Coarse Graining Monte Carlo Methods for Wireless Channels and Stochastic Differential Equations

HÅKON HOEL
Abstract

This thesis consists of two papers considering different aspects of stochastic process modelling and the minimisation of computational cost.

In the first paper, we analyse statistical signal properties and develop a Gaussian process model for scenarios with a moving receiver in a scattering environment, as in Clarke’s model, with the generalisation that noise is introduced through scatterers randomly flipping on and off as a function of time. The Gaussian process model is developed by extracting mean and covariance properties from the Multipath Fading Channel model (MFC) through coarse graining. That is, we verify that under certain assumptions, signal realisations of the MFC model converge to a Gaussian process and thereafter compute the Gaussian process’ covariance matrix, which is needed to construct Gaussian process signal realisations. The obtained Gaussian process model is under certain assumptions less computationally costly, containing more channel information and having very similar signal properties to its corresponding MFC model. We also study the problem of fitting our model’s flip rate and scatterer density to measured signal data.

The second paper generalises a multilevel Forward Euler Monte Carlo method introduced by Giles [10] for the approximation of expected values depending on the solution to an Itô stochastic differential equation. Giles work [10] proposed and analysed a Forward Euler Multilevel Monte Carlo method based on realisations on a hierarchy of uniform time discretisations and a coarse graining based control variates idea to reduce the computational effort required by a standard single level Forward Euler Monte Carlo method. This work introduces an adaptive hierarchy of non uniform time discretisations generated by adaptive algorithms developed by Moon et al. [18, 17]. These adaptive algorithms apply either deterministic time steps or stochastic time steps and are based on a posteriori error expansions first developed by Szepessy et al. [19]. Under sufficient regularity conditions, our numerical results, which include one case with singular drift and one with stopped diffusion, exhibit savings in the computational cost to achieve an accuracy of $O(T_{tol})$, from $O(T_{tol}^{-3})$ to $O\left((\log(T_{tol})/T_{tol})^2\right)$. We also include an analysis of a simplified version of the adaptive algorithm for which we prove similar accuracy and computational cost results.
Preface

This thesis consists of an introduction and two papers

**Paper I** H. Hoel, *Gaussian Coarse Graining of a Master Equation Generalisation of Clarke’s Model*, to be submitted.

The author was the main contributor to this paper, but assistance was given from supervisor Anders Szepessy and industrial contact Henrik Nyberg.


The author contributed to the introduction, description of algorithms and mathematical analysis.
Acknowledgements

First and foremost, I would like to thank my supervisor Anders Szepessy for creating a stimulating and challenging working environment. Your applied approach to mathematics impresses me and I find it a privilege to get to know some of your thoughts on mathematics. For the same reasons I would also like to thank Raúl Tempone. It has been a joy to work together with you on Paper II and I had a terrific time visiting you at Florida State University. The weekly meetings in our research group has (at times) also included Christian Bayer, Jesper Carlsson, Jonas Kiessling and Erik Von Schwerin. They have all provided helpful comments, presented interesting work and made me feel included. Henrik Nyberg, my industrial contact at Ericsson Research, has given much appreciated support on problems dealing with wireless communication and contributed to Paper I. Henrik Asplund from Ericsson Research provided very useful signal measurements for Paper I.

I thank my fellow colleagues at Nada for creating a friendly and tolerant atmosphere. In particular, Olof Runborg who proof read this thesis and contributed with many insightful suggestions for improvement, and Jelena and Sara: your company and positivity make work a little easier.

At the end, I would like to thank the closest versions of myself on Planet Earth; my family.

This work is funded by Centre for Industrial and Applied Mathematics (CIAM).
## Contents

1 Introduction 1

2 Statistical Channel Modelling 3

2.1 The Multipath Fading Channel 3

2.2 Clarke’s Model 5

2.3 MFC Model with Flipping Scatterers 6

2.4 From MFC Model to Gaussian Process 7

2.4.1 Gaussian Processes and Numerical Realisations 8

2.5 Results 9

3 Adaptive Multilevel Monte Carlo 11

3.0.1 The Multilevel Monte Carlo Method 12

3.1 A Single Level Adaptive Method for SDEs 13

3.1.1 The Time Discretisation Error 14

3.1.2 The statistical error 14

3.2 Multilevel Error Control 15

3.2.1 Control Variates 15

3.3 Results 16
Chapter 1

Introduction

A stochastic process is a non-deterministic process whose future evolution is described by a probability distribution, as opposed to the deterministic evolution of an ordinary differential equation. Complex phenomena which from observation seem non-deterministic, such as stock market evolution and wireless signal transmission, are frequently modelled by stochastic processes, cf. [4, 16] for option price models and [8, 9] for signal transmission models. This thesis consider different aspects of minimising the computational cost when modelling stochastic process. The first paper look at the applied problem of wireless channels modelling by Gaussian processes—a type of stochastic processes whose evolution is described by the normal distribution. The second paper introduces an algorithm for improving the computational cost of estimating expected values of functionals of Stochastic Differential Equations (SDE).

Wireless channel models try to describe the received version of signal transmitted wirelessly from a transmitter to a receiver. For industries developing wireless transmission equipment, such models are used to estimate the performance of equipment and software by simulations instead of more costly real-world tests. But for such models to be of interest, they must be accurate and computationally efficient with respect to running time.

One of the most popular models of today, the Multipath Fading Channel models (MFC), are based on approximating the signal from superpositioning a finite number of contributing wave paths. From a computational point of view, the superpositioning is costly, and in the first paper of this thesis, we study when the Central Limit Theorem can be invoked to replace the superpositioning of wave paths with a Gaussian process. For cases where replacement is possible, we develop a Gaussian process channel model which under certain assumptions is less computationally costly and gives more channel information than MFC models do. We also look into the problem of including noise in the channel model and offer a new interpretation of noise with scatterers flipping on and off to contribute to the total signal.

In the second paper, we develop an adaptive multilevel Forward Euler Monte Carlo algorithm which for any sufficiently well behaved function $g : \mathbb{R}^d \rightarrow \mathbb{R}$
approximates the expected value $E[g(X(T))]$ where $X(T)$ is the solution of an Itô SDE. A typical example of such an expected value is to compute option prices in mathematical finance.

The multilevel Forward Euler Monte Carlo method on uniform time grids, first introduced by Giles in [10], showed the improvement of computational cost of computing an approximation of $E[g(X(T))]$ with accuracy $O(Tol)$ from $O(Tol^{-3})$ with a single level method to $O((\log(Tol)/Tol)^2)$ with the multilevel method. This work is an extension of his method from uniform to adaptive time grids. It has the same improvement in computational cost and is applicable to a larger set of SDE problems.

The rest of this thesis is organised as follows. In Chapter 2, background on wireless channel modelling is given and results from Paper I are summarised. Chapter 3 gives a short introduction to the Forward Euler method for SDEs, the Monte Carlo method, variance reduction and results from paper II are summarised. At the end, Paper I and II are included.
Chapter 2

Statistical Channel Modelling

The first paper studies the modelling of a wireless channel with a transmitter, a moving receiver and a scattering boundary. Wireless channels operate through electromagnetic radiation from transmitter to receiver, and what we seek is the electromagnetic field at the receiver as a function of time. In principle, a numerical solution of Maxwell's equations with well resolved boundary would yield the electrical field at the receiver and everywhere around it. But, for standard modelling scenarios where the communication carrier frequency is of order $10^9$Hz and the wavelength consequently is of order centimetres, localisation of the scattering boundary would have to be made to the order of centimetre accuracy to obtain an acceptable electrical field solution. Since this typically implies the need for resolving milliards of boundary points, determining the electrical field through solving Maxwell's equations is considered too costly both from a boundary measurement and computational perspective.

2.1 The Multipath Fading Channel

A channel model studied by Clarke [5] is to approximate the output signal by a superposition of a large number $M$ of incoming signal wave paths, as illustrated by Figure 2.1. With $X(t)$ denoting the baseband input signal, the time invariant MFC model has the baseband output representation

$$Z(t) = \sum_{j=1}^{M} a(\alpha_j) e^{-i2\pi f_c \tau(\alpha_j,t)/c} X(t - \tau(\alpha_j,t)).$$

(2.1.1)

Here, $f_c$ denotes the carrier frequency, $\alpha_j$ the horizontal angle of arrival for the $j^{th}$ wave path, and $a(\alpha_j)$ and $\tau(\alpha_j,t)$ its amplitude and time delay function, respectively. The terminology baseband input and output signal derive from frequency modulated signal transmission where the processed signal you wish
to transmit, called the baseband input signal $X(t)$, is frequency modulated to a passband input signal. The passband input signal is transmitted to the receiver, and the received version of the signal, called the passband output signal, is demodulated to recover a signal with a similar frequency band as the baseband input signal. The demodulated signal is called the baseband output signal, here denoted $Z(t)$.

![Figure 2.1: Illustration of typical scattering scenario considered for the MFC model. Lines between the transmitter, scatterers and the receiver represent different radio wave paths.](https://example.com/figure2.1.png)

Introducing the channel response $H(s, t)$, one can represent the output signal (2.1.1) as the convolution

$$Z(t) = \int_{\mathbb{R}} H(s, t)X(t - s) \, ds$$

where

$$H(s, t) := \sum_{j=1}^{M} a(\alpha_j)e^{-j2\pi f_c\tau(\alpha_j, t)}\delta(s - \tau(\alpha_j, t)),$$  \hspace{1cm} (2.1.2)

and $\delta$ denotes the Dirac delta function. Since the channel response holds the full information of the channel, it is what we wish model. However, signal processing is performed on discrete signals and this changes the channel response of interest slightly. In the discrete setting, all wave paths arriving at the receiver within a sample period $\Delta t$ are averaged to become the received value for that time sample. The Nyquist-Shannon Sampling Theorem describes how small the sampling time has to be to fully resolve a signal.
2.2. CLARKE’S MODEL

Theorem 2.1.1 (Nyquist-Shannon Sampling Theorem [23]) If a function \( X(t) \) contains no frequencies higher than \( B \) Hertz, it is completely determined by giving its ordinates at a series of points spaced \( 1/(2B) \) seconds apart.

If we have a signal with frequency baseband bounded by \( B \), it is sufficient to sample it at rate \( \Delta t = 1/(2B) \) to completely determine it. The discrete baseband input signal might thus be represented by values \( \{X(t_m)\}_{m \in \mathbb{Z}} \), where \( t_m = m\Delta t \). Consequently, the discrete output baseband signal becomes

\[
Z(t_m) = \sum_{\ell \in \mathbb{Z}} X(t_m-\ell)H(t_\ell,t_m),
\]

where the discrete channel response is defined by

\[
H(t_\ell,t_m) = \Delta t^{-1} \sum_{j=1}^{M} a(\alpha_j) e^{-i2\pi f_c \tau(\alpha_j,t_m)} 1_{[-\Delta t/2,\Delta t/2]}(t_\ell - \tau(\alpha_j,t_m)), \quad \ell, m \in \mathbb{Z}.
\]

Restricting ourselves to flat fading scenarios, that is, when the time delay

\[
T_d := \max_{1 \leq i,j \leq M} |\tau(\alpha_i,t) - \tau(\alpha_j,t)|
\]

is very small compared to the needed time sampling \( 1/(2B) \), all the wave paths emitted at a given time from the transmitter arrives within the same sampling interval at the receiver. This implies that for each \( t_m \) the discrete channel response \( H(t_\ell,t_m) \) is non zero for only a single \( t_\ell \).

By choosing the input signal \( X(t) = 1 \), one models the baseband output signal

\[
Z(t_m) = \sum_{j=1}^{M} a(\alpha_j) e^{-i2\pi f_c \tau(\alpha_j,t_m)}.
\]

This signal is the one we are going to model since it is closely related to the discrete channel response.

Remark 2.1.2 The ideas presented in this section is inspired by Chapter 2 of [22].

2.2 Clarke’s Model

A famous MFC model which was motivational for our work is Clarke’s model [5]. It considers the superposition of \( M \) wave paths with equal sized amplitude of value \( 1/\sqrt{M} \),

\[
Z_{t,M} = \frac{1}{\sqrt{M}} \sum_{m=1}^{M} e^{-i(2\pi f_v \cos(\alpha_m) t/c + \Theta_m(s))}.
\]

Here, apart from the variables already introduced in Section 2.1, \( c \) is the speed of light, \( \{\Theta_m\}_{m=1}^{M} \) are i.i.d. initial phase shifts uniformly distributed in \([0,2\pi)\)
and the scatterer angle of arrivals \( \{\alpha_m\}_{m=1}^M \) are distributed according to the scatterer angle density \( p(\alpha) \) which is independent from the initial phase shift distribution. The delay function is on the form \( \tau(\alpha, t) = -f_c v \cos(\alpha)t/c \) and the factor \( \partial_t \tau(\alpha, t) = -f_c v \cos(\alpha)/c \) is the Doppler shift under the assumption that the receiver moves in the direction \((1, 0)\). The Doppler shift describes the change of frequency of a wave for a receiver moving relative to the transmitter of the wave (the tone of the siren of a passing ambulance is a classical illustration of this phenomenon).

Among the functions used to analyse channel properties is the autocorrelation function and the Power Spectral Density (PSD). The autocorrelation is defined by \( A_M(t) := E[Z_t, M \mid Z_0, M] \) and describes how \( Z_t, M \) correlates with \( Z_0, M \).

The PSD describes the received signal’s frequency spectrum and the Wiener-Khintchine Theorem links the autocorrelation of a Wide Sense Stationary (WSS) signal to its PSD.

**Definition 2.2.1 (Wide Sense Stationary Random Process)** A random process \( Z_t \) is WSS if there is a function \( g \) such that

\[
E[Z_{t, M} \bar{Z}_{t, M}] = g(|\bar{t} - t|) \quad \text{for any } \bar{t}, t \in \mathbb{R}.
\]

**Theorem 2.2.2 (Wiener-Khintchine Theorem [6, p. 49])** The power spectral density and autocorrelation of a wide sense stationary process are Fourier transform pairs.

The signals of Clarke’s model and the MFC model we propose are both WSS.

Considering the scenario with scatterer angle density \( p(\alpha) = (2\pi)^{-1} \), Clarke noted that for his model, the autocorrelation function \( A_M(t) \) converges to the zeroth-order Bessel function of the first kind, \( J_0(2\pi f_c v t/c) \), as \( M \to \infty \), and that its PSD then is on the form

\[
S(f) = \begin{cases} 
\frac{c}{\pi \sqrt{({vf}_c)^2 - (cf)^2}} & |f| < {vf}_c/c \\
0 & |f| \geq {vf}_c/c.
\end{cases} \quad (2.2.2)
\]

For plots of these results, see Figure 2.2.

### 2.3 MFC Model with Flipping Scatterers

Due to local shadowing by moving cars, bikers and pedestrians, leaves blowing in the wind, weather conditions etc. scatterers can flip from being active to passive and vice versa. Wishing to include local shadowing to Clarke’s model, the amplitude function is defined as a random process \( a(\alpha, t) \) which flips on when it changes value from 0 to \( a^+(\alpha) \geq 0 \) and off when it changes values oppositely. It is here assumed that the mapping \( a^+ : \Omega \to \mathbb{R}_+ \) is smooth; it might for example be constant or depend on the distance from scatterer to the receiver. The flip process is taken to be Poisson distributed with flip rate constant \( C \):

\[
P(a(\alpha) \text{ flips } k \text{ times on time step } \Delta t) = \frac{(C \Delta t)^k \exp(-C \Delta t)}{k!}.
\]
2.4. FROM MFC MODEL TO GAUSSIAN PROCESS

Figure 2.2: **Left plot:** The autocorrelation function for Clarke’s model with azimuth density $p(\alpha) = (2\pi)^{-1}$, $v = 5 m/s$ and $f_c = 1.8775 GHz$. **Right plot:** The power spectral density of Clarke’s model, often called Jakes’ spectrum with the same model conditions as for the left plot.

where flips are independent from the scatterers’ random initial phase shifts.

With the above defined amplitude function and the set of arrival angles $\Omega_M = \{\alpha_j\}_{j=1}^M$ distributed according to a scatterer density $p(\alpha)$, we propose the modification of Clarke’s model

$$Z_{t,M} = \frac{1}{\sqrt{M}} \sum_{\alpha \in \Omega_M} a(\alpha, t)e^{-i(2\pi f_c v \cos(\alpha) t/c + \Theta_\alpha)}.$$  

Realisations of (2.3.1) can be generated by sampling scatterer angles $\Omega_M$ and i.i.d. initial phase shifts $\{\Theta_\alpha\}_{\alpha \in \Omega_M}$, simulating the evolution of the wave paths with amplitudes flipping, and, at the end, superpositioning them. The left plot of Figure 2.3 illustrates the difference between an MFC signal realisation envelope generated with $C = 0$ and one with positive flip rate. The positive flip rate gives the signal envelope more small scale temporal noise and less smoothness than what is found in the non-flipping signal envelope. The right plot of Figure 2.3 is a measured signal envelope from Ericsson Labs which shows that the measured signal has a small scale noise contribution similar to the MFC signal realisation with positive flip rate.

### 2.4 From MFC Model to Gaussian Process

One drawback with generating signal realisations from the MFC model (2.3.1) is that it is computationally costly. To reduce the computational cost, we studied the possibility of approximating $Z_{t,M}$ by a Gaussian process.
2.4.1 Gaussian Processes and Numerical Realisations

A Gaussian process is defined as follows

**Definition 2.4.1 (Gaussian Process)** A Gaussian process is a stochastic process \( \{Z_t\}_{t \in [0,T]} \), \( Z_t \in \mathbb{R}^n \), for which any finite length sample vector \( Z = (Z_{t_1}, Z_{t_2}, \ldots, Z_{t_n}) \) with \( 0 \leq t_1 \leq t_2 \leq \ldots \leq t_n \) is multivariate normal distributed.

A numerical realisation of a Gaussian process on a set of times \( \{t_j\}_{j=1}^N \) can be created by first computing the process’ covariance matrix,

\[
K_{i,j} = E[Z_{t_i}Z^*_j], \quad i, j \in \{1, 2, \ldots, N\}
\]

and setting

\[
Z_{t_i} = \sum_{j=1}^N \sqrt{K}_{i,j}X_j,
\]

where \( \sqrt{K} \) is the square root of \( K \) in the sense that it fulfills \( K = \sqrt{K}\sqrt{K}^* \) (typically the lower diagonal Cholesky factorisation), and \( X_1, X_2, \ldots, X_N \) is a set of independent standard normal distributed random variables.

**Example 2.4.2** The Wiener process \( W_t \) is a famous example of a Gaussian process which has the increment property \( W_{t_2} - W_{t_1} \sim N(0, |t_2 - t_1|) \) and thereby the covariance matrix

\[
K_{i,j} = E[W_{t_i}W^*_{t_j}] = E[W^2_{\min(t_i, t_j)}] = \min(t_i, t_j).
\]
For the Wiener process, one is particularly fortunate with the structure of the covariance matrix obtained by Cholesky factorisation:

\[ \sqrt{K} = \begin{pmatrix} \sqrt{t_1} & 0 & \ldots & 0 \\ \sqrt{t_1} & \sqrt{t_2 - t_1} & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \sqrt{t_1} & \sqrt{t_2 - t_1} & \ldots & \sqrt{t_N - t_{N-1}} \end{pmatrix} . \]

This is fortunate because it makes the computational cost of generating a Wiener process realisation \( \{W_t\}_{j=1}^N \) of \( O(N) \) compared to the cost \( O(N^2) \) normally expected for a general Gaussian process. This can be seen from its scheme (2.4.1) which becomes \( W_{t_{j+1}} = W_{t_j} + \sqrt{t_{j+1} - t_j} X_{j+1} \).

Wiener processes can furthermore be linked to Stochastic Differential Equations (SDE) by the expression

\[ dZ_t = a(Z_t, t) dt + b(Z_t, t) dW_t. \]

The cost of constructing a realisation of an SDE on a set of times \( \{Z_t\}_{j=1}^N \) is also \( O(N) \), and due to this low computational complexity, it was initially our hope to link the MFC model to an SDE.

2.5 Results

Construction of Gaussian Process

By application of Lindeberg-Feller’s extension of the Central Limit Theorem, cf. [13, Appendix A], we show that the signal \( Z_{M,t} \) of the MFC model (2.3.1) converges weakly to a complex Gaussian process \( Z_t \) as \( M \to \infty \). We also compute the process’ covariance matrix through quadrature, and include algorithms for generating realisations of the Gaussian process, cf. [13, Algorithm 2 and Algorithm 3].

Determining the Flip Rate

An algorithm is developed for determining the flip rate \( C \) through fitting theoretical PSD to signal measure data PSD by minimising the distance between respective PSDs in a given norm, cf. [13, Algorithm 3]. The flip rate is directly related to the size of the diffusion noise of the measured signal. If the measured signal’s diffusion noise is high, its PSD will have large tail, and by our algorithm, this yields a high flip rate.

Similar Signal Behaviour

Both computationally and experimentally, signal realisations derived by the MFC model and by the Gaussian process have similar autocorrelation and PSD functions.
Computational Cost Lowered

For many scenarios, the Gaussian process algorithm has lower computational cost than the MFC model’s algorithm. For the MFC algorithm, the computational cost of generating $L$ realisations with $M$ scatterers on $N$ time steps by the ray tracing algorithm is $O(LMN)$, while for the Gaussian process algorithm it is $O(LN \log(N))$ when the signal is WSS and $O(N^3 + LN^2)$ when it is not.
Chapter 3

Adaptive Multilevel Monte Carlo

The second paper considers the problem of minimising the computational cost of approximating the expected value \( E[g(X(T))] \) within a given tolerance \( Tol \). Here \( g : \mathbb{R}^n \rightarrow \mathbb{R} \) is a given function and \( X \) solves the Itô Stochastic Differential Equation (SDE)

\[
dX = a(t, X)dt + b(t, X)dW_t, \quad 0 \leq t \leq T.
\]

\( X(0) = x. \) (3.0.1)

A typical example of such an expected value is to compute option prices in mathematical finance; see [14] and [11]. Other related models based on stochastic dynamics are used for example for stochastic climate prediction and for wave propagation in random media; see [15] and [1].

The standard approach to solving this problem is to first generate numerical realisations \( \mathbf{X}(T) \) by the Forward Euler scheme

\[
\mathbf{X}(t_{n+1}) = \mathbf{X}(t_n) + a(t_n, \mathbf{X}(t_n))\Delta t + b(t_n, \mathbf{X}(t_n))(W(t_{n+1}) - W(t_n)), \quad (3.0.2)
\]

with uniform step size \( \Delta t = T/N, \) \( t_n = n\Delta t \) and the Wiener increments \( W(t_{n+1}) - W(t_n) \sim N(0, \Delta t) \). Thereafter, approximate \( E[g(X(T))] \) by the Monte Carlo Multilevel estimate

\[
\mathcal{E} = \sum_{i=1}^{M} \frac{g(\mathbf{X}(T; \omega_i))}{M}. \quad (3.0.3)
\]

For a given \( Tol > 0 \), one wants the approximation to satisfy \( |E[g(X(T))] - \mathcal{E}| \leq Tol \) with the minimum computational cost. To estimate the amount of work needed, one starts by splitting the approximation error into two controllable parts

\[
|E[g(X(T))] - \mathcal{E}| \leq |E[g(X(T)) - g(\mathbf{X}(T))]| + |E[g(\mathbf{X}(T))] - \mathcal{E}| = \mathcal{E}_T + \mathcal{E}_S,
\]
the time discretisation error $E_T$ and the statistical error $E_S$. Talay and Tubaro [21] showed that under appropriate smoothness assumptions on the drift $a$ and diffusion $b$ of (3.0.2), the time discretisation error fulfills $E_T = O(\Delta t)$ when the step size is uniform. Further, by the Central Limit Theorem, one can control the statistical error noting that $\text{Var}(E_S) = O(\frac{1}{M})$. So to ensure that $|E[g(X(T))] - \mathcal{E}| \leq \text{Tol}$ while minimising the computational cost, one has to choose $\Delta t = O(\text{Tol})$ and $M = O(\text{Tol}^{-2})$ which in total yields the computational cost $O(\text{Tol}^{-3})$.

### 3.0.1 The Multilevel Monte Carlo Method

The multilevel Monte Carlo method based on uniform time stepping was introduced by Giles in [10]. His idea was to develop a variance reduction technique for a numerical method, denoted here by $X$, that approximates the solution of the SDE (3.0.1). The key to the variance reduction in [10] is to compute approximate solutions, $X_\ell$, on hierarchies of uniform time meshes with the typical size relation

$$\Delta t_\ell = 2^{-\ell} \Delta t_0, \quad \ell = 0, 1, 2, \ldots, L,$$

After computing numerical approximations on a mesh hierarchy, the expected value $E[g(X(T))|$ is approximated by the multilevel Monte Carlo estimator

$$E_L = \sum_{i=1}^{M_0} \frac{g(X_0(T; \omega_{i,0}))}{M_0} + \sum_{\ell=1}^{L} \sum_{i=1}^{M_\ell} \frac{g(X_\ell(T; \omega_{i,\ell})) - g(X_{\ell-1}(T; \omega_{i,\ell}))}{M_\ell}. \quad (3.0.4)$$

Here, $\omega_{i,\ell}$ refers to the stochastic realisation and $M_\ell = 2^{-\ell} M_0, \quad \ell = 0, 1, \ldots, L$

to the number of samples at level $\ell$. To reduce the variance in the above estimator, the realisation pairs $X_\ell(T; \omega_{i,\ell})$ and $X_{\ell-1}(T; \omega_{i,\ell})$ of the summands $g(X_\ell(T; \omega_{i,\ell})) - g(X_{\ell-1}(T; \omega_{i,\ell}))$ for each level $\ell > 0$ are generated by the same Brownian path, $W_T(\omega_{i,\ell})$, but they are realised on different temporal grids with uniform time steps, $\Delta t_\ell$ and $\Delta t_{\ell-1}$, respectively. The efficiency of this computation relies on an a priori known order of strong convergence for the numerical method employed on each level of the hierarchy.

Assuming Tol > 0 to be the desired order of accuracy in the approximation of $E[g(X(T))|$, Giles shows that the computational cost needed to achieve

$$|E_L - E[X(T)]| = O(\text{Tol}),$$

when using the Forward Euler scheme to create the approximate realisations $X_\ell(T; \omega)$, can be reduced from $O(\text{Tol}^{-3})$ with the single level Monte Carlo method to $O((\log(\text{Tol}^{-1})/\text{Tol})^2)$ with Giles’ multilevel Monte Carlo method.

This work is an extension of Giles’ method from computing Forward Euler realisations of $X_\ell$ with the uniform time step $2^{-\ell} \Delta t_0$ to computing them on adaptive refined grids $\Delta t(\omega)$ until the accuracy

$$|E[g(X(T)) - g(X_\ell(T))]| \lesssim 2^{-L-\ell} \text{Tol}, \quad (3.0.5)$$
3.1. A SINGLE LEVEL ADAPTIVE METHOD FOR SDES

is met. It has a similar improvement in computational cost as Giles’ method and is applicable to a larger set of SDE problems than the method with uniform time grids, for example, stopped diffusion and drift singularity problems.

The next section describes the adaptive method we use and Section 3.2 has more on multilevel variance reduction.

3.1 A Single Level Adaptive Method for SDEs

For a general approximation problem involving the solution of a differential equation, Bangerth and Rannacher state in [3] that “the goal of adaptivity is the optimal use of computing resources according to either one of the following principles:

• Minimal work \( N \) for prescribed accuracy \( Tol \),

\[ N \rightarrow \min, \quad Tol \text{ given.} \]

• Maximal accuracy of prescribed work,

\[ Tol \rightarrow \min, \quad N \text{ given.} \]

Here we consider the latter of these principles and look at a single level approach to the approximation problem given at the beginning of the chapter: For a prescribed tolerance \( Tol \), choose the size of the timesteps \( \Delta t_n = t_{n+1} - t_n \), used in to create realisations by the Forward Euler method (3.0.2), and the number of samples \( M \) which minimises the computational cost of computing an approximation

\[ \mathcal{E} = \sum_{i=1}^{M} \frac{g(X(T); \omega_i)}{M} \]  \hspace{1cm} (3.1.1)

which with high probability fulfills

\[ |E[g(X(T))] - \mathcal{E}| \leq Tol. \]

In [20], Szepessy et al. suggest a method for solving this problem which we will give a short summary of here. First, split the computational error into two parts,

\[ |E[g(X(T))] - \mathcal{E}| \leq |E[g(X(T)) - g(X(T))]| + |E[g(X(T))] - \mathcal{E}| \]

\[ = \mathcal{E}_T + \mathcal{E}_S; \]

the time discretisation error \( \mathcal{E}_T \) and the statistical error \( \mathcal{E}_S \). Next, control the errors \( \mathcal{E}_T \leq Tol_T \) and \( \mathcal{E}_S \leq Tol_S \) such that \( Tol_T + Tol_S = Tol \).
3.1.1 The Time Discretisation Error

The time discretisation error can be expressed by the expansion

\[ E[g(X(T)) - g(\overline{X}(T))] \approx E[\sum_n \rho(t_n, \omega) \Delta t_n(\omega)^2] + \text{higher order terms}, \]

where the error density \( \rho(t_n, \omega) \) measures the density of the global error in \( E[g(X(t)) - g(\overline{X}(t))] \). The error density is based on weighting with the solution of the dual backward problem

\[
\phi(t_n) = \left(1 + \partial_x a(t_n, \overline{X}(t_n)) \Delta t_n + \partial_x b(t_n, \overline{X}(t_n)) \Delta W_n \right) \phi(t_{n+1}), \quad t < T
\]

\[
\phi(T) = \partial_x g(\overline{X}(T)),
\]

which is obtained from linearising the forward problem (3.0.1) around the solution, cf. [20, Theorem 2.2]. The use of dual functions is standard in optimal control theory and well known for adaptive mesh control for ordinary and partial differential equations, cf. [2, 7]. The error indicators \( \rho(t_n, \omega) \Delta t_n(\omega)^2 \) provide information for further improvement of the time mesh. In particular, the computational cost of generating one realisation is minimised by ensuring that

\[ \rho(t_n, \omega) \Delta t_n(\omega)^2 = \text{const}, \quad \forall n = 1, 2, \ldots \] (3.1.3)

Given an initial time discretisation \( \Delta t[0](t, \omega) \), the following adaptive mesh refinement algorithm is used to ensure that (3.1.3) is fulfilled

1. Set \( \ell = 0 \).
2. If (3.1.3) is fulfilled then stop; otherwise
3. Refine the mesh \( \Delta t[\ell](t, \omega) \) at all points where (3.1.3) is not fulfilled by halving; \( \Delta t[\ell + 1](t, \omega) = \Delta t[\ell](t, \omega)/2 \). Set \( \ell = \ell + 1 \) and go back to 2.

3.1.2 The Statistical Error

The statistical error is controlled by choosing \( M \) so large that with high likelihood

\[
\left| E[g(\overline{X}(T))] - \frac{1}{M} \sum_{j=1}^{M} g(\overline{X}(T; \omega_j)) \right| \leq \text{Tol}_S. \] (3.1.4)

This can be done by noting that by the Central Limit Theorem

\[
\frac{1}{\sqrt{M}} \sum_{j=1}^{M} g(\overline{X}(T; \omega_j) - E[g(\overline{X}(T))] \sim N(0, 1)
\]

and approximating the variance

\[ \sigma^2 = \text{Var} (\mathcal{E}), \]
3.2. **Multilevel Error Control**

The control of the error for the single level Monte Carlo method presented above has a natural extension to the multilevel method. Recalling the definition of $\mathcal{E}_L$ in (3.1.1) and noting that $E[\mathcal{E}_L] = E[g(\mathcal{X}(T))]$, the error in the multilevel approximation also splits into a statistical error and a time discretisation error

$$|E[g(X(T))] - \mathcal{E}_L| \leq |E[g(X(T))] - E[g(\mathcal{X}(T))]| + |E[g(\mathcal{X}(T))] - \mathcal{E}_L| = \mathcal{E}_T + \mathcal{E}_S.$$  (3.2.1)

Computing realisations $\mathcal{X}_\ell(T)$ according to (3.0.5) ensures that the time discretisation error is met. The statistical error is met by increasing $M_0$, and thereby all other sample numbers $M_\ell = M_02^{-\ell}$, until the sample standard deviation of $\mathcal{E}_L$ fulfills $\hat{\sigma} \leq Tol_S/C_C$.

### 3.2.1 Control Variates

Suppose that for a given random variable $X$, we want to approximate the expected value $E[X]$ by the Monte Carlo method, and suppose we for another random variable $Y$ know the expected value $E[Y]$. Then we see that the random variable $Z = X + E[Y] - Y$ has the same expected value as $X$, so we could just as well approximate $E[X]$ by applying the Monte Carlo method on $Z$. By the reasoning of Subsection 3.1.2, we conclude that it is more efficient to perform the Monte Carlo estimate on $Z$ than on $X$ when $Var(Z) = Var(X - Y) < Var(X)$. This is because the lower the variance, the lower the sample size need be to control the statistical error of the Monte Carlo estimate. In this setting, $Y$ is called the control variate.

Using a control variates technique to estimate $E[g(\mathcal{X}(T))]$, yields a variance reduction which reduces the computational cost. A rough illustration of the variance reduction can be seen from first considering the single level estimator

$$\sum_{i=1}^{M_L} \frac{g(\mathcal{X}_L(T; \omega L, i))}{M_L},$$

whose variance is $O(M_L^{-1})$. Next, using the following strong convergence result valid under certain smoothness assumptions

$$E[|g(\mathcal{X}_\ell(T; \omega)) - g(\mathcal{X}_{\ell-1}(T; \omega))|^2] = O(\sup_t \Delta t[\ell](t)),$$
CHAPTER 3. ADAPTIVE MULTILEVEL MONTE CARLO

the two level estimator
\[
\sum_{i=1}^{M_L} \frac{g(X_L(T;\omega_{L,i})) - g(X_{L-1}(T;\omega_{L-1,i}))}{M_L} + \sum_{i=1}^{M_{L-1}} \frac{g(X_{L-1}(T;\omega_{L-1,i}))}{M_{L-1}}
\]
(3.2.2)

has variance \(O(\sup_t \Delta t[\ell](t)/M_L + 1/M_{L-1})\). Since \(M_{L-1} = 2M_L\), this is in general less than the single level estimator’s variance. Note that the term
\[
\sum_{i=1}^{M_L} \frac{g(X_{L-1}(T;\omega_{L-1,i}))}{M_L} - \sum_{i=1}^{M_{L-1}} \frac{g(X_{L-1}(T;\omega_{L-1,i}))}{M_{L-1}}
\]
of equation (3.2.2) acts as a control variate with expected value 0. Arguing iteratively like this, the variance decreases until the number of levels in the estimator has reached \(O(|\log(TOL)|)\). For more details on the multilevel variance reduction idea, see Giles [10], and for more on control variates, see [12, Chapter 22].

3.3 Results

We have developed a multilevel Monte Carlo method which extends Giles work on uniform grids to adaptive grids. This makes it possible to compute expected value approximations for more general problems than is possible on uniform grids.

Figure 3.1 compares the two methods’ computational cost of approximating \(E[g(X_\tau,\tau)]\) for the stopped diffusion problem \(\tau := \max\{t \in [0,2]|X(t) < 2\}\), \(g(x,t) := x^3e^{-t}\) and
\[
dX(t) = \frac{11}{36}X(t) \, dt + \frac{1}{6}X(t) \, dW(t), \quad \text{for } t \in [0,2] \text{ and } X(t) \in (-\infty, 2)
\]
(3.3.1)
\[X(0) = 1.6.\]
3.3. RESULTS

Figure 3.1: Experimental complexity for the barrier example in Section 3.3. The computational cost of the multilevel adaptive algorithm is shown for varying tolerances using different initial states in the pseudo random number algorithm. A least squares fit, in $\log_2 - \log_2$-scale, of the model $cost = c_1 \left( \log \left( \frac{Tol_0}{Tol} \right) / Tol \right)^{c_2}$ with equal weight on all observations results in $c_1 = 12$ and $c_2 = 1.9$. One realisation of the corresponding cost using a single level implementation of the same adaptive Monte Carlo method is included for reference.
Bibliography


