounded by $\left(\varepsilon|\mathbb{E}[Q]|\right)^2/2$, i.e. we choose N such that,

$$N \geq \frac{2\mathbb{V}[Q]}{(\varepsilon|\mathbb{E}[Q]|)^2}$$

Set-up Phase

The Set-up Phase then simply consists of initialising variables for use in the main part of the program.

Calculation Phase

The Calculation Phase is where the majority of the work takes places, for each of the N Monte Carlo samples the Euler loop computes an approximation to X and U, evaluating our functional at the final time step for each Monte Carlo sample and storing them.

Data Processing Phase

The final Data Processing Stage uses the stored data to compute our final Monte Carlo approximation to the quantity of interest, i.e.

$$\hat{Q}_{M_L,N} = \frac{1}{N} \sum_{i=1}^{N} Q_{M_L}^{(i)}$$

36

3.5.2 Algorithm for Multi Level Monte Carlo

The Multi Level Monte Carlo algorithm is similar in structure to that of the Standard Monte Carlo, with one additional stage,

1. Initial Run

2. Set-up Phase

3. Pre-Calculation Phase

4. Calculation Phase

5. Data Processing Phase

Initial Run

The initial run here is as described for the Standard Monte Carlo algorithm and it is only necessary to do this once in order to obtain relevant data for both the Standard and Multi Level Monte Carlo algorithms. For the Multi Level it provides the finest level for the method, the coarsest level is determined by the stability constraints as described in Section 3.4 and again this constraint could also be incorporated into the main program in an operational environment. The number of samples for the Standard Monte Carlo serves as a rough guide for the number of samples required for the Pre-Calculation Phase in which we determine, by the formula in [9], equation (2.12), how many samples are required on each level.

Set-up Phase

The Set-up Phase then simply consists of initialising variables for use in the main part of the program, for both the Pre-Calculation and Calculation Phases.

Pre-Calculation Phase

The Pre-Calculation Phase runs the Multi Level method for each level, ℓ, for an initial number of samples, $N_\ell^{(0)}$, which is chosen arbitrarily but sufficiently large such that we obtain a good approximation to the variance $V_\ell^{(0)}$ and from this we compute the required number of samples on each level for the main Calculation Phase, using Mike Giles formula (2.12). Since we are now using the relative error, however, the formula becomes,

$$N_\ell = \left\lceil 2(\varepsilon|\mathbb{E}[Q]|)^{-2}\sqrt{V_\ell^{(0)}h_\ell}\left(\sum_{\ell=0}^{L}\sqrt{V_\ell^{(0)}/h_\ell}\right)\right\rceil \tag{3.17}$$

37

In this particular implementation of the method, all the samples taken in this Pre-Calculation Phase are then discarded, thus this phase contributes greatly to the run time of the algorithm. In an optimised code, however, these samples would not be discarded. Instead the algorithm would be run for an initial number of samples, $N_\ell^{(0)}$, from which we would compute an approximation to the variance on each level, $V_\ell^{(0)}$, allowing us to compute the required number of samples on each level, $N_\ell^{(1)}$, using the formula in (3.17).

Then for those levels, ℓ, where $N_\ell^{(1)} > N_\ell^{(0)}$, we retain the information obtained from the initial $N_\ell^{(0)}$ samples and compute additional samples such that we then have $N_\ell^{(1)}$ samples. Then we use the new variance, $V_\ell^{(1)}$, of the $N_\ell^{(1)}$ samples, to compute $N_\ell^{(2)}$ using the formula in (3.17), testing again whether $N_\ell^{(2)} > N_\ell^{(1)}$ and repeating this process until we have computed sufficient samples on all levels.

Calculation Phase

The main stage is again the Calculation Phase in which the Euler loop is run for N_ℓ samples on each level, approximating the quantity of interest by computing an initial approximation on the coarsest level and refining it by the "corrections" on the subsequent finer levels, these are the Y_ℓ's as described in Section 2.2.2.

Data Processing Phase

The final Data Processing Phase computes the sum of the approximations on each level to give the Multi Level approximation to the quantity of interest.

$$\hat{Q}_{M_L,\{N_\ell\}}^{(ML)} = \sum_{\ell=0}^{L} \hat{Y}_{\ell,N_\ell}$$

3.5.3 Quantity of Interest: Mean Particle Position

To begin with we examine the simplified but relevant one dimensional model with constant coefficients, as described in Section 3.1, with $\sigma = 1/\sqrt{2}$, $\tau = 1$ and initial conditions $X_0 = 0$, $U_0 = 1$. Our initial tests take the quantity of interest as X, at time T. Physically the expected value of X is the average particle position and hence provides an idea of how far the cloud has travelled within the time T.

We assume that $\sigma_U = u^*/\sqrt{2}$ with u^* the friction velocity. We measure X in units of $H = 1$km, u in units of $u^* = 0.2\text{ms}^{-1}$ and T in units of τ, under the assumption that $u^*\tau = H$ (i.e. the decorrelation length is of the order of the height of the atmosphere). The value $T = 1$ corresponds to approximately 1.4 hours. Then the simplified equation

for which we give numerical results is,

$$dU = -U dt + dW$$

$$dX = U dt.$$

(3.18)

In order to establish the potential gains that Multi Level Monte Carlo may bring over the Standard Monte Carlo it is crucial to determine the convergence rates of the expected value of differences, i.e. $\mathbb{E}[Q_{h_\ell} - Q_{h_{\ell-1}}]$ and most importantly of the variance of the difference, $\mathbb{V}[Q_{h_\ell} - Q_{h_{\ell-1}}]$, which will enable us to apply the complexity theorems in both the case of the Standard and Multi Level Methods.

(a) Expected Value   (b) Variance
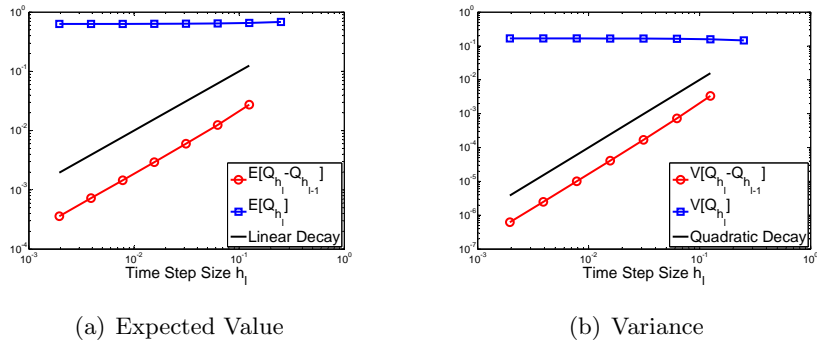
Figure 3-2: Expected Value and Variance for QoI $=X$, $T = 1$, $\varepsilon = 0.01$

Figure 3-2 shows, on the left, the expected value of the differences in the Multi Level method (in red), i.e. $\mathbb{E}[Q_{M_\ell} - Q_{M_{\ell-1}}]$, as well as the expected value, $\mathbb{E}[Q_{M_\ell}]$, for the Standard Monte Carlo (in blue) computed for each time step used in the Multi Level Method. On the right we have plots of the variance of the differences, in the Multi Level method (in red) i.e. $\mathbb{V}[Q_{M_\ell} - Q_{M_{\ell-1}}]$, as well as the variance, $\mathbb{V}[Q_{M_\ell}]$, for the Standard Method (in blue) computed for each time step used in the Multi Level Method.

We see that the expected value convergences with $\mathcal{O}(h)$, corresponding to $\alpha = 1$ in the Standard and Multi Level Complexity theorems. In addition we see that the convergence rate of the variance of the differences is $\mathcal{O}(h^2)$, so that we have $\beta = 2$ in the Complexity theorem; this has also been proven analytically, [24], however the proof is not included in this thesis. We can conclude from this theorem that the computational complexity for the Multi Level Method is $\mathcal{O}(\varepsilon^{-2})$ compared to $\mathcal{O}(\varepsilon^{-3})$ for Standard Monte Carlo (since the cost per sample of the Euler method is $\mathcal{O}(h^{-1})$, which implies $\gamma = 1$, in the Standard Monte Carlo complexity theorem).
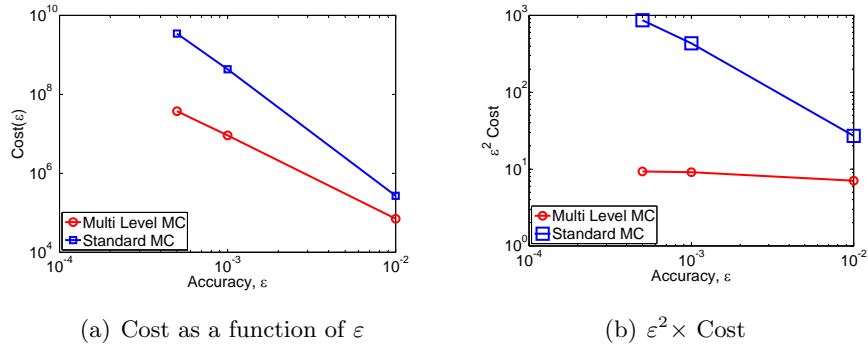
(a) Cost as a function of ε (b) $\varepsilon^2 \times$ Cost

Figure 3-3: Theoretical Cost of Multi Level MC vs Standard MC for QoI $=X$, $T = 1$, Where Unit of Cost is the Cost per Particle per Time Step

In figure 3-3 we look at the cost of Multi Level Monte Carlo versus that of the Standard Method with our quantity of interest again the mean particle position, X, at time $T = 1$. Both of these plots use theoretical cost computations which are based on the total number of time steps, computed using the N_ℓ's obtained from the formula in (3.17), the unit of cost being the cost per particle per time step.

For the Multi Level Method we assume that the cost of computing the difference $\mathbb{E}[Q_{M_\ell} - Q_{M_{\ell-1}}]$ is 1.5 times that of computing $\mathbb{E}[Q_{M_\ell}]$ since the quantity computed with the coarser time step will have half as many Euler time steps as the finer (with steps incremented in powers of 2).

Figure 3-3(a) plots the theoretical cost for both the Standard (blue) and Multi Level (red) Monte Carlo methods, for accuracies $\varepsilon = 10^{-2}, 10^{-3}$ and 5×10^{-4}. We see that even for $\varepsilon = 10^{-2}$ that the Multi Level Method beats the Standard Method by a significant amount.

Figure 3-3(b) plots $\varepsilon^2 \times$ Cost for both the Standard and Multi Level Monte Carlo, verifying the conclusion we drew from the Complexity theorem, that the Multi Level has computational cost of $\mathcal{O}(\varepsilon^{-2})$. In addition we see that, since $\varepsilon^2 \times$ Cost is clearly not constant in the Standard case, the computational complexity is of order higher than ε^{-2}, namely $\mathcal{O}(\varepsilon^{-3})$.
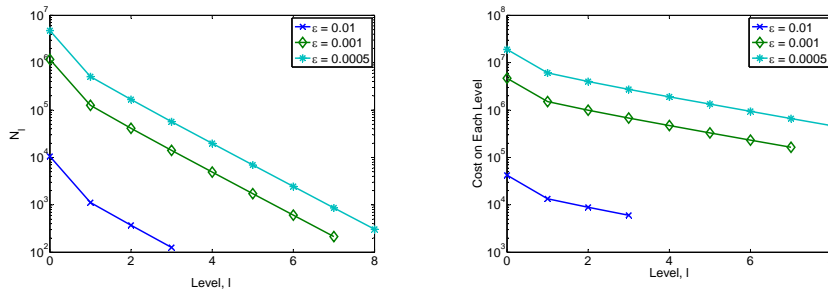
(a) Number of Samples on Each Level   (b) Cost on Each Level of Multi Level MC

Figure 3-4: Number of Samples computed with formula (2.12) and Cost on each level of Multi Level MC for varying ε, $T = 1$

Figure 3-4(a) plots the number of samples required on each level of the Multi Level algorithm for several different accuracy tolerances, ε. We see that the fast convergence rate of the variance of the differences means that on the finest, most expensive levels only around 100 samples are required for $\varepsilon = 10^{-2}$ and less than 1000 samples for $\varepsilon = 5 \times 10^{-4}$, compared to the Standard Method which requires approximately 8000 samples for $\varepsilon = 10^{-2}$ and more than 3,000,000 samples for $\varepsilon = 5 \times 10^{-4}$. So in this case the savings on the finest level clearly outweighs the burden of the additional coarse levels, making Multi Level very competitive with the quantity of interest being the mean particle position.

We also note from figure 3-4(b) how the cost is distributed on each level, we see that the cost here is clearly focussed on the coarser levels, which we recall from the complexity theorem is consistent with being in the case where $\beta = 2$ and the computational complexity is $\mathcal{O}(\varepsilon^{-2})$.

The following table demonstrates how significant a speed-up we can achieve over the Standard Monte Carlo for this simplified model problem with our quantity of interest the mean particle position, X.

ε	Standard MC Cost	Multi Level MC Cost	Speed-up
10^{-2}	2.70E+5	7.06E+4	3.82
10^{-3}	4.32E+8	9.08E+6	47.52
5×10^{-4}	3.45E+9	3.72E+7	92.85

Table 3.1: Speed-up of Multi Level MC over Standard MC, for QoI $= X$, $T = 1$

Table 3.1 shows the theoretical cost of both the Standard and Multi Level methods, where the cost is in units of the cost per sample per time step. We see that even for

accuracy $\varepsilon = 10^{-2}$, we observe a speed-up of nearly 4, and for $\varepsilon = 5 \times 10^{-4}$ we have a speed-up of nearly 100.

This section shows that with the quantity of interest the mean particle position at time T, the results look very promising with the Multi Level clearly outperforming the Standard even for our largest accuracy tolerance. The success in this case is largely down to the fast rate at which the variance of the difference converges, we now look to see how this changes when we vary the quantity of interest.

3.5.4 Quantity of Interest: Concentration

In this section we again look at the simplified, one dimensional model problem with constant coefficients, $\sigma = 1/\sqrt{2}$, $\tau = 1$ and initial conditions $X_0 = 0$, $U_0 = 1$. However this time we look at the probability of a particle landing in a particular interval, $[a, b]$, at time T. For this case we look at varying both the end time T as well as the interval in which we are interested, $[a, b]$ for varying accuracy tolerances, ε.

It is important to note here, that since some of the probabilities and thus our quantities of interest can be small, and in particular of the order of ε, in order to obtain good approximations to the solutions of these problems the results use bounds based on the relative error, as discussed in Section 3.5.1.
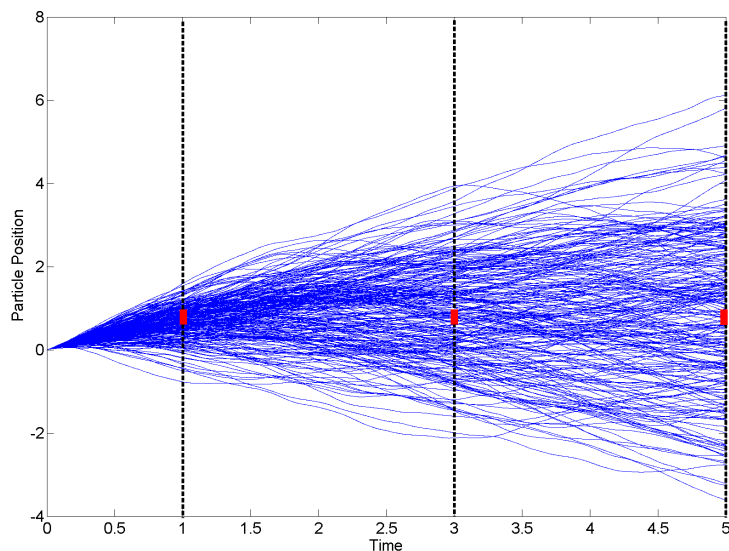

Figure 3-5: Plot Showing Trajectories of 250 Independent Particles

Figure 3-5 shows the spread of the plume for simplified model over time, the black lines indicate the points at which we chose to stop our model, $T = 1, 3, 5$ and the red

boxes show the position of the boxes in which we approximate the concentration of particles.

Varying ε

In this section we look at results for a fixed end time $T = 1$, and a fixed interval $[a, b] = [0.46, 0.63]$, which lies roughly in the centre of the plume, so particles will have a generally higher probability of landing in this interval.
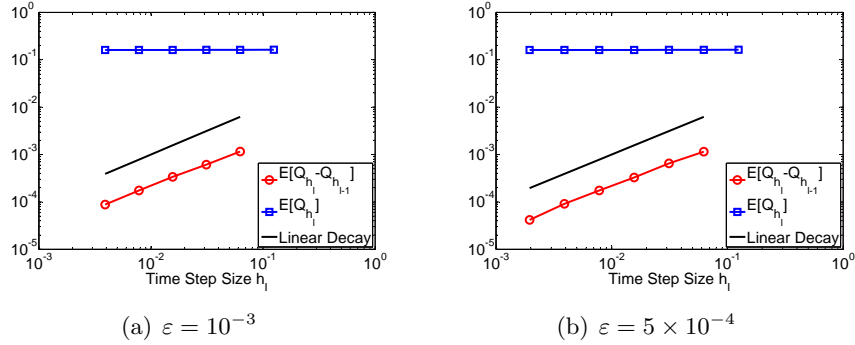

(a) $\varepsilon = 10^{-3}$ (b) $\varepsilon = 5 \times 10^{-4}$

Figure 3-6: Expected Value for varying ε, with $T = 1$, QoI $= \chi_{a,b}$, $[a, b] = [0.46, 0.63]$

Figure 3-6 shows the expected value of the differences in Multi Level method (in red), i.e. $\mathbb{E}[Q_{M_\ell} - Q_{M_{\ell-1}}]$, as well as the expected value, $\mathbb{E}[Q_{M_\ell}]$ for the Standard Monte Carlo (in blue) computed for each time step used in the Multi Level Method. The left hand plot is computed for accuracy tolerance $\varepsilon = 10^{-3}$ and the right hand for $\varepsilon = 5 \times 10^{-4}$. The coarsest level due to stability constraints is $h_0 = 2^{-3}$, we see that for the more accurate approximation, $\varepsilon = 5 \times 10^{-4}$ we need an additional level compared to that for $\varepsilon = 10^{-3}$, in order to bound the bias error. So the finest level for $\varepsilon = 10^{-3}$ is $h_L = 2^{-8}$ and for $\varepsilon = 5 \times 10^{-4}$ it is $h_L = 2^{-9}$. As with the quantity of interest the mean particle position, X, the convergence rate of the expected value of the differences is $\mathcal{O}(h)$ and so we are again in the case where $\alpha = 1$ in the complexity theorems.
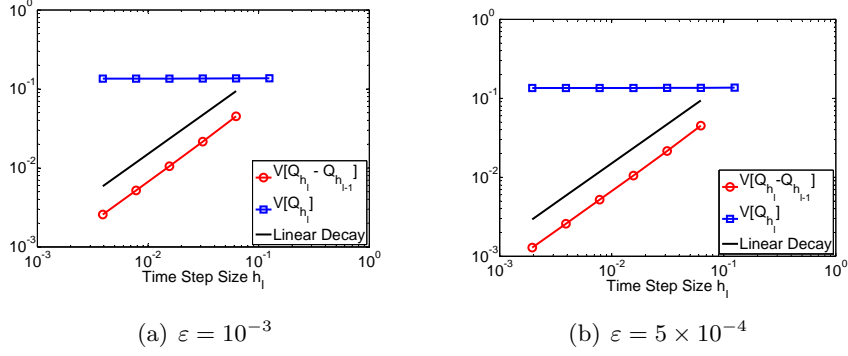
(a) $\varepsilon = 10^{-3}$ (b) $\varepsilon = 5 \times 10^{-4}$

Figure 3-7: Variance Plots for varying ε, with $T = 1$, QoI $= \chi_{a,b}$, $[a, b] = [0.46, 0.63]$

Figure 3-7 shows the variance of the differences in Multi Level method (in red), i.e. $\mathbb{V}[Q_{M_\ell} - Q_{M_{\ell-1}}]$, as well as the expected value, $\mathbb{V}[Q_{M_\ell}]$ for the Standard Monte Carlo (in blue) computed for each time step used in the Multi Level Method. Crucially we note that for the probability as our quantity of interest the convergence rate of the variance of the differences is no longer quadratic, instead the variance is $\mathcal{O}(h)$. Therefore $\beta = 1$ in the complexity theorem, which agrees with the analytical results discussed above so the complexity of the method is, in this case, $\mathcal{O}(\varepsilon^{-2}(\log \varepsilon)^2)$ and the cost is distributed evenly across all levels.
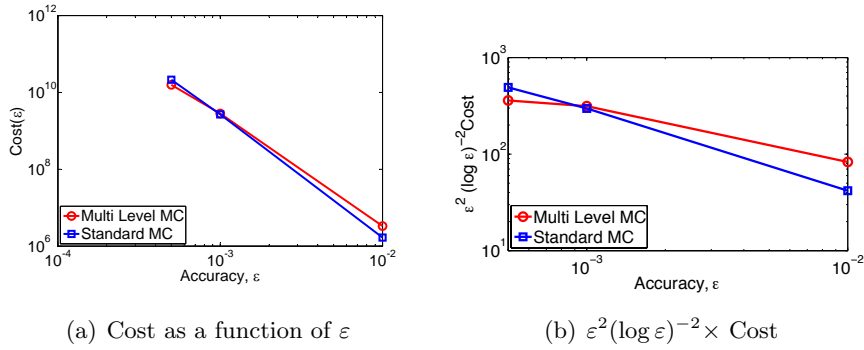

(a) Cost as a function of ε (b) $\varepsilon^2 (\log \varepsilon)^{-2} \times$ Cost

Figure 3-8: Cost of Standard MC vs Multi Level MC, with $T = 1$, QoI $= \chi_{a,b}$, $[a, b] = [0.46, 0.63]$

Figure 3-8(a) plots the theoretical cost for both the Standard and Multi Level Monte Carlo methods, for accuracies $\varepsilon = 10^{-2}, 10^{-3}$ and 5×10^{-4}, again based on the total number of time steps, using the N_ℓ's obtained by the formula in equation (3.17), with the cost measured in units of the cost per sample per time step.

We see here that for $\varepsilon = 10^{-2}$ and $\varepsilon = 10^{-3}$ the Standard Method beats the Multi Level. To see why, we refer back to the results for the mean particle position in figure

44

3-2 and we recall that the decay rate of the variance is fast; indeed much faster than that of the probability, in figure 3-7. As such, when our quantity of interest is the mean particle position, on the finest level of the Multi Level Method the variance is small, of order 10^{-6} whereas for the probability, the variance on the finest level of the Multi Level is of order 10^{-3}. Consequently, for the probability, we require more samples on the finer, more expensive levels in order to bound the larger variance term in the sampling error; thus increasing the cost of the Multi Level Method for this problem compared to that for the mean particle position.

Figure 3-8(b) plots $\varepsilon^2(\log \varepsilon)^{-2} \times$Cost for both the Standard and Multi Level Monte Carlo, here however it is not so clear that the cost is as we would expect from the complexity theorem, i.e. $\mathcal{O}(\varepsilon^{-2}(\log \varepsilon)^2)$. This may be due to an anomalous result for $\varepsilon = 10^{-2}$, for which the Multi Level Method limited to 2 levels, since it did not require a particularly fine time step to achieve the specified accuracy.

If, instead, we look at only the data for tolerances $\varepsilon = 10^{-3}$ and $\varepsilon = 5 \times 10^{-4}$, which both consist of more than 2 levels, we see that $\varepsilon^2(\log \varepsilon)^{-2} \times$Cost is roughly constant for the Multi Level, but clearly not constant for the Standard. We may conclude, therefore, that for the Standard Method the computational complexity is not $\mathcal{O}(\varepsilon^2(\log \varepsilon)^{-2})$, and in fact is $\mathcal{O}(\varepsilon^{-3})$, as we would expect from the Standard complexity theorem. For the Multi Level we have shown that the computational complexity is $\mathcal{O}(\varepsilon^{-2}(\log \varepsilon)^2)$ again as expected from the Multi Level complexity theorem.
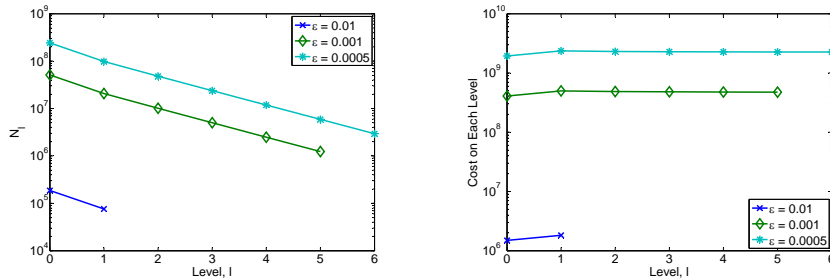
(a) Number of Samples on Each Level   (b) Cost on Each Level of Multi Level MC

Figure 3-9: Number of Samples computed with formula (2.12) and Cost on Each level of Multi Level MC for varying ε, $T = 1$

Figure 3-9(a) shows how the number of samples, computed with the formula (2.12) decreases with the levels, we see that compared to the plot for the QoI $= X$ in figure 3-4(a) the number of samples on each level decays at a slower rate, due the slower rate of converge of the variance of the differences. Indeed in figure 3-9(b) we see that the cost is spread evenly across all levels, again consistent with the case of the complexity

theorem where $\beta = 1$, so we see still see a gain by the implementation of Multi Level Monte Carlo.

ε	Standard MC Cost	Multi Level MC Cost	Speed-up
10^{-2}	1.67E+6	3.32E+6	0.50
10^{-3}	2.68E+9	2.82E+9	0.95
5×10^{-4}	2.14E+10	1.57E+10	1.36

Table 3.2: Speed-up of Multi Level MC over Standard MC, with $T = 1$, QoI $= \chi_{a,b}$, $[a, b] = [0.46, 0.63]$

Table 3.2 shows the theoretical cost of both the Standard and Multi Level methods, again computed using the number of samples and the number of time steps per sample. We see that for accuracy $\varepsilon = 10^{-2}$, the Standard Method outperforms the Multi Level, for $\varepsilon = 10^{-3}$ the cost for both methods is roughly the same however for $\varepsilon = 5 \times 10^{-4}$ we see the Multi Level outperform the Standard.

It is worth noting that although these initial results do not give quite as significant speed-up for the Multi Level method as we would have hoped at this stage, there are a number of factors that require further testing, not only the optimal size of the interval over which we evaluate our indicator function, but in particular the use of time step sizes of the form $h_\ell = 2^{-\ell}$ was arbitrary and as such changing this will likely give us better results. Taking a larger time step increment will require fewer levels in the Multi Level algorithm and thus reduce the cost. The effect of altering the time step increment is studied for the full model, see Section 4.5, with encouraging results.

The following two sections look at varying the end time and interval and we study how these affect how competitive the Multi Level method is compared to the Standard.
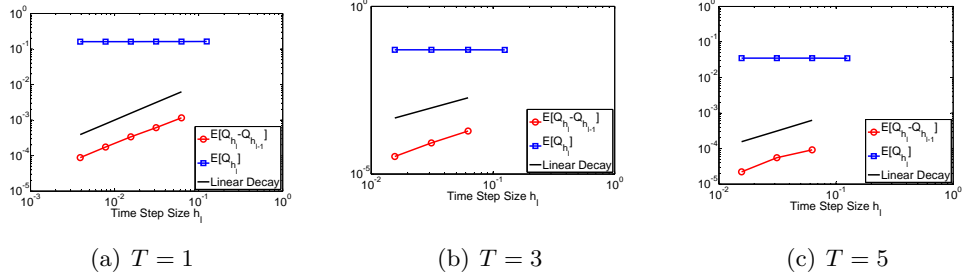
Varying End Time T

(a) $T = 1$ (b) $T = 3$ (c) $T = 5$

Figure 3-10: Expected Value for varying T, with $\varepsilon = 10^{-3}$, QoI $= \chi_{a,b}$, $[a, b] = [0.46, 0.63]$

Figure 3-10 shows the expected value of the differences, as well as the expected value for the Standard Monte Carlo on each level for $T = 1, 3, 5$. The coarsest level due to stability constraints is $h_0 = 2^{-3}$, we see that for the shortest time span $T = 1$, we require the most levels, with $h_L = 2^{-8}$ and for the longer time spans we require fewer levels, with $h_L = 2^{-6}$ in both cases. For each T we see again that the convergence rate of the expected value of the differences is $\mathcal{O}(h)$ and so we are again in the case where $\alpha = 1$ in the complexity theorems.

We also note that for the longer time spans the expected value of the quantity becomes smaller, as the plume becomes more spread out, see figure 3-5. In addition to this we see that the expected value of the differences is also smaller.
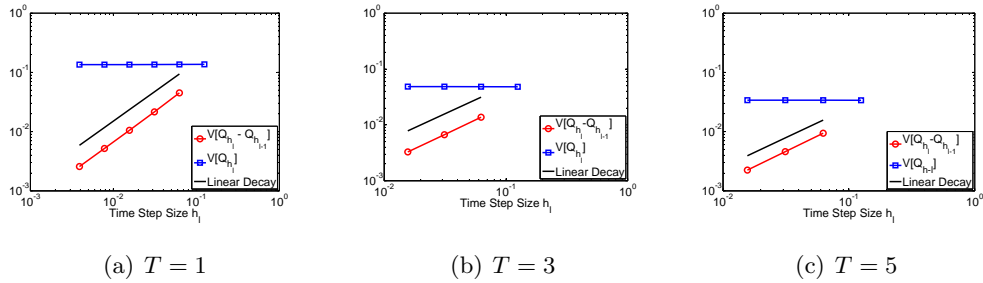

(a) $T = 1$ (b) $T = 3$ (c) $T = 5$

Figure 3-11: Variance Plots for varying T, with $\varepsilon = 10^{-3}$, QoI $= \chi_{a,b}$, $[a, b] = [0.46, 0.63]$

Figure 3-11 shows the variance of the differences in the Multi Level estimator, as well as the variance for the Standard Monte Carlo. Again we note that the convergence rate of the variance of the difference is linear for the quantity interest $\chi_{a,b}$ for each time $T = 1, 3, 5$. So again we have $\beta = 1$ in the complexity theorem and the complexity of the Multi Level Monte Carlo is $\mathcal{O}(\varepsilon^{-2}(\log \varepsilon)^2)$.

We note also that the variance becomes smaller as we increase T. As time progresses the plume spreads out, so it becomes wider but our interval $[a, b]$ does not and so a smaller fraction of particles will land in our interval, resulting in a decreasing mean. Therefore the reduction in the variance that we see may be as a result of the decreasing mean.
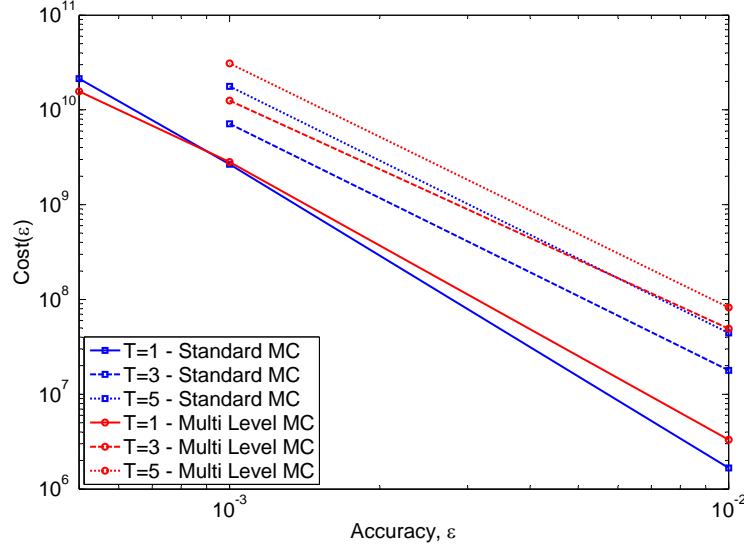

Figure 3-12: Cost as a function of ε for all $T = 1, 3, 5$, QoI $= \chi_{a,b}$, $[a, b] = [0.46, 0.63]$

Figure 3-12 shows the cost as a function of ε for the Standard and Multi Level methods for each T. Here due to time constraints we only have the results for $\varepsilon = 5 \times 10^{-4}$ for $T = 1$, however the plot still gives a good indication as to when the Multi Level becomes competitive. We see as before that for $T = 1$ the Multi Level only begins to perform in line with the Standard for $\varepsilon = 10^{-3}$ and only giving a marginal improvement for $\varepsilon = 5 \times 10^{-4}$.

We see that for the $T = 3$ and $T = 5$ that the cost for the Multi Level is still considerably higher than the Standard and is possible that the Standard would still outperform the Multi Level for $\varepsilon = 5 \times 10^{-4}$. So taking longer time spans, for this simplified problem the Multi Level method is never better than the Standard Method for any tolerances of practical interest. This may be due to the requirement of relatively few levels and a reasonably high variance of the differences for these levels. A more detailed investigation into this would be an interesting topic for further work.

T	h_0	h_L	Standard MC Cost	Multi Level MC Cost	Speed-up
1	2^{-3}	2^{-8}	2.68E+9	2.82E+9	0.95
3	2^{-3}	2^{-6}	7.16E+9	1.25E+10	0.57
5	2^{-3}	2^{-6}	1.78E+10	3.10E+10	0.57

Table 3.3: Speed-up of Multi Level MC over Standard MC for $\varepsilon = 10^{-3}$, QoI $= \chi_{a,b}$, $[a,b] = [0.46, 0.63]$, for varying end times, T

Table 3.3 shows the theoretical cost of both the Standard and Multi Level methods for accuracy $\varepsilon = 10^{-3}$, again computed using the number of samples and the number of time steps per sample. We see that for $T = 1$ the Standard and Multi Level Methods are roughly equivalent, but in the case of $T = 3, 5$, the Standard is around twice as fast as the Multi Level.

With these slightly discouraging results we look next at a different intervals $[a, b]$ to see whether results for the Multi Level Method improve when we look at intervals in which fewer particles are likely to land.

Varying Interval $[a, b]$

(a) $a, b = 0.46, 0.63$　　　(b) $a, b = 0.30, 0.46$　　　(c) $a, b = 0, 0.17$
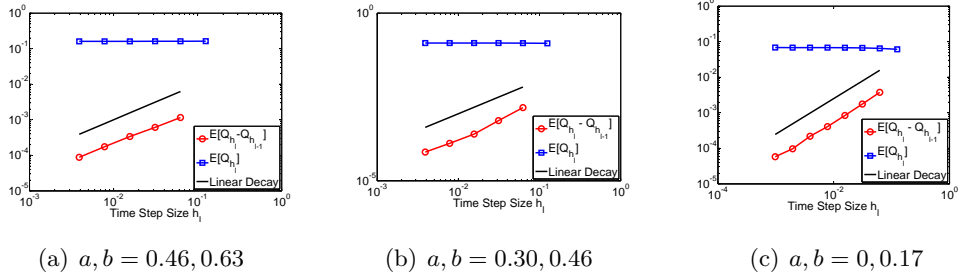
Figure 3-13: Expected Value for Varying Interval $[a, b]$, with $\varepsilon = 10^{-3}$, $T = 1$

Figure 3-13 shows the expected value of the differences, as well as the expected value for the Standard Monte Carlo on each level for three different intervals $[a, b]$, the first two of which being reasonably central in the plume, and the final one being slightly nearer the edge, in addition we note that the second two intervals are also smaller than the first, see figure 3-5.

The coarsest level due to stability constraints is $h_0 = 2^{-3}$, we see that for the two central intervals we require the six levels, with $h_L = 2^{-8}$ and for the final interval we require another two levels, so $h_L = 2^{-10}$.

For each interval we see that the convergence rate of the expected value of the differences is $\mathcal{O}(h)$ and so we are again in the case where $\alpha = 1$ in the complexity theorems.

(a) $a, b = 0.46, 0.63$ (b) $a, b = 0.30, 0.46$ (c) $a, b = 0, 0.17$
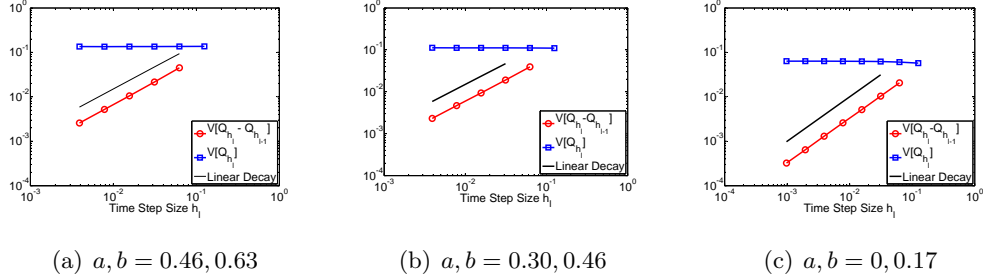
Figure 3-14: Variance Plots for Varying Interval $[a, b]$, with $\varepsilon = 10^{-3}$, $T = 1$

Figure 3-14 shows the variance of the differences in the Multi Level estimator, as well as the variance for the Standard Monte Carlo. Again we note that the convergence rate of the variance of the difference is linear for the quantity interest $\chi_{a,b}$ for each of the intervals. So we have $\beta = 1$ in the complexity theorem and the complexity of the Multi Level Monte Carlo is $\mathcal{O}(\varepsilon^{-2}(\log \varepsilon)^2)$.
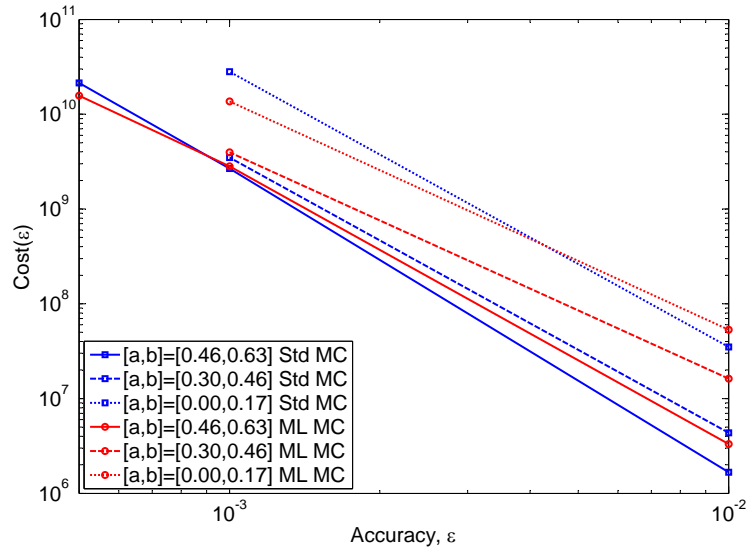

Figure 3-15: Cost as a function of ε for all Intervals, with $T = 1$

Figure 3-15 shows the cost as a function of ε for the Standard and Multi Level

50

methods for each interval. Here due to time constraints we only have have the results for $\varepsilon = 5 \times 10^{-4}$ for $[a, b] = [0.46, 0.63]$, however the plot still gives a good intuition as to when the Multi Level becomes competitive. We see as before that for $[a, b] = [0.46, 0.63]$ the Multi Level only begins to perform in line with the Standard for $\varepsilon = 10^{-3}$ and only giving a marginal improvement for $\varepsilon = 5 \times 10^{-4}$, we see similar results for the neighbouring interval $[a, b] = [0.3, 0.46]$. However for the interval $[a, b] = [0, 0.17]$ that the Multi Level outperforms the Standard much sooner, for $\varepsilon \approx 5 \times 10^{-3}$, which gives an indication that improvements could be seen in the prediction of rarer events, and more accurate predictions could be made for the movement of the edges of plumes.

Interval	h_0	h_L	Standard MC Cost	Multi Level MC Cost	Speed-up
$[0.46, 0.63]$	2^{-3}	2^{-8}	2.68E+9	2.82E+9	0.95
$[0.30, 0.46]$	2^{-3}	2^{-8}	3.48E+9	3.97E+9	0.88
$[0, 0.17]$	2^{-3}	2^{-10}	2.81E+10	1.37E+10	2.05

Table 3.4: Speed-up of Multi Level MC over Standard MC for $\varepsilon = 10^{-3}$, $T = 1$ QoI $= \chi_{a,b}$, with varying interval $[a, b]$

Table 3.4 shows the theoretical cost of both the Standard and Multi Level methods for accuracy $\varepsilon = 10^{-3}$, again computed using the number of samples and the number of time steps per sample. We see that for intervals $[0.46, 0.63]$ and $[0.30, 0.46]$ the Standard and Multi Level Methods are roughly equivalent, with the Standard just about outperforming the Multi Level. However with the interval $[0, 0.17]$ the Multi Level is around twice as fast as the Standard which could signal an encouraging result for the prediction of rarer events or the spread of particles at the edge of a plume, say.

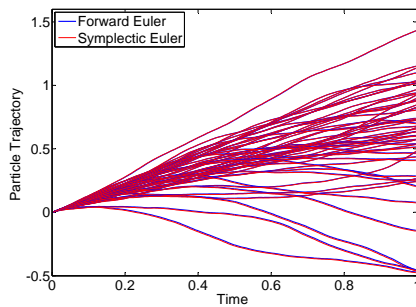### 3.5.5 Symplectic Euler vs Forward Euler

In this section we briefly examine the difference between the Forward Euler and Symplectic Euler methods.
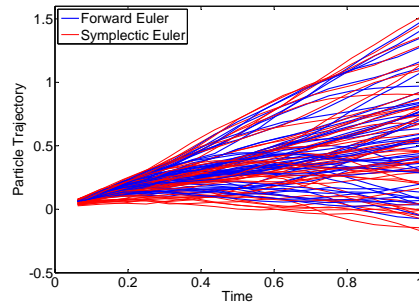
(a) $h = 2^{-4}$
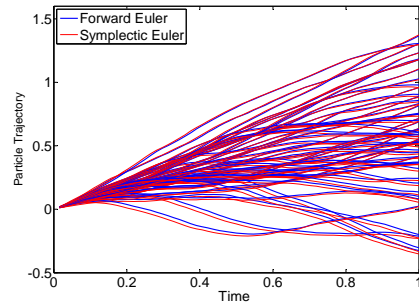
(b) $h = 2^{-6}$

(c) $h = 2^{-8}$

Figure 3-16: Trajectory Plots for Symplectic Euler and Forward Euler, for varying time step sizes

Figure 3-16 plots a small sample of trajectories computed with the Forward Euler method in blue and the Symplectic Euler method in red, for the same random variables, with three different time step sizes. We see that for coarser time steps the two trajectories quite frequently diverged by a significant amount, however as we made the time step smaller the two trajectories differ only by very small amounts.

For this simplified problem, at least for finer time steps we see no significant difference between the two methods, since there is no X dependence in the coefficients a and b, however for the full model, where there is X dependence the differences may be more substantial. As such for the full model tests we use the Symplectic Euler method in line with what the Met Office currently uses in NAME. It would be interesting to show how these methods differ for the full model, however unfortunately this is not investigated in this thesis.

3.6 Smoothed Indicator Function

In this section we present some preliminary results from the testing of a continuous, "smoothed" indicator function compared to the standard discontinuous indicator function $\chi_{a.b}$. We carry out some initial investigations into the improvements that could be seen in terms of variance reduction as well as any additional errors which may result from effectively smoothing the edges of our box.

The smoothed indicator function we study is as follows,

$$\chi_{a,b}^{\lambda}(X_M) = \frac{1}{2}\left(\tanh\left(\frac{X_M - a}{\lambda}\right) + \tanh\left(\frac{b - X_M}{\lambda}\right)\right) \qquad (3.19)$$

where λ is referred to as our *smoothing parameter* and controls by how much we smooth our box.
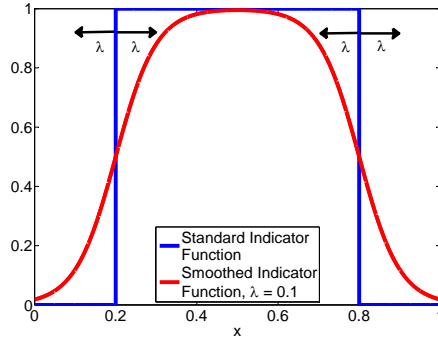

Figure 3-17: $\chi_{a,b}^{\lambda}(X_M)$, $[a,b] = [0.2, 0.8]$, $\lambda = 0.1$

We see from figure 3-17 that this function "smoothes out" our interval $[0.2, 0.8]$, by a width $\mathcal{O}(\lambda)$, so the additional error incurred by approximating our interval with the smoothed interval will depend on our smoothing parameter λ. However by choosing intervals [a,b] we no longer use the full pdf and so already introduce an error of $\sim |b-a|$, so strictly speaking this should be considered together with the smoothing error.

In order to study the error induced by the smoothing of our interval we study the two estimators, $\hat{Q}_{M,N}$ the estimator of the probability in our actual box and $\hat{Q}_{M,N}^{\lambda}$, the estimator of the probability in our smoothed box, which we define rigorously below.

Actual Interval:

$\mathbb{E}[Q_M]$ is defined as the expected value of $\chi_{a,b}$ i.e. the probability of landing in the **actual** interval $[a, b]$. To approximate this we have our estimator $\hat{Q}_{M,N} \approx \mathbb{E}[Q_M] = \mathbb{E}[\chi_{a,b}(X_M)]$, which we define as,

$$\hat{Q}_{M,N} = \frac{1}{N} \sum_{i=1}^{N} Q_M^{(i)},$$

where $Q_M^{(i)} = \chi_{a,b}(X_M^{(i)})$ is the functional, i.e. the standard indicator function, evaluated on the i-th sample of X_M, computed using Euler's Method with M time steps. Similarly,

Smoothed Interval:

$\mathbb{E}[Q_M^\lambda]$ is defined as the expected value of $\chi_{a,b}^\lambda$ i.e the probability of landing in the **smoothed** interval. To approximate this we have our estimator $\hat{Q}_{M,N}^\lambda \approx \mathbb{E}[Q_M^\lambda] = \mathbb{E}[\chi_{a,b}^\lambda(X_M)]$, which we define as

$$\hat{Q}_{M,N}^\lambda = \frac{1}{N} \sum_{i=1}^{N} Q_M^{\lambda,(i)}$$

where $Q_M^{\lambda,(i)} = \chi_{a,b}^\lambda(X_M^{(i)})$ is the functional, i.e. the smoothed indicator function, evaluated on the i-th sample of X_M, computed using Euler's Method with M time steps.

Using this notation we can now write our mean square error for the method with the smoothed indicator function as, see Appendix A.3,

$$\mathbb{E}\left[\left(\hat{Q}_{M,N}^\lambda - \mathbb{E}[Q]\right)^2\right] = \underbrace{\mathbb{V}[\hat{Q}_{M,N}^\lambda]}_{\text{sampling error}} + \underbrace{\left(\mathbb{E}[Q_M] - \mathbb{E}[Q]\right)^2}_{\text{discretisation error}} +$$

$$+ \underbrace{\left(\mathbb{E}[Q_M^\lambda] - \mathbb{E}[Q_M]\right)^2}_{\text{smoothing error}} + \underbrace{2\left(\mathbb{E}[Q_M^\lambda] - \mathbb{E}[Q_M]\right)\left(\mathbb{E}[Q_M] - \mathbb{E}[Q]\right)}_{\text{Cross Term}}$$

(3.20)

We see that the smoothing of the interval introduces two new error terms, a *smoothing error* term and an additional cross term involving the smoothing error and discretisation error. It is thus necessary to test the effect that a smoother interval has on the convergence rate of the variance of the differences and whether this is significant enough to warrant the introduction of two further error terms, as well as studying how these terms interact, particularly with respect to the time step size h as well as λ.

It is likely that for a large value of λ we may see gains in terms of variance reduction, however taking λ too large will result in our smoothed interval no longer giving a good representation of our actual interval, in particular, λ should be of the order of $|b - a|$ or smaller.

As well as considering the smoothing parameter, λ, when testing our method, we must also identify the effect of choosing different sizes and positions of our interval.

The density of particles in a particular part of the atmosphere may mean that our method performs better or worse, for example in a part of the atmosphere where very few particles land, a so-called "rare event", we are likely to need a lot of samples to control the variance and in turn the sampling error. In this section however we present only a few preliminary tests for varying values of λ.
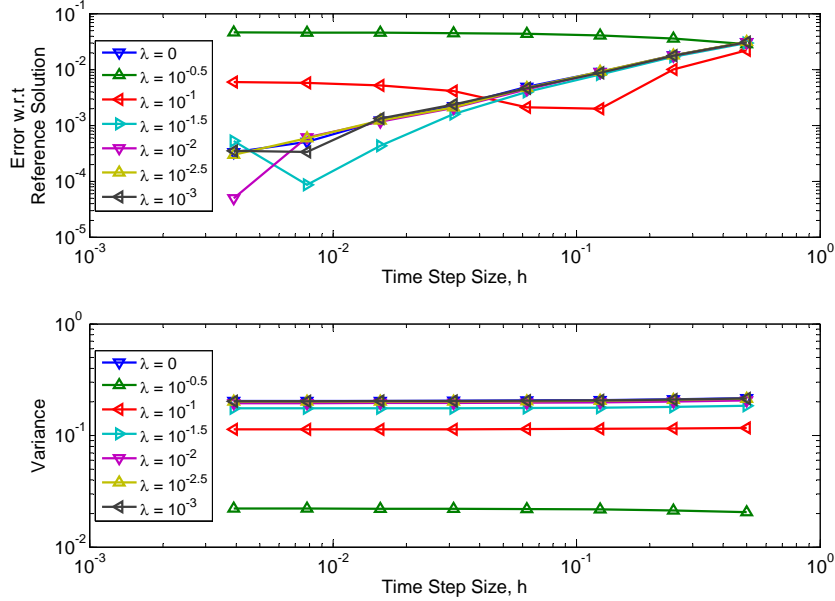

Figure 3-18: Error (top) with respect to reference solution with $\lambda = 10^{-6}$, $N^* = 10^9$, $h^* = 2^{-10}$ and Variance (bottom) for various smoothing parameters λ for $h = 2^{-1}, \ldots, 2^{-8}$, with $N = 10^7$

The first plot in Figure 3-18 shows how the error behaves with respect a reference solution, which was calculated to a high accuracy with a small λ. The second plot shows the variance for our smoothed indicator function with varying values of λ, including $\lambda = 0$, i.e. with no smoothing.

It can be shown theoretically, [24], that at lowest order the error, $E(h, \lambda)$, is of the form,

$$E(h, \lambda) = C_h . \frac{h}{\tau} + C_\lambda . \lambda^2.$$

Indeed if we look at the error plot in Figure 3-18, we see that for the smallest time step size the error is reduced by one order of magnitude when going from $\lambda = 10^{-0.5}$ to 10^{-1}. Unfortunately for these preliminary tests we cannot infer much from the smaller

values of λ as these results are clearly dominated by statistical noise and as such would require more samples in order to able to draw conclusions from them. The results for $\lambda = 10^{-0.5}$ and $\lambda = 10^{-1}$, however, agree with the above formula.

In the second plot in Figure 3-18 we see that for $\lambda \leq 10^{-1.5}$ the variance for each of the values of λ are quite close to each other and appear to be converging. For $\lambda = 10^{-0.5}$ and $\lambda = 10^{-1}$ we see significantly reduced variance, which would be an advantage for the Multi Level method, thus requiring fewer samples to bound the sampling error; however these values of λ are those for which the error with respect the reference solution was significant, at around 10^{-2} to 10^{-1}.

From these preliminary tests for this particular smoothed indicator function we concluded that for $\lambda \leq 10^{-1.5}$ the difference between the variance and that of the "non-smoothed" reference solution was not significant enough to result in major gains for the Multi Level algorithm, and for those values that we would see gains, the error introduced was too large. So for the purposes of this thesis we used only the standard, discontinuous indicator function, however in future work it would be important to do both further tests with this smoothed indicator function as well as exploring the possibilities of more advanced smoothing functions.

3.7   Conclusions

In this chapter we have been able to show that the Multi Level Method has a lot of potential in its application to atmospheric dispersion modelling, in particular when our quantity of interest was taken to be the mean particle position we showed that by implementing the Multi Level Method we can achieve an order of magnitude improvement in terms of the computational complexity over the Standard Method.

Taking a quantity of interest which is more relevant to the application, i.e. the probability of a particle being a particular "box" at time T, we saw that although not as significant, improvements were still evident over the Standard Method from the implementation of the Multi Level.

In addition we studied how various model parameters, such as box size and timescales, interact with the Multi Level Method and how these affect the method's performance. A next step in continuing this project would then be to quantify these effects in order to ensure the optimal performance of the method's implementation should the Met Office choose to adopt it.

It is however worth bearing in mind that the results presented here are for reasonably low level of accuracy which corresponds to that required by the Met Office on an operational level, where 10^{-2} and 10^{-3} are already considered to be small error

tolerances. As such the gains seen by implementing Multi Level Monte Carlo here are perhaps not as striking as those considered in mathematical papers which may show results for much smaller tolerances. However, they still signify an improvement over the current method and, with questions surrounding how exactly the box size, timescales and functionals relate to the performance, further testing may indeed show that the Multi Level Method is even better than we have seen so far.

It is also important to mention at this stage that for this simplified model problem we have so far only considered time steps of the form $h_\ell = 2^{-\ell}$, in the following chapter we consider larger time step increments, which in turn reduce the number of levels in the Multi Level algorithm and thus improve its competitiveness with the Standard as such the outlook for the Multi Level Method may be better than the results in this chapter indicate.

In the next chapter we move on to look at a more complicated test; this so called full model is very close to that which is currently used by the Met Office and as such will enable us to better identify the potential the method has in this application to atmospheric dispersion modelling.

Chapter 4

Full Model Problem

4.1 Full Model

The Met Office has supplied us with a model close to that which they use currently in NAME to approximate the dispersion of atmospheric pollutants; again in the 1-dimensional case at time t, we take a typical particle with trajectory $X(t) \in \mathbb{R}$, i.e. vertical height above ground, mean velocity $v(X(t), t) \in \mathbb{R}$ and vertical deviation of the velocity from the mean due to turbulence $U(t) \in \mathbb{R}$. The time evolution of $U(t)$ and $X(t)$ are described by the SDEs,

$$
\begin{aligned}
\mathrm{d}U &= a(U, X, t)\mathrm{d}t + b(X, t)\mathrm{d}W(t), \\
\mathrm{d}X &= \Big[v(X, t) + U\Big]\mathrm{d}t.
\end{aligned}
\tag{4.1}
$$

For simplicity in our numerical tests we take the mean velocity $v(X(t), t)$ to be zero.

For this problem we note that the first coefficient, $a(U, X, t)$, is as in the simplified case in equation (3.1) but with an additional contribution necessary to satisfy the well-mixed condition which expresses the physical fact that an initially uniform distribution of particles should not become unmixed at a later time, [11][12][18][19].

$$
a(U, X, t) = -\frac{U}{\tau(X)} + \underbrace{\frac{1}{2}\Big(1 + \frac{U^2}{\sigma_U^2(X)}\Big)\frac{\mathrm{d}\sigma_U^2(X)}{\mathrm{d}X}}_{\text{additional term}}.
\tag{4.2}
$$

The second coefficient has the same form as our simplified model problem (3.1); how-

ever, now both $\sigma_U(X)$ and $\tau(X)$ vary with X,

$$b(X,t) = \sqrt{\frac{2\sigma_U^2(X)}{\tau(X)}}, \qquad (4.3)$$

where $\sigma_U(X)$, the velocity variance term, and $\tau(X)$, the velocity memory term, are given by,

$$\sigma_U(X) = k_\sigma u^* \left(1 - \frac{X}{H}\right)^{3/4},$$

$$\tau(X) = k_\tau \frac{X}{\sigma_U(X)}. \qquad (4.4)$$

The analytical form given in equations (4.4) describes a moderate-wind boundary layer, [11][12], with u^* the frictional velocity, which for our tests we take as $u^* = 0.2\text{ms}^{-1}$. H is the boundary layer depth and in our tests we take $H = 1\text{km}$. k_τ and k_σ are dimensionless parameters which we have taken to be $k_\tau = 0.5$ and $k_\sigma = 1.3$.

We observe that for particles close to the top and bottom boundary layers that $\sigma_U(X)$ and $\tau(X)$ have singularities, these cause $a(U, X, t)$ and $b(X, t)$ to grow; the numerical solution of the SDE then rapidly becomes unstable near the boundaries, making unfeasible jumps at each time step, no longer giving a good representation of the behaviour of a particle in the atmosphere. It should be stressed, however, that the formulae in (4.4) are parametrisations (obtained from fitting measurements to a simple function) which are not necessarily physically accurate over the whole boundary layer depth.

We must address these singularities in the coefficients $a(U, X, t)$ and $b(X, t)$, in order to ensure that our model represents, as closely as possible, the real life dispersion of atmospheric pollutants. In the next section we study these singularities and regularise both $\sigma_U(X)$ and $\tau(X)$ at the boundaries in order to ensure that our solutions remain stable.
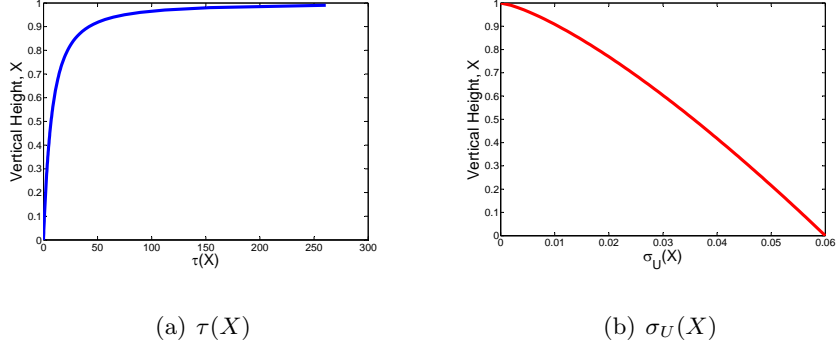
4.2   Regularisation

(a) $\tau(X)$ (b) $\sigma_U(X)$

Figure 4-1: Plots showing τ and σ_U as a function of their vertical height

Figure 4-1 shows τ and σ as a function of the vertical height of a particle, we see that for particles close to the ground i.e. as $X \to 0$, that

$$\sigma_U(X) \;=\; k_\sigma u^* \left(1 - \frac{X}{H}\right)^{3/4} \to k_\sigma u^*, \quad \text{a constant}$$

$$\tau(X) \;=\; k_\tau \frac{X}{\sigma_U(X)} \to \frac{k_\tau}{k_\sigma u^*} X \to 0.$$

Since both $a(X, U, t)$ and $b(X, U, t)$ include terms proportional to $1/\tau(X)$, both of these terms will blow up as $X \to 0$.

Similarly for particles near to the top boundary, i.e. as $X \to H$,

$$\sigma_U(X) \;=\; k_\sigma u^* \left(1 - \frac{X}{H}\right)^{3/4} \to 0.$$

Since the second term in the coefficient $a(X, U, t)$ has a dependence on $1/\sigma_U^2(X)$, this term will blow up for particles near the top boundary.

In order to ensure that these terms do not blow up and thus that our method remains stable, we regularise our parameters. In other words we take $\tau(X)$ to be a constant when our particle comes within a distance ε_{reg} of the bottom boundary, where ε_{reg} is a, generally small, parameter which we may choose. So at the bottom boundary, i.e. for $X < \varepsilon_{reg}$, we take,

$$\tau(X) = \tau(\varepsilon_{reg}).$$

Similarly, at the top boundary, i.e. $X > H - \varepsilon_{reg}$, we take,

$$\sigma_U(X) = \sigma_U(H - \varepsilon_{reg}).$$

$\sigma_U(X)$ and $\tau(X)$, however, depend closely on one another, thus, if they are not consistent, our model will not be a good approximation of the behaviour of airborne particles in the atmosphere, in particular it would violate the well-mixed condition. As such to ensure that we do satisfy this condition we regularise both $\tau(X)$ and $\sigma_U(X)$ at the top *and* bottom boundaries, as well as choosing appropriate constants for $\mathrm{d}\sigma_U^2/\mathrm{d}X$. Thus for our regularisation at the bottom boundary, i.e. for $X < \varepsilon_{reg}$, we take,

$$\tau(X) \quad = \quad \tau(\varepsilon_{reg}), \quad \sigma_U(X) \quad = \quad \sigma_U(\varepsilon_{reg}).$$

Similarly, at the top boundary, i.e. $X > H - \varepsilon_{reg}$, we take,

$$\sigma_U(X) \quad = \quad \sigma_U(H - \varepsilon_{reg}), \quad \tau(X) \quad = \quad \tau(H - \varepsilon_{reg}).$$

This regularisation, however, gives us an additional parameter, which will determine, to some extent, the accuracy of our solution. Taking a large regularisation parameter, ε_{reg}, will mean that $\tau(X)$ and $\sigma_U(X)$ will be constant for a significant part of the particle's path. This may affect how well our model represents true atmospheric dispersion, although it is important to bear in mind that (4.4) is only a parametrisation. Choosing a small ε_{reg}, however, will mean that at the boundaries the coefficients $a(X, U, t)$ and $b(X, t)$ will still diverge.

It is worth noting that $\sigma_U(X)$ and $\tau(X)$ are only models and are not likely to be accurate for $X \to 0$ and $X \to H$, thus it is debatable as to whether the error induced by regularising these functions is significant or whether this can be included in the parametrisation error already in $\sigma_U(X)$ and $\tau(X)$. A thorough investigation of this would be an interesting and important question for future work.

We must also consider the affect that regularising these parameters has on the stability of the numerical method, and the restrictions that we must place on the time step size in order to ensure that stability. This is investigated in the following section.

4.3   Stability Analysis

In order to determine the coarsest time step feasible for the Multi Level Method and thus how many levels we can use, we must study the stability of the numerical method. As with the simplified model in Section 3.4, for the full model we have so far only

studied the stability of the deterministic part, in other words the ODE described by,

$$dU = \left(-\frac{U}{\tau(X)} + \frac{1}{2}\left(1 + \frac{U^2}{\sigma_U^2(X)}\right)\frac{d\sigma_U^2(X)}{dX} \right)dt.$$

However this ODE is still not straightforward to study, so in order to simplify our calculations we note that the dominant term, for $X \to 0$, is $U/\tau(X)$. We have not yet studied the stability constraints which are determined by the divergence of the $d\sigma_U^2(X)/dX$ term at the top boundary i.e. for $X \to H$, however future work should allow for further investigation in this area. For now we assume that the stability of the method is determined primarily by the $U/\tau(X)$ term, thus in a similar vein to the simplified case we analyse the stability of the ODE given by,

$$dU = -\frac{U}{\tau(X)}dt. \tag{4.5}$$

In contrast to the simplified case we now have that $\tau(X)$ depends on X, so we must ensure that the condition $h < 2\tau(X)$ holds for all X, so in particular where $\tau(X)$ takes its minimum. We observe that $\tau(X)$ is a strictly increasing function and since we have regularised $\tau(X)$ its minimum thus occurs at $\tau(\varepsilon_{reg})$. As a result, the largest time step, h_0, that we can choose to ensure stability, changes for different values of ε_{reg}. This allows for the possibility to take larger regularisation parameters, ε_{reg}, on coarser levels of the Multi Level method whilst still ensuring that the method remains stable on all levels. It is, however, possible that this may result in the growth of the variance of the differences $Y_\ell = Q_{M_\ell} - Q_{M_{\ell-1}}$ and thus we would need to balance these two considerations.

Here we present some preliminary numerical tests which verify our assumption that the stability of the full method for approximating the solution of the SDE in equation (4.1) can be reduced to that of the ODE in equation (4.5).
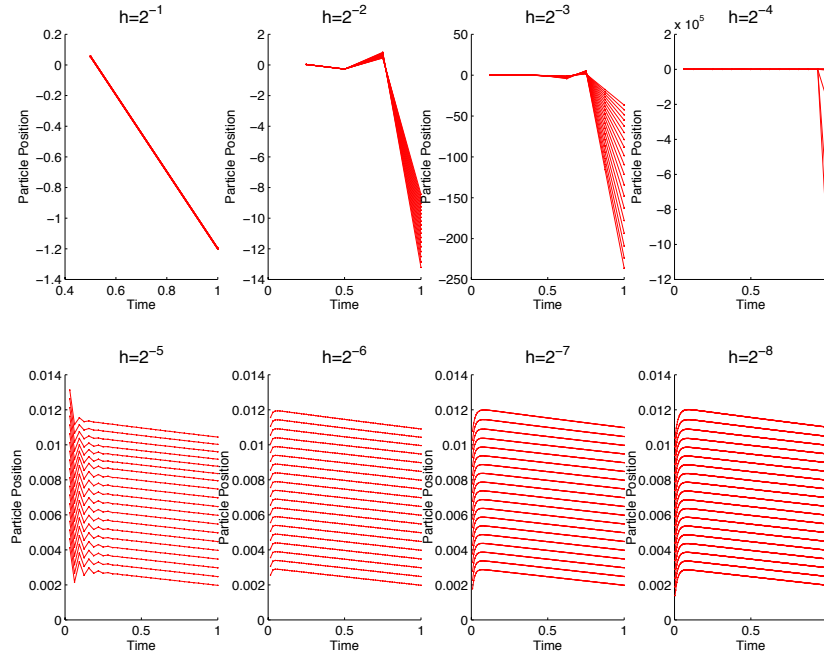
Figure 4-2: Plots showing stability of trajectory for ODE in equation (4.5), coupled with $dX = Udt$. Varying time step size $h = 2^{-1}, \ldots, 2^{-8}$, $T = 1$, for fixed $\varepsilon_{reg} = 0.01$.

Figure 4-2 plots for each different time step size $h_\ell = 2^{-\ell}$, $\ell = 1, \ldots, 8$, the trajectory of particles released at $X_0 = 0.001, 0.00015, \ldots, 0.01$, with initial velocity $U_0 = 0.1$, from release up until time $T = 1$. We can see from these plots that with $\varepsilon_{reg} = 0.01$ our method is clearly unstable for $h \geq 2^{-4}$.

By using the formula for $\tau(X)$, we can write,

$$\tau(\varepsilon_{reg}) = \frac{K_\tau}{K_\sigma u*}\varepsilon_{reg}(1 - \varepsilon_{reg})^{-3/4},$$

then with the parameters as given in Section 4.1 we have,

$$\tau(\varepsilon_{reg}) \approx 1.92\varepsilon_{reg}\underbrace{(1 - \varepsilon_{reg})^{-3/4}}_{\approx\, 1,\text{ for small }\varepsilon_{reg}}$$

$$\approx 2\varepsilon_{reg}$$

so we would expect to see instability for time step sizes $h > 2\tau(\varepsilon_{reg}) \approx 2 \times 2\varepsilon_{reg} = 4\varepsilon_{reg}$.

For our plots we have taken $\varepsilon_{reg} = 0.01$, this analysis implies we would see instability for $h > 4 \times 0.01 = 0.04$. Indeed, as we noted before, we see on our plots that for $h = 2^{-4} = 0.0625 > 0.04$ the method is clearly unstable. However for $h = 2^{-5} = 0.03125 < 0.04$ we see that our method is stable (although the initial kink implies it
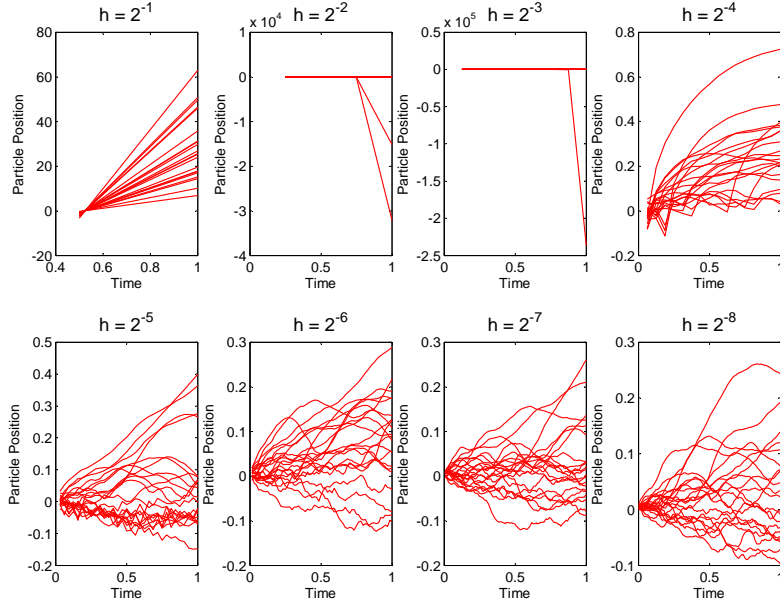
may be inaccurate).

Figure 4-3: Plots showing stability of trajectory for SDE in equation (4.1) with $v = 0$. Varying $h = 2^{-1}, \ldots, 2^{-8}$, $T = 1$, for fixed $\varepsilon_{reg} = 0.01$

Figure 4-3 shows again for each different time step size $h_\ell = 2^{-\ell}$, $\ell = 1, \ldots, 8$, the trajectories for particles released at $X_0 = 0.001, 0.00015, \ldots, 0.01$ with initial velocity $U_0 = 0.1$, from release up until time $T = 1$, with $\varepsilon_{reg} = 0.01$, but this time for the full SDE as in equation (4.5), with mean velocity, $v = 0$.

We see from this plot, for the particles released at the bottom of the boundary layer that indeed it appears as if the stability of the SDE corresponds to that of the ODE, with the instability for time step size $h = 2^{-4}$ less prominent as for the ODE although we note one particle's trajectory clearly diverges in this plot. Again we see that for time steps of size larger than $h = 2^{-4}$ that the method appears to be stable.

Similar preliminary tests were also carried out for different values of ε_{reg}, as well as for the full model with reflection and for particles released at the top boundary layer where it is the $\mathrm{d}\sigma_U^2(X)/\mathrm{d}X$ term that dominates. For these tests the method appeared to agree with the stability constraints imposed by the simplified ODE in equation (4.5). The complete stability analysis of the SDE would, however, be an interesting topic for further work.

Further to this it is well known that implicit numerical methods give way to much less restrictive stability constraints and thus it is likely that we would see improvements

in our method, allowing coarser levels to be used in the Multi Level algorithm, as a result of implementing an implicit method for the deterministic part of the SDE. The use of implicit methods for the stochastic component, however, is not a trivial subject and is beyond the scope of this thesis.

The improved stability properties and often unconditional stability associated with implicit methods however comes at a cost; implementing an implicit method for even just the deterministic part of the SDE can be very expensive compared to an explicit method, such as Forward Euler.

As such this topic would be interesting to investigate further once the current method is established to work well and it is determined that the benefit of using a coarser time step in the Multi Level algorithm outweighs the cost of the implementation of a more complicated implicit method for either the deterministic part, or eventually the full SDE.

4.4   Reflection at Boundaries

For our model as described in the previous section, there are no restrictions on where a particle can travel; in a practical sense this means that we currently allow particles to go beneath the ground and come back up, which simply isn't physical. Thus in order to ensure that the model correctly represents the behaviour of real particles in the atmosphere we must include conditions which restrict the particle's trajectory to only those physically feasible paths.

Currently the Met Office implements a combination of reflective and absorbing boundary conditions, where the absorption represents the deposition of pollutants on the ground. For simplicity we do not consider absorption in this thesis; however we do look at reflecting particles at both the top and bottom boundaries.

As well as reflecting those particles that "hit" the ground, we also reflect particles at the top of the atmosphere (an alternative treatment would be to allow them to also propagate in the free troposphere and re-enter the boundary layer, but we do not consider this here for simplicity).

In line with the Met Office's current method reflection is treated as follows; for a particle which hits the ground, the particle position, X, is replaced by $X = -X$ and the velocity is reversed, $U = -U$.
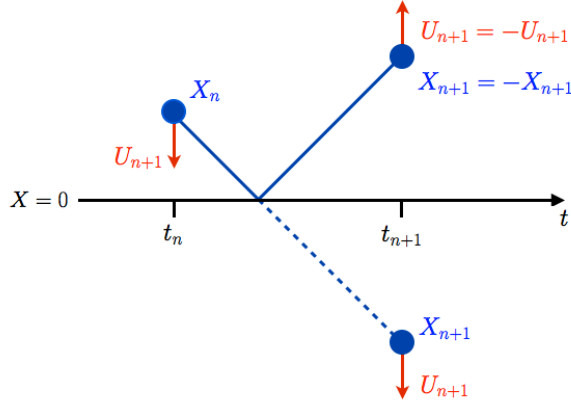
Figure 4-4: Illustration of Reflection at Boundary $X = 0$

This method of reflection at the boundary is illustrated in figure 4-4. We see that X is reflected geometrically about the ground $X = 0$ and, since U has memory, we reflect this also, as if we don't change the sign of U, the particle would then repeatedly hit the boundary. In addition, since we have regularised $\tau(X)$ near the boundary, particles hit the ground more often than they should, and so if U is not reflected the resultant concentration of particles near the ground will be too high indicating that the well-mixed condition, [11][12], is no longer satisfied.

4.5 Numerics

4.5.1 Without Reflection at Boundaries

We examine the full model, for the 1-dimensional case, as described in Section 4.1 with our quantity of interest restricted to the probability of a particle landing in an interval $[a, b]$ at time T, since this is what the Met Office requires as output of the program. For our tests we took the initial conditions to be $U_0 = 0.1$, and $X_0 = 0.05$ corresponding to the release of a particle from a chimney, say, at 50m above ground level, with velocity $0.2ms^{-1}$.

Figure 4-5 shows the path of 50 model particles and their spread over time, until $T = 1$. Here the red blocks indicate the intervals in which we look for the probability of a particle landing at time T.
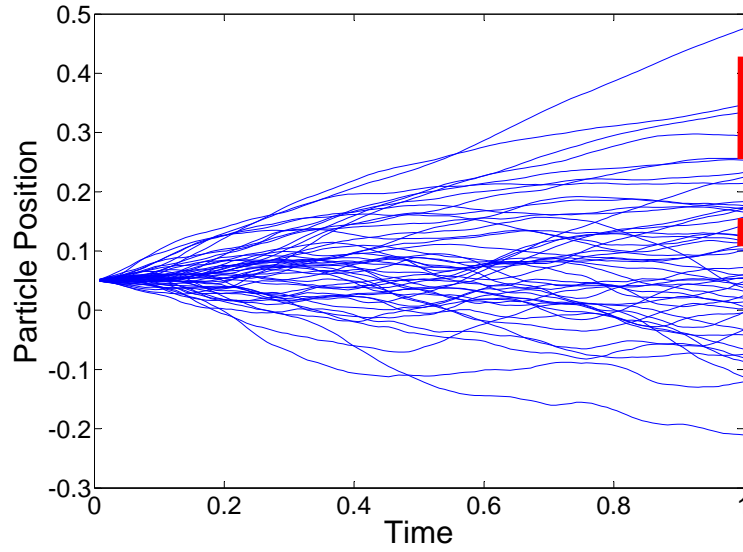
Figure 4-5: Full Model Trajectory Plot for 50 Independent Particles Released at $X_0 = 0.05$

For the following tests we implement the algorithm with regularisation at the boundaries, first without reflection, so, in a physical sense, particles are allowed to go beneath the ground and come back up again. The model with reflection is then studied in Section 4.5.2, allowing us to identify the effect that reflecting particles at the boundary may have on the efficiency of the method and the gains observed for the Multi Level Method.
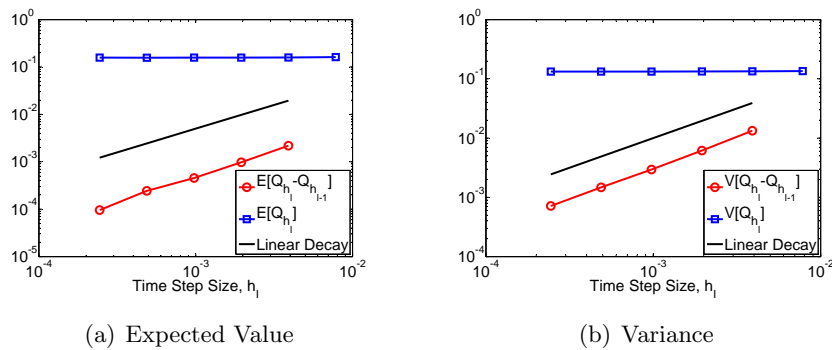

(a) Expected Value

(b) Variance

Figure 4-6: Expected Value and Variance for $[a, b] = [0.1055, 0.1555]$, $\varepsilon = 10^{-3}$, $T = 1$

On the lefthand side Figure 4-6 shows the expected value of differences in the Multi Level Method (in red), i.e. $\mathbb{E}[Q_{M_\ell} - Q_{M_{\ell-1}}]$, as well as the expected value, $\mathbb{E}[Q_{M_\ell}]$, for

the Standard Method (in blue) computed for each time step used in the Multi Level Method. On the right we have the variance of the differences in the Multi Level Method (in red), i.e. $\mathbb{V}[Q_{M_\ell} - Q_{M_{\ell-1}}]$, as well as the variance $\mathbb{V}[Q_{M_\ell}]$ for the Standard Method (in blue) computed for each time step used in the Multi Level Method.

As for the simplified model, we again have that the expected value for the Multi Level Method converges with $\mathcal{O}(h)$, corresponding to $\alpha = 1$ in the Standard and Multi Level complexity theorems. In addition we see that the convergence rate of the variance of the differences is linear, so $\beta = 1$ in the complexity theorem. We note that our quantity of interest here is the probability, so our functional is again the indicator function and as such our linear convergence rate of the variance is consistent with that for the simplified model problem.

As a result we are able to conclude from the complexity theorem, that the computational complexity for the Multi Level Method is $\mathcal{O}(\varepsilon^{-2}(\log \varepsilon)^2)$ compared to $\mathcal{O}(\varepsilon^{-3})$ for the Standard (since the cost per sample of the Euler Method is $\mathcal{O}(h^{-1})$, which implies $\gamma = 1$, in the Standard Monte Carlo complexity theorem).
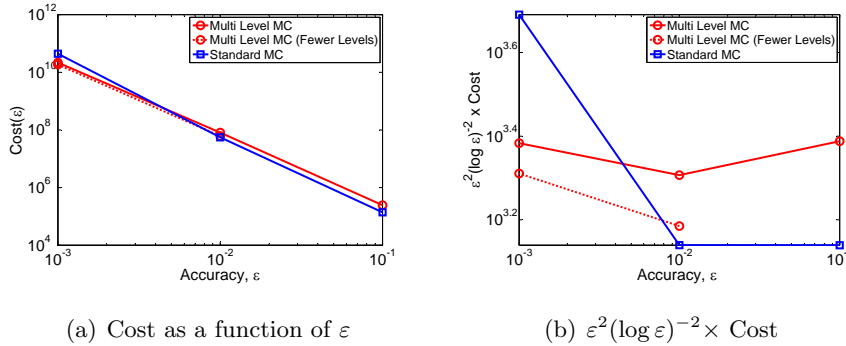

(a) Cost as a function of ε (b) $\varepsilon^2(\log \varepsilon)^{-2} \times$ Cost

Figure 4-7: Cost of Standard MC vs Multi Level MC, with $T = 1$, QoI $= \chi_{a,b}$, $[a,b] = [0.1055, 0.1555]$

In Figure 4-7 we look at the cost of the Multi Level Monte Carlo Method versus that of the Standard Method with our quantity of interest being the probability of a particle being in the interval $[a, b]$ at time $T = 1$. As with the simplified model problem these plots use theoretical cost computations which are based on the total number of time steps, computed using the N_ℓ's obtained from the formula in (2.12) (as for the simplified model this is scaled using the relative error, so we actually use the formula in equation (3.17)) and the unit of cost is the cost per sample per time step.

Again we assume that for the Multi Level Monte Carlo the cost of computing the difference $\mathbb{E}[Q_{M_\ell} - Q_{M_{\ell-1}}]$ is 1.5 times that of computing $\mathbb{E}[Q_{M_\ell}]$ since the quantity computed with the coarser time step will have half as many Euler time steps as the

finer (with steps incremented in powers of 2).

Figure 4-7(a) plots the theoretical cost for both the Standard (blue) and the Multi Level Methods (red), for accuracy tolerances $\varepsilon = 10^{-1}, 10^{-2}$ and 10^{-3}. We note that there are two sets of data for the Multi Level Method, the first, illustrated by the continuous line has the maximum number of levels with the coarsest level being that which is determined by the stability constraints associated with the method. The second, which is represented by the dashed line is that which has a smaller coarsest level, so our Multi Level Method requires fewer levels in this case. We see from these plots that there is a slight reduction in cost as a result of taking a smaller coarsest level, these comparisons are however examined in more detail in Table 4.1.

What we see from Figure 4-7(a), however, is that the Standard Method outperforms the Multi Level for accuracy tolerances of $\varepsilon = 10^{-1}$ and 10^{-2}, it is worth noting however that due to the longer run time of these problems the plots here only consider accuracy tolerances up to 10^{-3}, rather than the tolerances of 5×10^{-4} which we studied for the simplified model. We also observe that the Multi Level Method outperforms the Standard from accuracy tolerances of around 5×10^{-3} and clearly outperforms it for $\varepsilon = 10^{-3}$.

In Figure 4-7(b) we plot $\varepsilon^2 (\log \varepsilon)^{-2} \times$ Cost for both the Standard (blue) and Multi Level Method (red). From this plot we verify the conclusion that we drew from the complexity theorem, that the Multi Level Monte Carlo has computational cost of $\mathcal{O}(\varepsilon^{-2}(\log \varepsilon)^2)$, since for the Multi Level Method we see that $\varepsilon^2(\log \varepsilon)^{-2} \times$ Cost is roughly constant, with a different constant for the method with fewer levels. In addition we see that, since $\varepsilon^2 (\log \varepsilon)^{-2} \times$ Cost is clearly not constant in the Standard case, the computational complexity is of order higher than $\mathcal{O}(\varepsilon^{-2}(\log \varepsilon)^2)$, namely $\mathcal{O}(\varepsilon^{-3})$, as expected.
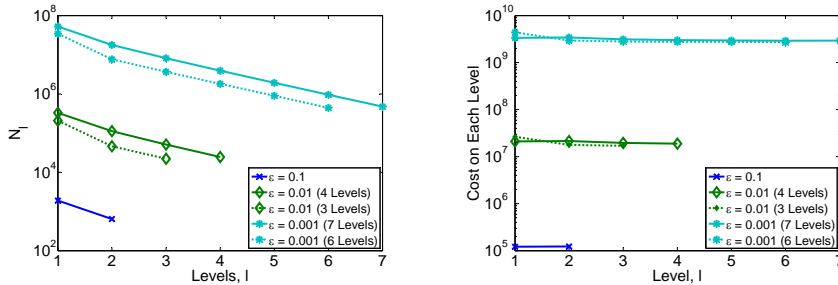
(a) Number of Samples on Each Level (b) Cost on Each Level of Multi Level MC

Figure 4-8: Number of Samples, computed with formula (2.12) and Cost on Each level of Multi Level MC for varying ε, $T = 1$

Figure 4-8 looks at the number of samples on each level of the Multi Level as well as the Cost on each level. Again we note that the dashed lines correspond to the Multi Level Method whose coarsest level is chosen to be smaller than the largest time step size determined by the stability constraints.

In Figure 4-8(a) we see how the number of samples decays with on each level of the Multi Level Algorithm for several different accuracy tolerances, ε. We see that again the linear decay rate of the variance means that the number of samples required on each level decays relatively slowly.

In addition to this it is important to note that, for the Multi Level Method when we used a smaller coarsest time step size and thus had fewer levels, the number of samples required on each level with time step size h_ℓ is similar to that for the same time step size when the method was implemented with more levels. In Figure 4-8 level 1 corresponds to the coarsest level which differs for the two different implementations of the Multi Level Method, which is misleading in this graph, it would have been better to plot against the time step size in this case rather than against the level.

From Figure 4-8(b) we note how the cost is distributed on each level, we see here that the cost is clearly distributed evenly across all levels which is consistent with being in the case where $\beta = 1$ in the complexity theorem.

ε	h_0	h_L	No. Levels	Speed-up
10^{-1}	2^{-5}	2^{-7}	3	0.29
10^{-1}	2^{-6}	2^{-7}	2	0.56
10^{-2}	2^{-5}	2^{-9}	5	0.47
10^{-2}	2^{-6}	2^{-9}	4	0.68
10^{-2}	2^{-7}	2^{-9}	3	0.90
10^{-3}	2^{-6}	2^{-12}	7	2.03
10^{-3}	2^{-7}	2^{-12}	6	2.40

Table 4.1: Speed-up of Multi Level MC over Standard MC for Differing Coarsest Levels, QoI = $\chi_{a,b}$, $[a, b] = [0.1055, 0.1555]$, Without Reflection

Table 4.1 helps to demonstrate how the number of levels and indeed the coarsest level that we take for the Multi Level Method influences how competitive it will be against the Standard Method. We see clearly that for these very large accuracy tolerances of 10^{-1} and 10^{-2} (where we mean large in a mathematical sense, for practical application these are actually quite relevant tolerances) the Standard performs much better than the Multi Level, but we see an improvement of nearly double in the speed-

up for accuracy of 10^{-1} by simply removing one level, which allows us to go from a speed-up of 0.29 to 0.56. Similarly for the tolerance 10^{-2} by removing two levels our speed-up is again doubled from 0.47 to 0.90. We also see improvements for the tolerance 10^{-3} by going from 7 levels to 6, as such it would be important to do additional tests for this tolerance with fewer levels in order to be able to establish a link between the number of levels and the speed-up for the Multi Level Method over the Standard.

Until this point in the thesis we have always considered time steps of the form $h_\ell = 2^{-\ell}$, however this choice was somewhat arbitrary and as such in order to ensure that this choice does not limit the success of the Multi Level Method we briefly look at the affect that choosing a different larger time step size increment has on the performance of the method.

For the next tests we simply looked at time steps of the form $h_\ell = 2^{-\ell}$ and $h_\ell = 4^{-\ell}$ since for this case we could directly compare their performance, in particular by using the same finest level, so that the solution is computed to precisely the same accuracy. We also ran some preliminary tests for $h_\ell = 6^{-\ell}$, which seemed to show even better performance, however here for simplicity we only study the time step increments which allow for direct comparison and thus the results for $h_\ell = 6^{-\ell}$ are not included in this thesis. Further, more thorough tests for different time step increments would, however, be an important consideration for further work.

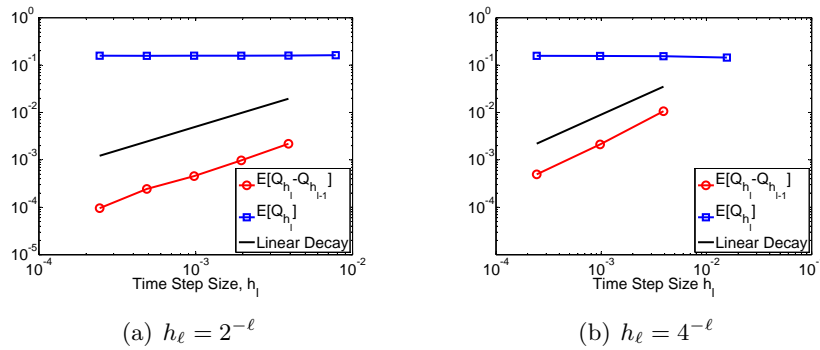(a) $h_\ell = 2^{-\ell}$ (b) $h_\ell = 4^{-\ell}$

Figure 4-9: Expected Value for Varying Increments, with $\varepsilon = 10^{-3}$, $T = 1$

Figure 4-9(a) shows the expected value of differences in the Multi Level Method (in red), i.e. $\mathbb{E}[Q_{M_\ell} - Q_{M_{\ell-1}}]$, as well as the expected value, $\mathbb{E}[Q_{M_\ell}]$, for the Standard Method (in blue) computed for each time step used in the Multi Level Method which is of the form $h_\ell = 2^{-\ell}$. On the right Figure 4-9(b) shows the same expected values but for time steps of the form $h_\ell = 4^{-\ell}$. From these plots we see clearly that the Multi Level Method requires fewer levels but the expected value in both cases decays linearly, so that we still have $\alpha = 1$ in the complexity theorem.

71

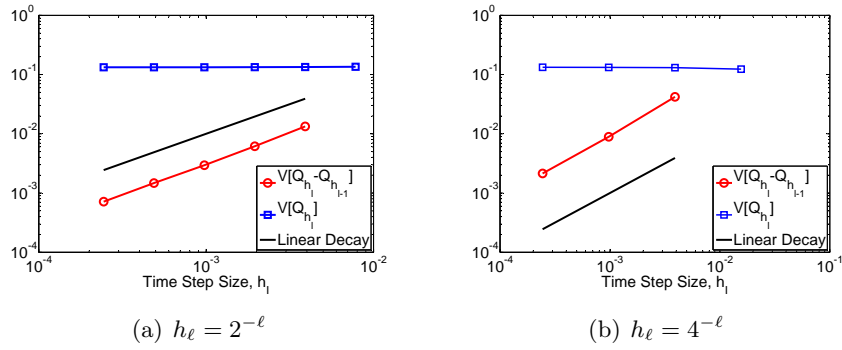(a) $h_\ell = 2^{-\ell}$ (b) $h_\ell = 4^{-\ell}$

Figure 4-10: Variance Plots for Varying Increment, with $\varepsilon = 10^{-3}$, $T = 1$

Similarly Figure 4-10(a) shows the variance of differences in the Multi Level Method (in red), i.e. $\mathbb{V}[Q_{M_\ell} - Q_{M_{\ell-1}}]$, as well as the variance, $\mathbb{E}[Q_{M_\ell}]$, for the Standard Method (in blue) computed for each time step used in the Multi Level Method which is of the form $h_\ell = 2^{-\ell}$. On the right, Figure 4-10(b) shows the same variances but for time steps of the form $h_\ell = 4^{-\ell}$. From these plots we see clearly that in both cases the variance decays linearly, consistent with $\beta = 1$ in the complexity theorem.

What we are most interested in when testing the different time step size increments is the affect this has on the cost of the method and in particular its speed-up over the Standard Method.

ε	h_0	h_L	Speed-up
10^{-2}	2^{-7}	2^{-10}	1.18
10^{-2}	4^{-3}	4^{-5}	1.39
10^{-3}	2^{-7}	2^{-12}	2.40
10^{-3}	4^{-3}	4^{-6}	3.11

Table 4.2: Speed-up of Multi Level MC over Standard MC for Differing Time Step Increments, QoI $= \chi_{a,b}$, $[a,b] = [0.1055, 0.1555]$, various ε, with $T = 1$

Table 4.2 compares directly the speed-up of the method for accuracy tolerances $\varepsilon = 10^{-2}$ and 10^{-3}, with the finest time step size the same for the increments in powers of 2 and of 4. We see for both tolerances that we obtain an improvement by using time steps of the form $h_\ell = 4^{-\ell}$ and for $\varepsilon = 10^{-3}$ the difference is a speed-up of 2.40 versus that of 3.11, which is quite a significant speed-up. As such it would be important to test these improvements more thoroughly as well as looking at other increments, in particular 6 or even 8, whilst these are still practically relevant for the accuracy

tolerances that we wish to achieve.

For the simplified problem we noted that performance of the Multi Level Method improved when we considered an interval in which particles were less likely to land. To briefly examine whether this remains the case when we move to the full, non-linear model we looked at the interval $[a, b] = [0.28, 0.45]$ and compared this to our more central interval $[a, b] = [0.1055.0.1555]$.

ε	Interval	h_0	h_L	Speed-up
10^{-1}	$[0.1055, 0.1555]$	2^{-6}	2^{-7}	0.56
10^{-1}	$[0.28, 0.45]$	2^{-6}	2^{-7}	0.89
10^{-2}	$[0.1055, 0.1555]$	2^{-7}	2^{-9}	0.90
10^{-2}	$[0.28, 0.45]$	2^{-7}	2^{-10}	2.26
10^{-3}	$[0.1055, 0.1555]$	2^{-7}	2^{-12}	2.40
10^{-3}	$[0.28, 0.45]$	2^{-7}	2^{-13}	8.84

Table 4.3: Speed-up of Multi Level MC over Standard MC for various ε, with $T = 1$ QoI $= \chi_{a,b}$, with varying interval $[a, b]$, Without Reflection

Table 4.3 compares the speed-up for the two different intervals for different accuracy tolerances, ε. We see that where the Multi Level Method seems to perform fairly poorly for $[a, b] = [0.1055.0.1555]$, not giving an improvement over the Standard, say for $\varepsilon = 10^{-2}$, then for $[a, b] = [0.28, 0.45]$ the Multi Level actually gives a speed-up of over 2. The fact that the probability of a particle landing in $[a, b] = [0.28, 0.45]$ at time $T = 1$ is smaller than for the other interval may be the reason behind this improvement and as such this should again be studied more carefully in order to give an accurate picture of the potential improvements that the implementation of Multi Level Monte Carlo has to offer this application to atmospheric dispersion modelling.

With what are generally optimistic results for the full model problem and gains evident over the Standard Method for reasonable levels of accuracy our next stage of testing is to implement reflection of particles that go beneath the ground in line with the method used by the Met Office.
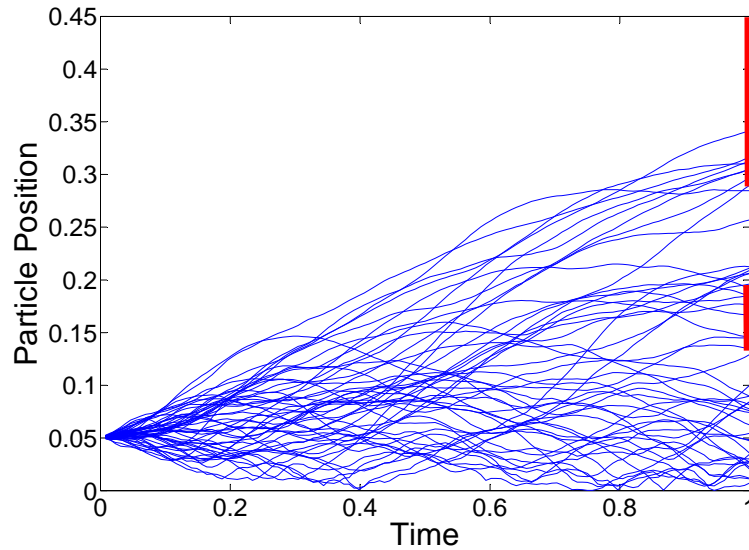
4.5.2   With Reflection at Boundaries

Figure 4-11: Full Model with Reflection Trajectory Plot for 50 Independent Particles
Released at $X_0 = 0.05$

In this section we examine the full model, for the 1-dimensional case with reflection
implemented as discussed in Section 4.4. Our quantity of interest is again the prob-
ability of a particle landing in an interval, $[a, b]$ at time T and our initial conditions
are $U_0 = 0.1$ and $X_0 = 0.05$ again corresponding to the release of a particle from a
chimney, say, at 50m above ground level, with velocity $0.2ms^{-1}$. For this problem we
have taken ε_{reg}, our regularisation parameter to be $\varepsilon_{reg} = 0.01$, and thus our coarsest
time step size is restricted to $h_0 = 2^{-5}$.

Figure 4-11 shows the path of 50 model particles and their spread over time, until
$T = 1$, where the particles are reflected about $X = 0$. Here the red blocks indicate
the intervals in which we look at the probability of a particle landing at time T. The
following are some initial results for tests with the reflection added to our full model.
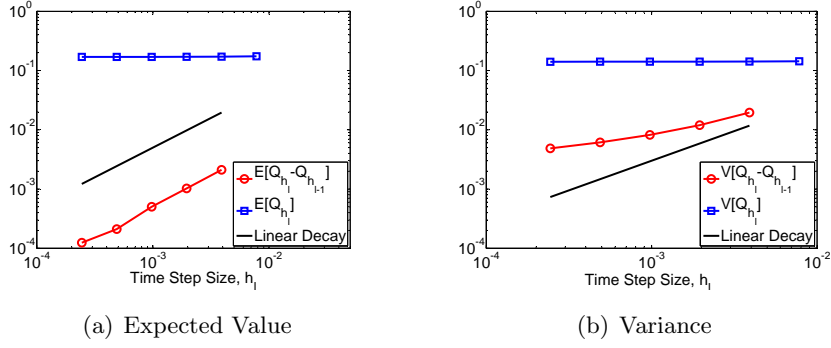
| (a) Expected Value | (b) Variance |

Figure 4-12: Expected Value and Variance for $[a, b] = [0.1055, 0.1555]$, $\varepsilon = 10^{-3}$, $T = 1$

Figure 4-12(a) shows the expected value of differences in the Multi Level Method with reflection (in red), i.e. $\mathbb{E}[Q_{M_\ell} - Q_{M_{\ell-1}}]$, as well as the expected value, $\mathbb{E}[Q_{M_\ell}]$, for the Standard Method with reflection (in blue) computed for each time step used in the Multi Level Method. We see that as before the expected value converges with $\mathcal{O}(h)$, corresponding to $\alpha = 1$ in the Standard and Multi Level complexity theorems.

Figure 4-12(b) shows the variance of differences in the Multi Level Method with reflection (in red), i.e. $\mathbb{V}[Q_{M_\ell} - Q_{M_{\ell-1}}]$, as well as the variance, $\mathbb{E}[Q_{M_\ell}]$, for the Standard Method with reflection (in blue) computed for each time step used in the Multi Level Method. What we notice here, however, is that the variance of the differences for the Multi Level Method no longer converges linearly and the rate is in fact slower than linear. This result somewhat restricts the success of the Multi Level Method in this case. We have carried out various tests of the code that we have written for this problem in order to ensure that these rates are not the result of a bug. It is thus crucial for the next steps in the continuing research of this topic to do further checks and try to establish the reasons for the slow decay rate of the variances, noting that the convergence of numerical approximations for SDEs with reflective boundary conditions has been investigated before, [27], [28].

In particular, the regularisation near the boundaries and indeed the regularisation parameter, ε_{reg}, may affect the model. In particular it may cause more particles to approach the ground and thus result in more particles being reflected than is physically likely. These are all possibilities that would require thorough study before we can make any conclusions about the potential for the full model with reflection.
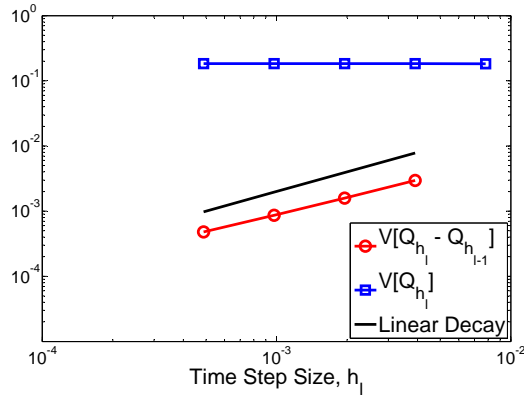
Figure 4-13: Variance for Model with Constant Profile, with $T = 1$, QoI $= \chi_{a,b}$, $[a, b] = [0.1055, 0.1555]$

Figure 4-13 shows the variance for the full model with a constant profile, i.e. we set both $\tau(X)$ and $\sigma_U(X)$ constant for all X. For this problem, by setting a counter, we observe that a smaller proportion of particles are reflected than for the full non-linear model. We note that convergence of the differences appears to be linear, although we note that for the finest time step we observe a slight slow down in this rate. Further tests would be vital for establishing whether this behaviour relates to the number of particles that are reflected, or perhaps the number of times a particle is reflected, in which case it may be that we need to more carefully choose our model parameters in particular ε_{reg} such that fewer particles require reflection at the boundary or indeed that we need a different method for reflecting or computing the velocity and trajectory of reflected particles.
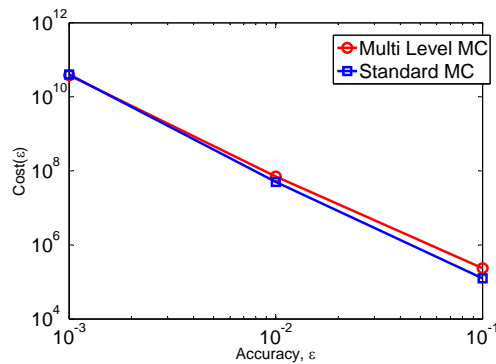

Figure 4-14: Cost of Standard MC vs Multi Level MC, with $T = 1$, QoI $= \chi_{a,b}$, $[a, b] = [0.1055, 0.1555]$

Plotting the cost of the Standard Method versus that of the Multi Level Method illustrates how the slow decay rate affects the performance of the Multi Level Method, comparing this to the cost plot without reflection in Figure 4-7(a) we see that instead of the Multi Level outperforming the standard for accuracy tolerance $\varepsilon = 5 \times 10^{-3}$ we now see that it is only just performing in line with the Standard Method for $\varepsilon = 10^{-3}$.

ε	Interval	Speed-up Without Reflection	Speed-up With Reflection
10^{-1}	$[0.1055, 0.1555]$	0.56	0.53
10^{-1}	$[0.28, 0.45]$	0.89	0.87
10^{-2}	$[0.1055, 0.1555]$	0.90	0.71
10^{-2}	$[0.28, 0.45]$	2.26	2.11
10^{-3}	$[0.1055, 0.1555]$	2.40	1.05
10^{-3}	$[0.28, 0.45]$	8.84	

Table 4.4: Speed-up of Multi Level MC over Standard MC for Method Without Reflection and With Reflection for varying intervals

Finally Table 4.4 gives a clear comparison of how the model with reflection performs in terms of speed-up compared to that without reflections. The table shows that for larger accuracy tolerances the affect of the reflection is not so evident, whereas for accuracy tolerance of 10^{-3} the speed-up from Multi Level over Standard is more than halved when the reflection is implemented. This may be as a result of the large number of reflected particles, which when counted a significant and perhaps not realistic proportion of particles were reflected at the boundary. As such further testing of this is a crucial next step in establishing the realistic benefits that could result from the implementation of the Multi Level Monte Carlo Method in this application to atmospheric dispersion modelling.

4.6   Conclusions

In this chapter we have identified that the Multi Level Method could provide an improvement over the Standard for the full model problem, but that its application to this problem in atmospheric dispersion modelling requires significant further work. Of great importance is the study of the impact that reflecting particles at the boundary has on the variance of the differences in the Multi Level Method and what changes can be made to the model or method in order to overcome this.

Overall, however, we have been able to show that for a model close to that which is currently used by the Met Office to model the dispersion of atmospheric pollutants,

we can achieve reasonable improvements in terms of the computational cost by the implementation of Multi Level Monte Carlo. In particular we have seen improvements for accuracy tolerances which are considered to be relatively large in mathematical terms, but are the practical tolerances required by the Met Office to ensure sufficiently accurate predictions. As such this is seen as a positive step towards improving the predictions in this application to atmospheric dispersion modelling.

Chapter 5

Further Work

In this thesis for both the simplified model problem as well as the full model problem we have shown that there is potential to gain in terms of computational cost as well as to allow the Met Office to model the dispersion of airborne pollutants more accurately, using more precise physics. That being said there are still a number of questions which this thesis has not yet answered and as such it poses a variety of interesting research problems. These include but are not limited to the following;

Reflection at Boundaries: After implementing the reflective boundary conditions in our full model we noted that this reflection adversely affected the decay rate of the variance of the differences for the Multi Level Method, as such it would be a priority in the continuation of this project to ensure that the reason for this increased variance is properly investigated so that we can establish what measures can be taken in order to prevent this or control the way in which it influences the success of the method. In particular a study of the literature in which this problem has already been investigated, such as [27] and [28], would provide an obvious starting point.

Regularisation Parameter: Another interesting question that arises from this thesis is that of how varying the regularisation parameter, ε_{reg}, with each level would affect the performance of the Multi Level Method. The choice of this parameter influences the stability constraint on the size of the time step we can take such that the method remains stable. So taking a smaller value of ε_{reg} will impose a tighter restriction on the size of the time step that will ensure we remain in the stability region of the method.

In the tests we have carried out so far, we have taken one value of ε_{reg} and chosen our coarsest time step in the Multi Level algorithm such that we ensure stability

on all levels, however an adaptive algorithm would allow us to choose our time steps for the method and then from this calculate the ε_{reg} we can take on each level such that the method remains stable.

To investigate this we would need to study how the error behaves with ε_{reg} and whether this adaptive method would allow us to gain by the use of smaller values of ε_{reg}; indeed whether this gives improved accuracy, or whether in fact our functions still diverge for a small regularisation parameter.

Different Time Step Increments: For the full model we looked briefly at how the size of the time step increment influences the performance. As such it would be important to further test the potential improvements seen by varying the time step size increment. In particular we note that in [9] it is discussed that increments of 7 give optimal results, thus testing these and other increments would be another option for further developments of the Multi Level Method in this context.

Further Smoothed Indicator Function Investigation: In Section 3.6 we carried out some preliminary tests for a smoothed indicator function, it would be useful to perform additional tests with this function or investigate a more advanced "smoothed" indicator function which may allow for further variance reduction or a reduced smoothing error and thus an improved performance.

Further Parameter Tests: Our numerical tests gave an interesting insight into how some of the model parameters influence the performance of our method. It would be important from an operational viewpoint to quantify these affects. In particular both the position and size of the box in which we estimate the probability of a particle landing are important to how well the Multi Level Method performs, as such quantifying this affect would be crucial improving the competitiveness of the method.

Longer Time Spans: For long time spans where the plume becomes wider relative to the interval $[a, b]$ a smaller fraction of particles land will land in our interval than for a shorter time span. Quantifying this behaviour could be useful for allowing longer scale models to run with increased accuracy.

Full Stability Analysis: For the full model we completed the theoretical stability analysis only for the ODE in equation (4.5). We carried out some preliminary numerical stability tests for the full model, however, to ensure that we are not in fact introducing any instability into our method, it would be beneficial to further

80

study the full problem; including the theoretical stability of the method at the top of the boundary layer resulting from the divergence of $\partial \sigma_U^2(X)/\partial X$.

Implicit Method for SDE: We also discussed that the stability constraints imposed by these numerical methods could be eliminated by implementing an implicit method for solving the SDE; this would be an interesting, although non-trivial problem to study.

Splitting Domain: When considering the probability of a particle landing in a box at time T, our quantity of interest was often small, as a result we therefore used the relative error to ensure that our approximations were sufficiently accurate. In studying this it was noted that the boxes in which we approximate the probability could be chosen in such a way that across the plume each of the different sized boxes will have approximately the same probability of a particle landing in it at time T. Thus our relative error will be the same across all boxes. This would be an interesting topic of investigation both for the Standard as well as the Multi Level Method.

Implementing Full Algorithm in NAME: Ultimately our aim is to provide an operational algorithm which we could implement in NAME, as such it would be important to integrate all steps of the method in order to give the best efficiency. This would, in particular, involve implementing the method as described in [9] and discussed in Section 3.5.2. In addition to this, it would be interesting to investigate the possibility of using the same samples on different levels of the Multi Level Method in order to reduce the cost from generating new samples, however this would involve some careful programming and analysis.

We have observed that there is great potential for the application of the Multi Level Monte Carlo Method, to enable better predictions on similar or even improved operational timescales for this problem in atmospheric dispersion modelling. Indeed with so many open questions emerging from this thesis there is a wealth of interesting options for the continuation of this project and ultimately we have a solid starting point in the argument for the implementation of Multi Level Monte Carlo in the Met Office's atmospheric model, NAME.

Bibliography

[1] C. S. Witham, M. C. Hort, R. Potts, R. Servranckx, P. Husson, F. Bonnardot. Comparison of VAAC atmospheric dispersion models using 1 November 2004 Grimsvotn eruption. *Meteorological Applications*, 14:27-38, 2007.

[2] S. J. Leadbetter, M. C. Hort, Volcanic ash hazard climatology for an eruption of Hekla Volcano, Iceland. *Journal of Volcanology and Geothermal Research*. 199:230-241, 2011.

[3] H. Dacre, A. Ansmann, I. Mattis, L. Clarisse, A. Grant, R.Hogan, S. Belcher, D. Thomson, B. Devenish, F. Marenco, M. Hort. Evaluating the structure and magnitude of the ash plume during the initial phase of the 2010 Eyjafjallajokull eruption using lidar observations and NAME simulations. *Journal Geophysical Research*. 116(20):1-13, 2011.

[4] A. Jones, D. Thomson, M. Hort, B. Devenish. The U.K. Met. Office's Next-Generation Atmospheric Dispersion Model, NAME III. *Air Pollution Modelling and Its Application*. XVII:580-589, 2007.

[5] A. Jones. Atmospheric dispersion modelling at the Met. Office. *Weather*. 59:311-316, 2004.

[6] R.H. Maryon, D.B. Ryall, A.L. Malcolm. The NAME 4 Dispersion Model: Science Documentation. *Met. Office Turbulence and Diffusion Note No. 262*. (unpublished), 1999.

[7] F. B. Smith, M. J. Clark. The transport and deposition of airborne debris from the Chernobyl nuclear power plant accident with special emphasis on the consequence to the United Kingdom. *Meteorological Office Scientific Paper No. 42*. 1989.

[8] W. Klug, G. Graziani, G. Grippa, D. Pierce, C. Tassone. Evaluation of long range atmospheric transport models using environmental radioactivity data from the Chernobyl accident: The ATMES report. *Elsevier Applied Science*. 1992.

[9] M. B. Giles. Multilevel Monte Carlo Path Simulations. *Operations Research.* 56(3):607-617, 2008.

[10] Benedict Leimkuhler and Sebastian Reich. *Simulating Hamiltonian dynamics*, volume 14 of *Monographs on Applied and Computational Mathematics.* Cambridge University Press, Cambridge, 2004.

[11] D. J. Thomson. Random walk modelling of diffusion in inhomogeneous turbulence. *Quarterly Journal of the Royal Meteorological Society.* 110(466):1107-1120, 1984.

[12] D. J. Thomson. Criteria for the selection of stochastic models of particle trajectories in turbulent flows. *Journal of Fluid Mechanics.* 180:529-556, 1987.

[13] N. Metropolis, S. Ulam. The Monte Carlo Method. *Journal of the American Statistical Association.* 44(247):335-341, 1949.

[14] M. E. J. Newman, G. T. Barkema. *Monte Carlo Methods in Statistical Physics.* Clarendon, Oxford, 1999.

[15] I. M. Sobol. *A primer for the Monte Carlo Method*, Taylor & Francis, London, 1994.

[16] Yu. A. Shreider, ed. *The Monte Carlo Method.* Pergamon Press, Oxford, 1966.

[17] Peter E. Kloeden and Eckhard Platen. *Numerical solution of stochastic differential equations*, volume 23 of *Applications of Mathematics (New York).* Springer-Verlag, Berlin, 1992.

[18] B. Devenish. Notes on Lagrangian stochastic models. (unpublished), 2012.

[19] H. C. Rodean. *Stochastic Lagrangian Models of Turbulent Diffusion, Meteorological Monographs.* Volume 26, no.48, American Meteorological Society, Boston, Mass, 1996.

[20] I. A. Tanski. Fundamental solution of Fokker-Planck equation. arXiv:nlin.SI/0407007, 2007.

[21] K. A. Cliffe, M. B. Giles, R. Scheichl, A. L. Teckentrup. Multilevel Monte Carlo Methods and Applications to Elliptic PDEs with Random Coefficients. *Multiscale Modeling and Simulation and Analysis* 9(4):1677-1699, 2011.

[22] D. Duffie, P. Glynn. Effiicient Monte Carlo simulations of security prices. *Annals of Applied Probability.* 5(4):897-905, 1995.

[23] H. N. Webster, E. B. Carroll, A. R. Jones, A. J. Manning and D. J. Thomson. The Buncefield oil depot incident: a discussion of the meteorology. *Weather.* 62(12):325-330, 2007.

[24] S. Cook, E. Mueller, R. Scheichl. Multi Level Monte Carlo Methods for Atmospheric Dispersion Modelling. (in preparation), 2013.

[25] K. Atkinson, W. Han. *Elementary Numerical Analysis.* John Wiley & Sons Inc. Pages 394-396, 2004.

[26] A. Iserles. *A First Course in the Numerical Analysis of Differential Equations.* Cambridge University Press. Pages 3-10, 1996.

[27] C. Costantini, B. Pacchiarotti and F. Sartoretto. Numerical approximation for functionals of reflecting diffusion processes. *SIAM J. Appl. Math.*, 58(1):73-102, 1998.

[28] G. N. Milstein and M. V. Tretyakov. A probabilistic approach to the solution of the Neumann problem for nonlinear parabolic equations. *IMA J. Numer. Anal.*, 22(4):599-622, 2002/

Appendix A

Appendix

A.1   Standard Monte Carlo Mean Square Error

The expected mean square error of our Standard Monte Carlo estimator, $\hat{Q}_{M,N}$, is defined as,

$$\mathbb{E}\left[\left(\hat{Q}_{M,N} - \mathbb{E}[Q]\right)^2\right].$$

By adding a zero, i.e. $\mathbb{E}[\hat{Q}_{M,N}] - \mathbb{E}[\hat{Q}_{M,N}]$, our mean square error becomes,

$$\mathbb{E}\left[\left(\hat{Q}_{M,N} - \mathbb{E}[Q]\right)^2\right] = \mathbb{E}\left[\left(\hat{Q}_{M,N} - \mathbb{E}[\hat{Q}_{M,N}] + \mathbb{E}[\hat{Q}_{M,N}] - \mathbb{E}[Q]\right)^2\right].$$

Now treating the first and second two terms separately and computing the square gives,

$$\mathbb{E}\left[\left(\hat{Q}_{M,N} - \mathbb{E}[\hat{Q}_{M,N}]\right)^2 + 2\left(\hat{Q}_{M,N} - \mathbb{E}[\hat{Q}_{M,N}]\right)\left(\mathbb{E}[\hat{Q}_{M,N}] - \mathbb{E}[Q]\right) + \left(\mathbb{E}[\hat{Q}_{M,N}] - \mathbb{E}[Q]\right)^2\right].$$

The linearity of the expected value allows us to write this as,

$$\mathbb{E}\left[\left(\hat{Q}_{M,N} - \mathbb{E}[\hat{Q}_{M,N}]\right)^2\right] + \mathbb{E}\left[\left(\mathbb{E}[\hat{Q}_{M,N}] - \mathbb{E}[Q]\right)^2\right] +$$

$$+ 2\mathbb{E}\left[\left(\hat{Q}_{M,N} - \mathbb{E}[\hat{Q}_{M,N}]\right)\left(\mathbb{E}[\hat{Q}_{M,N}] - \mathbb{E}[Q]\right)\right],$$

$$= \mathbb{E}\left[\left(\hat{Q}_{M,N} - \mathbb{E}[\hat{Q}_{M,N}]\right)^2\right] + \left(\mathbb{E}[\hat{Q}_{M,N}] - \mathbb{E}[Q]\right)^2 +$$

$$+ 2\Big(\underbrace{\mathbb{E}[\hat{Q}_{M,N}] - \mathbb{E}[\hat{Q}_{M,N}]}_{=0}\Big)\left(\mathbb{E}[\hat{Q}_{M,N}] - \mathbb{E}[Q]\right).$$

Here we are again using the linearity of expectation as well as the fact that $\mathbb{E}[\mathbb{E}[Y]] =$

$\mathbb{E}[Y]$ for some random variable Y.

Thus the mean square error can be written as,

$$
\begin{aligned}
\mathbb{E}\left[\left(\hat{Q}_{M,N} - \mathbb{E}[Q]\right)^2\right] &= \mathbb{E}\left[(\hat{Q}_{M,N} - \mathbb{E}[\hat{Q}_{M,N}])^2\right] + \left(\mathbb{E}[\hat{Q}_{M,N}] - \mathbb{E}[Q]\right)^2 \\
&= \mathbb{V}[\hat{Q}_{M,N}] + \left(\mathbb{E}[\hat{Q}_{M,N} - Q]\right)^2,
\end{aligned}
\tag{A.1}
$$

where the first term in the last line follows by definition of the variance for some random variable, Y, i.e. $\mathbb{V}[Y] = \mathbb{E}[(Y - \mathbb{E}[Y])^2]$.

Finally we observe that the expected value of our estimator $\hat{Q}_{M,N}$ is equal to that of our Euler approximation Q_M

$$
\mathbb{E}[\hat{Q}_{M,N}] = \mathbb{E}\left[\frac{1}{N}\sum_{i=1}^{N} Q_M^{(i)}\right] = \frac{1}{N}\sum_{i=1}^{N}\mathbb{E}[Q_M^{(i)}] = \frac{1}{N}\sum_{i=1}^{N}\mathbb{E}[Q_M] = \mathbb{E}[Q_M],
\tag{A.2}
$$

and that the variance of our Monte Carlo estimator $\hat{Q}_{M,N}$ is the variance of Q_M, scaled by $1/N$, since,

$$
\mathbb{V}[\hat{Q}_{M,N}] = \mathbb{V}\left[\frac{1}{N}\sum_{i=1}^{N} Q_M^{(i)}\right] = \frac{1}{N^2}\sum_{i=1}^{N}\mathbb{V}[Q_M^{(i)}] = \frac{1}{N^2}\sum_{i=1}^{N}\mathbb{V}[Q_M] = \frac{1}{N}\mathbb{V}[Q_M].
\tag{A.3}
$$

By combining the results in equations (A.5), (A.2) and (A.3) we can now write our mean square error as,

$$
\mathbb{E}\left[\left(\hat{Q}_{M,N} - \mathbb{E}[Q]\right)^2\right] = \underbrace{N^{-1}\mathbb{V}[Q_M]}_{\text{sampling error}} + \underbrace{\left(\mathbb{E}[Q_M - Q]\right)^2}_{\text{discretisation error}},
\tag{A.4}
$$

giving a contribution from both the sampling error, and the discretisation error.

A.2 Multi Level Monte Carlo Mean Square Error

The expected mean square error of our Multi Level Monte Carlo estimator, $\hat{Q}_{M_L,\{N_\ell\}}^{(ML)}$, is defined as,

$$
\mathbb{E}\left[\left(\hat{Q}_{M_L,\{N_\ell\}}^{(ML)} - \mathbb{E}[Q]\right)^2\right].
$$

By adding a zero, i.e. $\mathbb{E}\left[\hat{Q}_{M_L,\{N_\ell\}}^{(ML)}\right] - \mathbb{E}\left[\hat{Q}_{M_L,\{N_\ell\}}^{(ML)}\right]$, our mean square error becomes,

$$\mathbb{E}\left[\left(\hat{Q}_{M_L,\{N_\ell\}}^{(ML)} - \mathbb{E}[Q]\right)^2\right] = \mathbb{E}\left[\left(\hat{Q}_{M_L,\{N_\ell\}}^{(ML)} - \mathbb{E}\left[\hat{Q}_{M_L,\{N_\ell\}}^{(ML)}\right] + \mathbb{E}\left[\hat{Q}_{M_L,\{N_\ell\}}^{(ML)}\right] - \mathbb{E}[Q]\right)^2\right].$$

Now treating the first and second two terms separately and computing the square gives,

$$\mathbb{E}\left[\left(\hat{Q}_{M_L,\{N_\ell\}}^{(ML)} - \mathbb{E}\left[\hat{Q}_{M_L,\{N_\ell\}}^{(ML)}\right]\right)^2 + 2\left(\hat{Q}_{M_L,\{N_\ell\}}^{(ML)} - \mathbb{E}\left[\hat{Q}_{M_L,\{N_\ell\}}^{(ML)}\right]\right)\left(\mathbb{E}\left[\hat{Q}_{M_L,\{N_\ell\}}^{(ML)}\right] - \mathbb{E}[Q]\right)\right.$$

$$\left. + \left(\mathbb{E}[\hat{Q}_{M,N}] - \mathbb{E}[Q]\right)^2\right].$$

The linearity of the expected value allows us to write this as,

$$\mathbb{E}\left[\left(\hat{Q}_{M_L,\{N_\ell\}}^{(ML)} - \mathbb{E}\left[\hat{Q}_{M_L,\{N_\ell\}}^{(ML)}\right]\right)^2\right] + \mathbb{E}\left[\left(\mathbb{E}\left[\hat{Q}_{M_L,\{N_\ell\}}^{(ML)}\right] - \mathbb{E}[Q]\right)^2\right] +$$

$$+ 2\mathbb{E}\left[\left(\hat{Q}_{M_L,\{N_\ell\}}^{(ML)} - \mathbb{E}\left[\hat{Q}_{M_L,\{N_\ell\}}^{(ML)}\right]\right)\left(\mathbb{E}\left[\hat{Q}_{M_L,\{N_\ell\}}^{(ML)}\right] - \mathbb{E}[Q]\right)\right],$$

$$= \mathbb{E}\left[\left(\hat{Q}_{M_L,\{N_\ell\}}^{(ML)} - \mathbb{E}\left[\hat{Q}_{M_L,\{N_\ell\}}^{(ML)}\right]\right)^2\right] + \left(\mathbb{E}\left[\hat{Q}_{M_L,\{N_\ell\}}^{(ML)}\right] - \mathbb{E}[Q]\right)^2 +$$

$$+ 2\underbrace{\left(\mathbb{E}\left[\hat{Q}_{M_L,\{N_\ell\}}^{(ML)}\right] - \mathbb{E}\left[\hat{Q}_{M_L,\{N_\ell\}}^{(ML)}\right]\right)}_{=0}\left(\mathbb{E}\left[\hat{Q}_{M_L,\{N_\ell\}}^{(ML)}\right] - \mathbb{E}[Q]\right).$$

Here we are again using the linearity of expectation as well as the fact that $\mathbb{E}[\mathbb{E}[Y]] = \mathbb{E}[Y]$ for some random variable Y.

Thus the mean square error can be written as,

$$\mathbb{E}\left[\left(\hat{Q}_{M_L,\{N_\ell\}}^{(ML)} - \mathbb{E}[Q]\right)^2\right] = \mathbb{E}\left[\left(\hat{Q}_{M_L,\{N_\ell\}}^{(ML)} - \mathbb{E}\left[\hat{Q}_{M_L,\{N_\ell\}}^{(ML)}\right]\right)^2\right] + \left(\mathbb{E}\left[\hat{Q}_{M_L,\{N_\ell\}}^{(ML)}\right] - \mathbb{E}[Q]\right)^2$$

$$= \mathbb{V}\left[\hat{Q}_{M_L,\{N_\ell\}}^{(ML)}\right] + \left(\mathbb{E}\left[\hat{Q}_{M_L,\{N_\ell\}}^{(ML)} - Q\right]\right)^2,$$

(A.5)

where the first term in the last line follows by definition of the variance for some random variable, Y, i.e. $\mathbb{V}[Y] = \mathbb{E}[(Y - \mathbb{E}[Y])^2]$.

We observe that the expected value of our estimator, $\hat{Q}_{M_L,\{N_\ell\}}^{(ML)}$, is equal to that of our Euler approximation on the finest level, Q_{M_L}, since by the definition of our estimator we have,

$$\mathbb{E}\left[\hat{Q}^{(ML)}_{M_L,\{N_\ell\}}\right] \quad = \quad \mathbb{E}\left[\sum_{\ell=0}^{L}\hat{Y}_{\ell,\{N_\ell\}}\right] = \sum_{\ell=0}^{L}\mathbb{E}[\hat{Y}_{\ell,\{N_\ell\}}] = \sum_{\ell=0}^{L}\frac{1}{N_\ell}\sum_{i=1}^{N_\ell}\mathbb{E}[Y_\ell^{(i)}]$$

$$= \quad \sum_{\ell=0}^{L}\frac{1}{N_\ell}\sum_{i=1}^{N_\ell}\mathbb{E}[Y_\ell] = \sum_{\ell=0}^{L}\frac{1}{N_\ell}N_\ell\mathbb{E}[Y_\ell] = \sum_{\ell=0}^{L}\mathbb{E}[Y_\ell].$$

(A.6)

Finally $\sum_{\ell=0}^{L}\mathbb{E}[Y_\ell] = \mathbb{E}[Q_{M_L}]$ from equation (2.6) so we have that,

$$\mathbb{E}\left[\hat{Q}^{(ML)}_{M_L,\{N_\ell\}}\right] = \mathbb{E}[Q_{M_L}],\tag{A.7}$$

i.e. that the discretisation error of the Multi Level method is that corresponding the finest time step h_L, with no contributions from any other levels.

Also by the definition of our estimator and properties of the variance we observe that,

$$\mathbb{V}\left[\hat{Q}^{(ML)}_{M_L,\{N_\ell\}}\right] \quad = \quad \mathbb{V}\left[\sum_{\ell=0}^{L}\hat{Y}_{\ell,\{N_\ell\}}\right] = \sum_{\ell=0}^{L}\mathbb{V}[\hat{Y}_{\ell,\{N_\ell\}}] = \sum_{\ell=0}^{L}\frac{1}{N_\ell^2}\sum_{i=1}^{N_\ell}\mathbb{V}[Y_\ell^{(i)}]$$

$$= \quad \sum_{\ell=0}^{L}\frac{1}{N_\ell^2}\sum_{i=1}^{N_\ell}\mathbb{V}[Y_\ell] = \sum_{\ell=0}^{L}\frac{1}{N_\ell^2}N_\ell\mathbb{E}[Y_\ell] = \sum_{\ell=0}^{L}\frac{1}{N_\ell}\mathbb{V}[Y_\ell].$$

(A.8)

Substituting the results in equations (A.7) and (A.8) gives our final formula for the Multi Level MSE,

$$\mathbb{E}\left[\left(\hat{Q}^{(ML)}_{M_L,\{N_\ell\}} - \mathbb{E}[Q]\right)^2\right] = \underbrace{\sum_{\ell=0}^{L}\frac{1}{N_\ell}\mathbb{V}[Y_\ell]}_{\text{Sampling Error}} + \underbrace{\left(\mathbb{E}[Q_{M_L} - Q]\right)^2}_{\text{Discretisation Error}}.\tag{A.9}$$

A.3   Smoothed indicator Mean Square Error

The expected mean square error of our Smoothed Monte Carlo estimator, $\hat{Q}^\lambda_{M,N}$, is defined as,

$$\mathbb{E}\left[\left(\hat{Q}^\lambda_{M,N} - \mathbb{E}[Q]\right)^2\right].$$

Adding two zeros, $\mathbb{E}[\hat{Q}^\lambda_{M,N}] - \mathbb{E}[\hat{Q}^\lambda_{M,N}]$ and $\mathbb{E}[\hat{Q}_{M,N}] - \mathbb{E}[\hat{Q}_{M,N}]$, our mean square

error becomes,

$$\mathbb{E}\left[\left(\hat{Q}^{\lambda}_{M,N} - \mathbb{E}[\hat{Q}^{\lambda}_{M,N}] + \mathbb{E}[\hat{Q}^{\lambda}_{M,N}] - \mathbb{E}[\hat{Q}_{M,N}] + \mathbb{E}[\hat{Q}_{M,N}] - \mathbb{E}[Q]\right)^2\right].$$

Treating the $\left(\hat{Q}^{\lambda}_{M,N} - \mathbb{E}[\hat{Q}^{\lambda}_{M,N}]\right)$, $\left(\mathbb{E}[\hat{Q}^{\lambda}_{M,N}] - \mathbb{E}[\hat{Q}_{M,N}]\right)$ and $\left(\mathbb{E}[\hat{Q}_{M,N}] - \mathbb{E}[Q]\right)$ terms separately and computing the square, our mean square error becomes,

$$\mathbb{E}\left[\left(\hat{Q}^{\lambda}_{M,N} - \mathbb{E}[\hat{Q}^{\lambda}_{M,N}]\right)^2 + \left(\mathbb{E}[\hat{Q}^{\lambda}_{M,N}] - \mathbb{E}[\hat{Q}_{M,N}]\right)^2 + \left(\mathbb{E}[\hat{Q}_{M,N}] - \mathbb{E}[Q]\right)^2 + \right.$$

$$+2\left(\hat{Q}^{\lambda}_{M,N} - \mathbb{E}[\hat{Q}^{\lambda}_{M,N}]\right)\left(\mathbb{E}[\hat{Q}^{\lambda}_{M,N}] - \mathbb{E}[\hat{Q}_{M,N}]\right) +$$

$$+2\left(\hat{Q}^{\lambda}_{M,N} - \mathbb{E}[\hat{Q}^{\lambda}_{M,N}]\right)\left(\mathbb{E}[\hat{Q}_{M,N}] - \mathbb{E}[Q]\right) +$$

$$\left. +2\left(\mathbb{E}[\hat{Q}^{\lambda}_{M,N}] - \mathbb{E}[\hat{Q}_{M,N}]\right)\left(\mathbb{E}[\hat{Q}_{M,N}] - \mathbb{E}[Q]\right)\right].$$

The linearity of the expected value and the fact that $\mathbb{E}[\mathbb{E}[Y]] = \mathbb{E}[Y]$ for some random variable Y, allows us to write this as,

$$= \mathbb{E}\left[\left(\hat{Q}^{\lambda}_{M,N} - \mathbb{E}[\hat{Q}^{\lambda}_{M,N}]\right)^2\right] + \mathbb{E}\left[\left(\mathbb{E}[\hat{Q}^{\lambda}_{M,N}] - \mathbb{E}[\hat{Q}_{M,N}]\right)^2\right] + \mathbb{E}\left[\left(\mathbb{E}[\hat{Q}_{M,N}] - \mathbb{E}[Q]\right)^2\right] +$$

$$+2\Big(\underbrace{\mathbb{E}[\hat{Q}^{\lambda}_{M,N}] - \mathbb{E}[\hat{Q}^{\lambda}_{M,N}]}_{=\,0}\Big)\left(\mathbb{E}[\hat{Q}^{\lambda}_{M,N}] - \mathbb{E}[\hat{Q}_{M,N}]\right) +$$

$$+2\Big(\underbrace{\mathbb{E}[\hat{Q}^{\lambda}_{M,N}] - \mathbb{E}[\hat{Q}^{\lambda}_{M,N}]}_{=\,0}\Big)\left(\mathbb{E}[\hat{Q}_{M,N}] - \mathbb{E}[Q]\right) +$$

$$+2\left(\mathbb{E}[\hat{Q}^{\lambda}_{M,N}] - \mathbb{E}[\hat{Q}_{M,N}]\right)\left(\mathbb{E}[\hat{Q}_{M,N}] - \mathbb{E}[Q]\right).$$

Thus the mean square error can be written as,

$$
\begin{aligned}
\mathbb{E}\left[\left(\hat{Q}^\lambda_{M,N} - \mathbb{E}[Q]\right)^2\right] &= \mathbb{E}\left[\left(\hat{Q}^\lambda_{M,N} - \mathbb{E}[\hat{Q}^\lambda_{M,N}]\right)^2\right] + \left(\mathbb{E}[\hat{Q}^\lambda_{M,N}] - \mathbb{E}[\hat{Q}_{M,N}]\right)^2 + \\
&\quad + \left(\mathbb{E}[\hat{Q}_{M,N}] - \mathbb{E}[Q]\right)^2 + \\
&\quad + 2\left(\mathbb{E}[\hat{Q}^\lambda_{M,N}] - \mathbb{E}[\hat{Q}_{M,N}]\right)\left(\mathbb{E}[\hat{Q}_{M,N}] - \mathbb{E}[Q]\right) \\
&= \underbrace{\mathbb{V}[\hat{Q}^\lambda_{M,N}]}_{\text{sampling error}} + \underbrace{\left(\mathbb{E}[Q_M] - \mathbb{E}[Q]\right)^2}_{\text{discretisation error}} + \\
&\quad + \underbrace{\left(\mathbb{E}[Q^\lambda_M] - \mathbb{E}[Q_M]\right)^2}_{\text{smoothing error}} + \underbrace{2\left(\mathbb{E}[Q^\lambda_M] - \mathbb{E}[Q_M]\right)\left(\mathbb{E}[Q_M] - \mathbb{E}[Q]\right)}_{\text{Cross Term}},
\end{aligned}
$$

$$(A.10)$$

which gives a contribution from the sampling error and discretisation error as well as the smoothing error and a cross term involving the smoothing error as well as the discretisation error.

A.4 Euler Induction

For our simplified model the Euler method is given by,

$$
\begin{aligned}
U_{n+1} &= \left(1 - \frac{h}{\tau}\right)U_n + \sqrt{\frac{2\sigma_U^2 h}{\tau}}\Delta W_{n+1} \\
X_{n+1} &= X_n + U_n h.
\end{aligned}
$$

$$(A.11)$$

We prove by induction that U_M and X_M can be written as follows,

$$
U_M = (U_0 + \sigma_U^* \Delta W_0)\left(1 - \frac{h}{\tau}\right)^M + \sqrt{\frac{2\sigma_U^2 h}{\tau}}\sum_{k=1}^{M}\left(1 - \frac{h}{\tau}\right)^{M-k}\Delta W_k \qquad (A.12)
$$

$$X_M = \underbrace{X_0 + U_0 h \sum_{n=0}^{M-1}\left(1 - \frac{h}{\tau}\right)^n +}_{\text{Deterministic Part}}$$

$$\underbrace{+ \left(\sigma_U^* h \sum_{n=0}^{M-1}\left(1 - \frac{h}{\tau}\right)^n\right)\Delta W_0 + h\sqrt{\frac{2\sigma_U^2 h}{\tau}} \sum_{n=0}^{M-1}\sum_{k=1}^{n}\left(1 - \frac{h}{\tau}\right)^{n-k}\Delta W_k}_{\text{Stochastic Part}}.$$

(A.13)

First we show that our formula holds for the base case, the formula for Euler's method gives,

$$U_1 = (U_0 + \sigma_U^*\Delta W_0)\left(1 - \frac{h}{\tau}\right) + \sqrt{\frac{2\sigma_U^2 h}{\tau}}\Delta W_1, \qquad\qquad (A.14)$$

where we are assuming that the particles start with some additional turbulent component, $\sigma_U^*\Delta W_0$, to their initial velocity U_0, to be consistent with the results in [18].

Comparing this to the U_1 computed with our formula for U_M, which is given by,

$$U_1 = (U_0 + \sigma_U^*\Delta W_0)\left(1 - \frac{h}{\tau}\right)^1 + \sqrt{\frac{2\sigma_U^2 h}{\tau}} \sum_{k=1}^{1}\left(1 - \frac{h}{\tau}\right)^{1-k}\Delta W_k$$

(A.15)

$$= (U_0 + \sigma_U^*\Delta W_0)\left(1 - \frac{h}{\tau}\right) + \sqrt{\frac{2\sigma_U^2 h}{\tau}}\Delta W_1,$$

we see that our formulas for U_1 in equations (A.14) and (A.15) agree and thus our inductive hypothesis holds for $M = 1$.

Now we assume that our formula for U holds for M and show that it holds for $M + 1$. By Euler's formula,

$$U_{M+1} = \left(1 - \frac{h}{\tau}\right)U_M + \sqrt{\frac{2\sigma_U^2 h}{\tau}}\Delta W_{M+1}$$

$$= \left(1 - \frac{h}{\tau}\right)\left[(U_0 + \sigma_U^*\Delta W_0)\left(1 - \frac{h}{\tau}\right)^M + \sqrt{\frac{2\sigma_U^2 h}{\tau}} \sum_{k=1}^{M}\left(1 - \frac{h}{\tau}\right)^{M-k}\Delta W_k\right] +$$

$$+ \sqrt{\frac{2\sigma_U^2 h}{\tau}}\Delta W_{M+1},$$

by substituting in the formula for U_M.

Rearranging gives,

$$
\begin{aligned}
U_{M+1} &= (U_0 + \sigma_U^* \Delta W_0)\left(1 - \frac{h}{\tau}\right)^{M+1} + \sqrt{\frac{2\sigma_U^2 h}{\tau}} \sum_{k=1}^{M} \left(1 - \frac{h}{\tau}\right)^{M-k} \Delta W_k + \\
&\quad + \sqrt{\frac{2\sigma_U^2 h}{\tau}}\left(1 - \frac{h}{\tau}\right)^0 \Delta W_{M+1} \\
&= (U_0 + \sigma_U^* \Delta W_0)\left(1 - \frac{h}{\tau}\right)^{M+1} + \sqrt{\frac{2\sigma_U^2 h}{\tau}} \sum_{k=1}^{M+1} \left(1 - \frac{h}{\tau}\right)^{M+1-k} \Delta W_k,
\end{aligned}
$$

so U_{M+1} is of the form of (A.13) and thus we have shown that if the formula holds for M, it holds for $M + 1$ and thus by the principle of mathematical induction our formula holds for all $M \geq 1$.

Now we must also prove that the formula for X_M holds for all M, so for the base case, the formula for Euler's method gives,

$$
X_1 \;\; = \;\; X_0 + (U_0 + \sigma_U^* \Delta W_0)h \tag{A.16}
$$

where we are again assuming that the particles start with some additional turbulent component, $\sigma_U^* \Delta W_0$, to their initial velocity U_0.

We now compare this with our formula for X_M, which gives,

$$
\begin{aligned}
X_1 &= X_0 + U_0 h \sum_{n=0}^{1-1} \left(1 - \frac{h}{\tau}\right)^n + \left(\sigma_U^* h \sum_{n=0}^{1-1} \left(1 - \frac{h}{\tau}\right)^n\right) \Delta W_0 + \\
&\qquad + h\sqrt{\frac{2\sigma_U^2 h}{\tau}} \sum_{n=0}^{1-1}\sum_{k=1}^{n} \left(1 - \frac{h}{\tau}\right)^{n-k} \Delta W_k \\
&= X_0 + U_0 h \underbrace{\left(1 - \frac{h}{\tau}\right)^0}_{=1} + \left(\sigma_U^* h \underbrace{\left(1 - \frac{h}{\tau}\right)^0}_{=1}\right) \Delta W_0 + \\
&\qquad + h\sqrt{\frac{2\sigma_U^2 h}{\tau}} \underbrace{\sum_{n=0}^{1-1}\sum_{k=1}^{n} \left(1 - \frac{h}{\tau}\right)^{n-k} \Delta W_k}_{=0}
\end{aligned}
\tag{A.17}
$$

$$X_1 = X_0 + (U_0 + \sigma_U^* \Delta W_0)h, \tag{A.18}$$

by evaluating the sums, so our formulas in (A.16) and (A.18) agree and thus our inductive hypothesis holds for $M = 1$.

Now we assume that our formula for X holds for M and show that it holds for $M + 1$. By Euler's formula,

$$X_{M+1} = X_M + U_M h$$

$$= X_M + h \left[U_0 + \sigma_U^* \Delta W_0) \left(1 - \frac{h}{\tau}\right)^M + \sqrt{\frac{2\sigma_U^2 h}{\tau}} \sum_{k=1}^{M} \left(1 - \frac{h}{\tau}\right)^{M-k} \Delta W_k \right],$$

by substituting in our formula for U_M, which we have proved to be true. Now we substitute in our formula X_M which we are assuming to be true.

$$X_{M+1} = X_0 + U_0 h \sum_{n=0}^{M-1} \left(1 - \frac{h}{\tau}\right)^n + \left(\sigma_U^* h \sum_{n=0}^{M-1} \left(1 - \frac{h}{\tau}\right)^n\right) \Delta W_0$$

$$+ h \sqrt{\frac{2\sigma_U^2 h}{\tau}} \sum_{n=0}^{M-1} \sum_{k=1}^{n} \left(1 - \frac{h}{\tau}\right)^{n-k} \Delta W_k.$$

$$+ h \left[U_0 + \sigma_U^* \Delta W_0) \left(1 - \frac{h}{\tau}\right)^M + \sqrt{\frac{2\sigma_U^2 h}{\tau}} \sum_{k=1}^{M} \left(1 - \frac{h}{\tau}\right)^{M-k} \Delta W_k \right].$$

Rearranging and grouping terms gives,

$$X_{M+1} = X_0 + U_0 h \sum_{n=0}^{(M+1)-1} \left(1 - \frac{h}{\tau}\right)^n + \left(\sigma_U^* h \sum_{n=0}^{(M+1)-1} \left(1 - \frac{h}{\tau}\right)^n\right) \Delta W_0$$

$$+ h \sqrt{\frac{2\sigma_U^2 h}{\tau}} \sum_{n=0}^{(M+1)-1} \sum_{k=1}^{n} \left(1 - \frac{h}{\tau}\right)^{n-k} \Delta W_k. \tag{A.19}$$

so X_{M+1} is of the form of (A.13) thus we have shown that if the formula holds for M it holds for $M + 1$ and thus by the principle of mathematical induction our formula for X holds for all $M \geq 1$.

A.5 Expected Value and Variance Calculations

Here we derive for the simplified problem the explicit formulas for the expected value, $\mathbb{E}[X_M]$ and the variance, $\mathbb{V}[X_M]$.

If we have, $\Delta W_0, \ldots, \Delta W_M$, normally distributed random variables with mean 0 and variance 1. Then defining S_M as the sum,

$$S_M = \bar{a} + \sum_{n=0}^{M-1} a_n \Delta W_n,$$

we have that S_M is distributed according to a Gaussian with mean $\mathbb{E}[S_M] = \bar{a}$ and variance $\sigma_U^2 = \mathbb{V}[S_M] = \sum_{n=0}^{M-1} a_n^2$.

We have the following formula for X_M,

$$
\begin{aligned}
X_M \;=\; & X_0 + U_0 h \sum_{n=0}^{M-1} \left(1 - \frac{h}{\tau}\right)^n + \left(\sigma_U^* h \sum_{n=0}^{M-1} \left(1 - \frac{h}{\tau}\right)^n\right) \Delta W_0 + \\
& + h \sqrt{\frac{2\sigma_U^2 h}{\tau}} \sum_{n=0}^{M-1} \sum_{k=1}^{n} \left(1 - \frac{h}{\tau}\right)^{n-k} \Delta W_k,
\end{aligned}
\tag{A.20}
$$

which is of the form of S_M, as such we use the above the result to compute the expected value and variance of X_M. So,

$$
\begin{aligned}
\mathbb{E}[X_M] \;&= X_0 + U_0 h \sum_{n=0}^{M-1} \left(1 - \frac{h}{\tau}\right)^n && \text{since } \mathbb{E}[\Delta W_i] = 0 \;\; \forall i \\[2mm]
&= X_0 + U_0 h \frac{1 - (1 - \frac{h}{\tau})^M}{1 - (1 - \frac{h}{\tau})} && \text{using } \sum_{n=0}^{M-1} y^n = \frac{1 - y^M}{1 - y} && \text{(A.21)} \\[2mm]
&= X_0 + U_0 \tau \left[1 - \left(1 - \frac{h}{\tau}\right)^M\right] \equiv \mu_h,
\end{aligned}
$$

where $\mu_h := \mu_0 + \frac{h}{\tau}.\delta\mu + \mathcal{O}\left(\left(\frac{h}{\tau}\right)^2\right)$.

We obtain the form of μ_0 and $\delta\mu$ by the following. Since $h = T/M$, in the limit as $h \to 0$ we have,

$$\left(1 - \frac{h}{\tau}\right)^M = \left(1 - \frac{T}{M\tau}\right)^M \to \exp\left(-\frac{T}{\tau}\right).$$

We also know that,

94

$$\log\left(1 - \frac{h}{\tau}\right)^M = M\log\left(1 - \frac{h}{\tau}\right) = M\left(-\frac{h}{\tau} - \frac{1}{2}\left(\frac{h}{\tau}\right)^2 + \dots\right), \tag{A.22}$$

by Taylor expansion of $\log(z)$, which gives, $\log(z) = (z-1) - \frac{(z-1)^2}{2} + \dots$

Taking the exponential of both sides gives,

$$\left(1 - \frac{h}{\tau}\right)^M = \exp\left(-\frac{Mh}{\tau} - \frac{M}{2}\left(\frac{h}{\tau}\right)^2 + \dots\right) = \exp\left(-\frac{T}{\tau} - \frac{1}{2}\cdot\frac{T}{\tau}\cdot\frac{h}{\tau} + \dots\right),$$

where the last equation follows since $Mh = T$.

Using the elementary property of exponentials that $\exp(z + y) = \exp(z)\exp(y)$, we can write,

$$\left(1 - \frac{h}{\tau}\right)^M = \exp\left(-\frac{T}{\tau} - \frac{1}{2}\cdot\frac{T}{\tau}\cdot\frac{h}{\tau} + \dots\right) = \exp\left[\frac{-T}{\tau}\right]\exp\left(-\frac{T}{2\tau}\cdot\frac{h}{\tau} + \dots\right)$$

$$= \exp\left[\frac{-T}{\tau}\right]\left(1 - \frac{T}{2\tau}\cdot\frac{h}{\tau} + \dots\right) \quad \text{using } \exp(z) = \sum_{n=0}^{\infty}\frac{z^n}{n!}. \tag{A.23}$$

Substituting this into our formula for $\mathbb{E}[X_M]$, we have,

$$\mathbb{E}[X_M] = X_0 + U_0\tau\left[1 - \left(1 - \frac{h}{\tau}\right)^M\right]$$

$$= X_0 + U_0\tau\left[1 - \exp\left[\frac{-T}{\tau}\right]\left(1 - \frac{T}{2\tau}\cdot\frac{h}{\tau} + \dots\right)\right],$$

and rearranging terms gives,

$$\mathbb{E}[X_M] = X_0 + U_0\tau\left[1 - \exp\left[\frac{-T}{\tau}\right]\right] + U_0\tau\left[\exp\left[\frac{-T}{\tau}\right]\frac{T}{2\tau}\cdot\frac{h}{\tau} + \dots\right]$$

$$= X_0 + U_0\tau\left[1 - \exp\left[\frac{-T}{\tau}\right]\right] + \frac{U_0 T}{2}\exp\left[\frac{-T}{\tau}\right]\frac{h}{\tau} + \dots$$

So we have $\mu_0 = X_0 + U_0\tau\left[1 - \exp\left[\frac{-T}{\tau}\right]\right]$ and $\delta\mu = \frac{U_0 T}{2}\exp\left[\frac{-T}{\tau}\right]$.

To compute the variance of X_M, $\mathbb{V}[X_M]$, we begin by rearranging and simplifying the double sum in the stochastic term in equation (A.24), so we have,

$$h\sqrt{\frac{2\sigma_U^2 h}{\tau}} \sum_{n=0}^{M-1} \sum_{k=1}^{n} \left(1 - \frac{h}{\tau}\right)^{n-k} \Delta W_k \quad = h\sqrt{\frac{2\sigma_U^2 h}{\tau}} \sum_{k=1}^{M-1} \sum_{n=k}^{M-1} \left(1 - \frac{h}{\tau}\right)^{n-k} \Delta W_k$$

$$= h\sqrt{\frac{2\sigma_U^2 h}{\tau}} \sum_{k=1}^{M-1} \left(1 - \frac{h}{\tau}\right)^{-k} \Delta W_k \sum_{n=k}^{M-1} \left(1 - \frac{h}{\tau}\right)^{n}$$

$$= h\sqrt{\frac{2\sigma_U^2 h}{\tau}} \sum_{k=1}^{M-1} \frac{\tau}{h} \left[\left(1 - \frac{h}{\tau}\right)^{k} - \left(1 - \frac{h}{\tau}\right)^{M}\right] \Delta W_k,$$

where the last line follows by writing the sum from k to $M-1$ as the difference between the sum from 1 to $M-1$ and the sum from 1 to $k-1$, i.e.

$$\sum_{n=k}^{M-1} y^n = \sum_{n=1}^{M-1} y^n - \sum_{n=1}^{k-1} y^n = \frac{1 - y^M}{1 - y} - \frac{1 - y^k}{1 - y} = \frac{y^k - y^M}{1 - y}.$$

Thus the stochastic term above can be written as,

$$h\sqrt{\frac{2\sigma_U^2 h}{\tau}} \sum_{k=1}^{M-1} \frac{\tau}{h} \left[\left(1 - \frac{h}{\tau}\right)^{k} - \left(1 - \frac{h}{\tau}\right)^{M}\right] \Delta W_k \quad = \sqrt{2\sigma_U^2 \tau h} \sum_{k=1}^{M-1} \left[1 - \left(1 - \frac{h}{\tau}\right)^{M-k}\right] \Delta W_k$$

$$= \sqrt{2\sigma_U^2 \tau h} \sum_{k=1}^{M-1} \left[1 - \left(1 - \frac{h}{\tau}\right)^{k}\right] \Delta \tilde{W}_k,$$

where $\Delta \tilde{W}_k = \Delta W_{M-k}$ and the last line simply represents a change the order in which the sum is computed.

Now we can use the result discussed at the beginning of this section, that for a random variable with the form of $S_M = \bar{a} + \sum_{n=0}^{M-1} a_n \Delta W_n$, the variance is given by, $\mathbb{V}[S_M] = \sum_{n=0}^{M-1} a_n^2$.

We recall that X_M is given by,

$$X_M = X_0 + U_0 h \sum_{n=0}^{M-1} \left(1 - \frac{h}{\tau}\right)^{n} + \left(\sigma_U^* h \sum_{n=0}^{M-1} \left(1 - \frac{h}{\tau}\right)^{n}\right) \Delta W_0 + \tag{A.24}$$

$$+ h\sqrt{\frac{2\sigma_U^2 h}{\tau}} \sum_{n=0}^{M-1} \sum_{k=1}^{n} \left(1 - \frac{h}{\tau}\right)^{n-k} \Delta W_k,$$

and consider here, however, the special case $\sigma_U^* = \sigma_U$. The derivation for $\sigma_U^* \neq \sigma_U$ is similar, but for simplicity is not treated in this thesis. With the above form for X_M and using the fact that, for any random variable Y and constant a we have $\mathbb{V}[aY] = a^2\mathbb{V}[Y]$, we have,

$$\mathbb{V}[X_M] \quad = h^2\sigma_U^2 \left(\sum_{n=0}^{M-1} \left(1 - \frac{h}{\tau}\right)^n \right)^2 \mathbb{V}[\Delta W_0] + 2\sigma_U^2\tau h \sum_{k=1}^{M-1} \left[1 - \left(1 - \frac{h}{\tau}\right)^k\right]^2 \mathbb{V}[\Delta \tilde{W}_k].$$

Recalling that $\mathbb{V}[\Delta W_k] = 1 \ \forall k$, we have,

$$\mathbb{V}[X_M] \quad = h^2\sigma_U^2 \left(\sum_{n=0}^{M-1} \left(1 - \frac{h}{\tau}\right)^n \right)^2 + 2\sigma_U^2\tau h \sum_{k=1}^{M-1} \left[1 - \left(1 - \frac{h}{\tau}\right)^k\right]^2.$$

Now, using the fact that, $\sum_{n=0}^{M-1} y^n = \dfrac{1 - y^M}{1 - y}$, and expanding the square in the second term, we have,

$$\mathbb{V}[X_M] \quad = h^2\sigma_U^2 \left(\frac{1 - (1 - \frac{h}{\tau})^M}{1 - (1 - \frac{h}{\tau})} \right)^2 + 2\sigma_U^2\tau h \left((M - 1) - 2\sum_{k=1}^{M-1} \left(\frac{h}{\tau}\right)^k + \sum_{k=1}^{M-1} \left(\frac{h}{\tau}\right)^{2k} \right)$$

$$= h^2\sigma_U^2 \frac{\tau^2}{h^2} \left(1 - \left(1 - \frac{h}{\tau}\right)^M \right)^2 +$$

$$+ 2\sigma_U^2\tau h \left((M - 1) - 2\sum_{k=0}^{M-1} \left(\frac{h}{\tau}\right)^k - 2 + \sum_{k=0}^{M-1} \left(\frac{h}{\tau}\right)^{2k} + 1 \right),$$

where the second line follows by adding in a zero-th term to the sum and subtracting it again outside of the sum.

Again we use, $\sum_{n=0}^{M-1} y^n = \dfrac{1 - y^M}{1 - y}$, in the second term, to give,

$$\mathbb{V}[X_M] \quad = \sigma_U^2\tau^2 \left(1 - \left(1 - \frac{h}{\tau}\right)^M \right)^2 +$$

$$+ 2\sigma_U^2\tau h \left(M - 2\frac{\tau}{h} \left[1 - \left(1 - \frac{h}{\tau}\right)^M\right] + \frac{1 - (1 - \frac{h}{\tau})^{2M}}{1 - (1 - \frac{h}{\tau})^2} \right).$$

Expanding the denominator in, $\dfrac{1 - (1 - \frac{h}{\tau})^{2M}}{1 - (1 - \frac{h}{\tau})^2}$, gives,

$$\frac{1 - (1 - \frac{h}{\tau})^{2M}}{1 - (1 - \frac{h}{\tau})^2} = \frac{1 - (1 - \frac{h}{\tau})^{2M}}{1 - 1 + 2\frac{h}{\tau} - \frac{h}{\tau}^2} = \frac{1 - (1 - \frac{h}{\tau})^{2M}}{2\frac{h}{\tau} - \frac{h}{\tau}^2}$$

$$= \frac{\tau}{2h} \cdot \frac{1 - (1 - \frac{h}{\tau})^{2M}}{1 - \frac{h}{2\tau}} \approx \frac{\tau}{2h}(1 + \frac{h}{2\tau})(1 - (1 - \frac{h}{\tau})^{2M}),$$

where the final approximation follows from the fact that $\dfrac{1}{1 - y} = \displaystyle\sum_{n=0}^{\infty} y^n$ which implies that $\dfrac{1}{1 - \frac{h}{2\tau}} = 1 + \dfrac{h}{2\tau} + \dots$.

Putting this back into our derivation for $\mathbb{V}[X_M]$, and expanding the square of the first term, we now have,

$$\mathbb{V}[X_M] = \sigma_U^2 \tau^2 \left(1 - \left(1 - \frac{h}{\tau}\right)^M\right)^2 +$$

$$+ 2\sigma_U^2 \tau h \left(M - 2\frac{\tau}{h}\left[1 - \left(1 - \frac{h}{\tau}\right)^M\right] + \frac{1 - (1 - \frac{h}{\tau})^{2M}}{1 - (1 - \frac{h}{\tau})^2}\right)$$

$$= 2\sigma_U^2 \tau^2 \left(\frac{T}{\tau} - 2\left[1 - \left(1 - \frac{h}{\tau}\right)^M\right] + \frac{1}{2}\left(1 + \frac{h}{2\tau}\right)\left(1 - \left(1 - \frac{h}{\tau}\right)^{2M}\right)\right.$$

$$\left. + \frac{1}{2} - \left(1 - \frac{h}{\tau}\right)^M + \frac{1}{2}\left(1 - \frac{h}{\tau}\right)^{2M}\right)$$

which after some simplification becomes,

$$\mathbb{V}[X_M] = 2\sigma_U^2 \tau^2 \left(\frac{T}{\tau} - 1 + \left(1 - \frac{h}{\tau}\right)^M - \frac{h}{4\tau}\left(1 - \frac{h}{\tau}\right)^{2M} + \frac{h}{4\tau}\right)$$

Recalling from equation (A.23) that we can write

$$\left(1 - \frac{h}{\tau}\right)^M = \exp\left[\frac{-T}{\tau}\right]\left(1 - \frac{T}{2\tau} \cdot \frac{h}{\tau} + \dots\right)$$

and also

$$\left(1 - \frac{h}{\tau}\right)^{2M} = \exp\left[\frac{-2T}{\tau}\right]\left(1 - \frac{2T}{2\tau} \cdot \frac{h}{\tau} + \dots\right).$$

finally we have,

$$\mathbb{V}[X_M] \;=\; 2\sigma_U^2\tau^2\left(\frac{T}{\tau} - 1 + \exp\left[\frac{-T}{\tau}\right]\left(1 - \frac{T}{2\tau}\cdot\frac{h}{\tau} + \ldots\right) - \right.$$

$$\left. - \frac{1}{4}\cdot\frac{h}{\tau}\exp\left[\frac{-2T}{\tau}\right]\left(1 - \frac{2T}{2\tau}\cdot\frac{h}{\tau} + \ldots\right) + \frac{1}{4}\cdot\frac{h}{\tau}\right),$$

which we rearrange to give,

$$\mathbb{V}[X_M] \;=\; 2\sigma_U^2\tau^2\left(\frac{T}{\tau} - 1 + \exp\left[\frac{-T}{\tau}\right]\right) +$$

$$+ \sigma_U^2\tau^2\left(\frac{1}{2} - \frac{T}{\tau}\exp\left[\frac{-T}{\tau}\right] - \frac{1}{2}\exp\left[\frac{-2T}{\tau}\right]\right)\frac{h}{\tau} + \ldots$$

$$= \sigma_0^2 + \delta\sigma^2\cdot\frac{h}{\tau},$$

so we have

$$\sigma_0^2 = 2\sigma_U^2\tau^2\left(\frac{T}{\tau} - 1 + \exp\left[\frac{-T}{\tau}\right]\right)$$

and

$$\delta\sigma^2 = \sigma_U^2\tau^2\left(\frac{1}{2} - \frac{T}{\tau}\exp\left[\frac{-T}{\tau}\right] - \frac{1}{2}\exp\left[\frac{-2T}{\tau}\right]\right)$$

Thus we have a formula for the expected value for X_M, $\mathbb{E}[X_M] = \mu_h := \mu_0 + \frac{h}{\tau}\cdot\delta\mu$ and the variance, $\mathbb{V}[X_M] = \sigma_0^2 + \delta\sigma^2\cdot\frac{h}{\tau}$ accurate to $\mathcal{O}\left(\frac{h}{\tau}\right)$.

A.6   Probability Density Function Calculations

For a normally distributed random variable e.g. $Y \sim N(\mu, \sigma^2)$ the probability distribution function is given by,

$$P(Y) = \frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(-\frac{(Y - \mu)^2}{2\sigma^2}\right).$$

Using the mean, μ and variance, σ for X_M as calculated above, our probability density function, up to corrections of $\mathcal{O}\left(\left(\frac{h}{\tau}\right)^2\right)$, is thus,

$$P(Y) = \frac{1}{\sqrt{2\pi(\sigma_0^2 + \delta\sigma^2 \cdot \frac{h}{\tau})}} \exp\left(-\frac{(X_M - \mu_0 - \frac{h}{\tau}.\delta\mu)^2}{2(\sigma_0^2 + \delta\sigma^2 \cdot \frac{h}{\tau})}\right).$$

We can write the fraction, $\dfrac{1}{\sqrt{2\pi(\sigma_0^2 + \delta\sigma^2 \cdot \frac{h}{\tau})}}$, as

$$\frac{1}{\sqrt{2\pi\sigma_0^2(1 + \frac{\delta\sigma^2}{\sigma_0^2} \cdot \frac{h}{\tau})}} = \frac{1}{\sqrt{2\pi\sigma_0^2}}(1 + \frac{\delta\sigma^2}{\sigma_0^2} \cdot \frac{h}{\tau})^{-1/2} = \frac{1}{\sqrt{2\pi\sigma_0^2}}(1 - \frac{\delta\sigma^2}{2\sigma_0^2} \cdot \frac{h}{\tau} + \dots)$$

by Taylor expanding the $(1 + \dfrac{\delta\sigma^2}{\sigma_0^2} \cdot \dfrac{h}{\tau})^{-1/2}$ term.

Then expanding the numerator in the exponential term gives,

$$(X_M - \mu_0 - \frac{h}{\tau}.\delta\mu)^2 = (X_M - \mu_0)^2 - 2(X_M - \mu_0)\delta\mu.\frac{h}{\tau} + \mathcal{O}\left(\left(\frac{h}{\tau}\right)^2\right) \qquad \text{(A.25)}$$

Similarly by Taylor expansion, we can write the denominator in the exponential term as,

$$\frac{1}{2(\sigma_0^2 + \delta\sigma^2 \cdot \frac{h}{\tau})} = \frac{1}{2\sigma_0^2(1 + \frac{\delta\sigma^2}{\sigma_0^2} \cdot \frac{h}{\tau})} = \frac{1}{2\sigma_0^2}(1 - \frac{\delta\sigma^2}{\sigma_0^2} \cdot \frac{h}{\tau} + \dots). \qquad \text{(A.26)}$$

So combining the results in (A.25) and (A.26) the exponential term in the pdf becomes,

$$\exp\left(-\frac{(X_M - \mu_0 - \frac{h}{\tau}.\delta\mu)^2}{2(\sigma_0^2 + \delta\sigma^2 \cdot \frac{h}{\tau})}\right) = \exp\left(-\frac{(X_M - \mu_0)^2}{2\sigma_0^2} + \frac{2(X_M - \mu_0)}{2\sigma_0^2}\delta\mu.\frac{h}{\tau} + \right.$$

$$\left. + \frac{(X_M - \mu_0)^2}{2\sigma_0^2} \cdot \frac{\delta\sigma^2}{\sigma_0^2} \cdot \frac{h}{\tau} + \dots\right)$$

which, by using the elementary property of the exponential function, $\exp(z + y) = \exp(z)\exp(y)$, and the fact that $\exp(z) = \displaystyle\sum_{n=0}^{\infty} \frac{z^n}{n!}$, allows us to write this as,

$$\exp\left(-\frac{(X_M - \mu_0)^2}{2\sigma_0^2}\right)\left(1 - \left(\frac{2(X_M - \mu_0)}{2\sigma_0^2}\delta\mu + \frac{(X_M - \mu_0)^2}{2\sigma_0^2} \cdot \frac{\delta\sigma^2}{\sigma_0^2}\right)\frac{h}{\tau} + \dots\right)$$

Thus we can write the pdf of X_M as,

$$P(X_M) = \frac{1}{\sqrt{2\pi\sigma_0^2}}(1 - \frac{\delta\sigma^2}{2\sigma_0^2}\cdot\frac{h}{\tau} + \ldots)\exp\left(-\frac{(X_M - \mu_0)^2}{2\sigma_0^2}\right)\times$$

$$\times\left(1 - \left(\frac{2(X_M - \mu_0)}{2\sigma_0^2}\delta\mu + \frac{(X_M - \mu_0)^2}{2\sigma_0^2}\cdot\frac{\delta\sigma^2}{\sigma_0^2}\right)\frac{h}{\tau} + \ldots\right)$$

$$= \frac{1}{\sqrt{2\pi\sigma_0^2}}\exp\left(-\frac{(X_M - \mu_0)^2}{2\sigma_0^2}\right)\times$$

$$\times\left(1 - \underbrace{\left(\frac{2(X_M - \mu_0)}{2\sigma_0^2}\delta\mu + \frac{(X_M - \mu_0)^2}{2\sigma_0^2}\cdot\frac{\delta\sigma^2}{\sigma_0^2}\right)}_{=:\ q(X_M)}\frac{h}{\tau} + \ldots\right)$$

$$\text{(A.27)}$$

where $q(X_M)$ is a polynomial in X_M, defined by $q(X_M) = q_0 + q_1 X_M + q_2 X_M^2$, with

$$q_0 = -\frac{2\mu_0\delta\mu}{2\sigma_0^2} - \frac{\mu_0^2}{2\sigma_0^2}\cdot\frac{\delta\sigma^2}{\sigma_0^2} = -\frac{1}{\sigma_0^2}\left(-\mu_0\delta\mu + \mu_0^2\frac{\delta\sigma^2}{2\sigma_0^2}\right)$$

$$q_1 = -\frac{\delta\mu}{\sigma_0^2} + \frac{\mu_0\delta\sigma^2}{\sigma_0^4} = -\frac{1}{\sigma_0^2}\left(\delta\mu - \mu_0\frac{\delta\sigma^2}{\sigma_0^2}\right)$$

$$q_2 = -\frac{\delta\sigma^2}{2\sigma_0^4}$$

Thus the pdf can be written more concisely as,

$$P(X_M) = \frac{1}{\sqrt{2\pi\sigma_0^2}}\exp\left(-\frac{(X_M - \mu_0)^2}{2\sigma_0^2}\right)\cdot\left(1 + (q_0 + q_1 X_M + q_2 X_M^2)\frac{h}{\tau}\right) + \mathcal{O}\left(\left(\frac{h}{\tau}\right)^2\right)$$

The expected value of a random variable, Y, is defined as the integral over the probability space, Ω, i.e.

$$\mathbb{E}[Y] := \int_\Omega P(Y)Y\,\mathrm{d}Y$$

Similarly for some functional, f, of a random variable, Y, the expected value is given by,

$$\mathbb{E}[f(Y)] := \int_\Omega P(Y)f(Y)\,\mathrm{d}Y$$

Thus in order to compute the expected value of some functional of X_M, we need to be able to evaluate the integral of the pdf, $P(X_M)$. To simplify this we have the following, so called *master integrals*,

$$\int_a^b N(\mu_0, \sigma_0^2; X_M)\,\mathrm{d}X_M \;=\; \sqrt{\frac{\pi}{2}}\sigma_0\left(\mathrm{erf}\Big(\frac{\mu_0-a}{\sqrt{2}\sigma_0}\Big)-\mathrm{erf}\Big(\frac{\mu_0-b}{\sqrt{2}\sigma_0}\Big)\right)$$

$$\equiv I_0(\mu_0,\sigma_0;a,b)$$

(A.28)

$$\int_a^b N(\mu_0, \sigma_0^2; X_M)X_M\,\mathrm{d}X_M \;=\; \sigma_0^2\left(\exp\Big[\frac{(a-\mu_0)^2}{2\sigma_0}\Big]-\exp\Big[\frac{(b-\mu_0)^2}{2\sigma_0}\Big]\right)+$$

$$+\sqrt{\frac{\pi}{2}}\mu_0\sigma_0\left(\mathrm{erf}\Big(\frac{\mu_0-b}{\sqrt{2}\sigma_0}\Big)-\mathrm{erf}\Big(\frac{\mu_0-a}{\sqrt{2}\sigma_0}\Big)\right)$$

$$\equiv I_1(\mu_0,\sigma_0;a,b)$$

(A.29)

$$\int_a^b N(\mu_0, \sigma_0^2; X_M)X_M^2\,\mathrm{d}X_M \;=\; \frac{\sigma_0}{2}\left[\sqrt{2\pi}(\mu_0^2+\sigma_0^2)\left(\mathrm{erf}\Big(\frac{\mu_0-a}{\sqrt{2}\sigma_0}\Big)-\mathrm{erf}\Big(\frac{\mu_0-b}{\sqrt{2}\sigma_0}\Big)\right)-\right.$$

$$-2\sigma_0\left((b+\mu_0)\exp\Big[-\frac{(b-\mu_0)^2}{2\sigma_0}\Big]-\right.$$

$$\left.\left.-(a+\mu_0)\exp\Big[-\frac{(a-\mu_0)^2}{2\sigma_0}\Big]\right)\right]$$

$$\equiv I_2(\mu_0,\sigma_0;a,b)$$

(A.30)

Using these master integrals to compute the expected value of our functional of the random variable, X_M, gives us the following,

$$\mathbb{E}[\chi_{a.b}(X_M)] = I_0(\mu_0,\sigma_0;a,b) + \left(\sum_{k=0}^{2} q_k I_k(\mu_0,\sigma_0;a,b)\right).\frac{h}{\tau} \tag{A.31}$$

where $\chi_{a.b}(X_M)$ is in particular the indicator function on the interval $[a,b]$, i.e.

$$\chi_{a,b}(X_M) = \begin{cases} 1 & \text{if } X_M \in [a,b] \\ 0 & \text{otherwise} \end{cases}$$

So we now have a formula for the expected value of $\chi_{a,b}(X_M)$, from this we are able to compute the variance of our functional also. First we write the expected value as,

$$\mathbb{E}[\chi_{a,b}(X_M)] = \int P(X_M)\chi_{a,b}(X_M)\,\mathrm{d}X_M =: A \tag{A.32}$$

By definition, the variance of $\chi_{a,b}(X_M)$ is,

$$\mathbb{V}[\chi_{a,b}] = \mathbb{E}\left[\left(\chi_{a,b} - \mathbb{E}[\chi_{a,b}]\right)^2\right]$$

which by the definition of the expected value can be written as,

$$\mathbb{V}[\chi_{a,b}(X_M)] = \int P(X_M)\left(\chi_{a,b}(X_M) - A\right)^2 \mathrm{d}X_M$$

expanding the square gives,

$$\mathbb{V}[\chi_{a,b}(X_M)] = \int P(X_M)\left(\chi_{a,b}^2(X_M) - 2A\chi_{a,b}(X_M) + A^2\right)\mathrm{d}X_M$$

$$= \int P(X_M)\chi_{a,b}^2(X_M)\,\mathrm{d}X_M - 2A\underbrace{\int P(X_M)\chi_{a,b}(X_M)\,\mathrm{d}X_M}_{=A} +$$

$$+ A^2 \underbrace{\int P(X_M)\,\mathrm{d}X_M}_{=1}$$

where we have $\underbrace{\int P(X_M)\,\mathrm{d}X_M}_{=1}$, by the definition of the probability density function.

Since the the indicator function takes values only of either 0 or 1, the square of the indicator function is the indicator function itself and thus the variance of the functional of the solution is given by,

$$\mathbb{V}[\chi_{a,b}(X_M)] = A - 2A^2 + A^2 = A(1-A) = \mathbb{E}[\chi_{a,b}]\left(1 - \mathbb{E}[\chi_{a,b}]\right).$$