Multi-scale methods for stochastic differential equations

by

Niklas Zettervall

Department of Physics
Umeå University

February 2012
Abstract

Standard Monte Carlo methods are used extensively to solve stochastic differential equations. This thesis investigates a Monte Carlo (MC) method called multilevel Monte Carlo that solves the equations on several grids, each with a specific number of grid points. The multilevel MC reduces the computational cost compared to standard MC. When using a fixed computational cost the variance can be reduced by using the multilevel method compared to the standard one. Discretization and statistical error calculations are also being conducted and the possibility to evaluate the errors coupled with the multilevel MC creates a powerful numerical tool for calculating equations numerically. By using the multilevel MC method together with the error calculations it is possible to efficiently determine how to spend an extended computational budget.

Sammanfattning

Acknowledgements

I would like to express my sincere thanks to Kaj Nyström at Uppsala University and to Oleg Seleznjev at Umeå University for their help and guidance in this master thesis.
Contents

1 Project background 7

2 Introduction to standard Monte Carlo 7
   2.1 Options 8
      2.1.1 European Option 9
      2.1.2 Asian Option 9
      2.1.3 Lookback Option 9

3 Brownian Motion 9
   3.1 Geometric Brownian Motion 10

4 Stochastic Differential Equations 11
   4.1 Introduction 11
   4.2 Ito’s formula 12
   4.3 Solving a SDE 13
   4.4 Discretization schemes 14
      4.4.1 Euler-Maruyama scheme 14
      4.4.2 Milstein scheme 15

5 The multilevel Monte Carlo method 15
   5.0.3 Complexity theorem 17
   5.1 Estimating the remaining bias 18
   5.2 Choosing an optimal $N_l$ 18
   5.3 Numerical algorithm 19
   5.4 Multilevel Monte Carlo and the
       Euler-Maruyama scheme 19
      5.4.1 European Call Option 19
      5.4.2 Asian Call Option 19
      5.4.3 Lookback Call Option 19
   5.5 Multilevel Monte Carlo and the Milstein scheme 20
      5.5.1 European Call Option 20
      5.5.2 Asian Call Option 20
      5.5.3 Lookback Call Option 21

6 Adaptive weak approximation of stochastic differential equations 22
   6.1 Introduction 22
   6.2 An error estimate 23
      6.2.1 Error expansion 23
      6.2.2 Calculation of $u$ 24
      6.2.3 Discrete dual functions 26
      6.2.4 Statistical error 27
   6.3 Cauchy problem 27
6.4 Algorithm for the Cauchy problem .......................... 29

7 Software Issues 29
  7.1 C/C++ ......................................................... 29
  7.2 Random Number Generator .................................. 30

8 Results 30
  8.1 Euler-Maruyama scheme ...................................... 30
     8.1.1 European call option .................................... 30
     8.1.2 Asian call option ....................................... 33
     8.1.3 Lookback call option ................................... 34
  8.2 Milstein scheme .............................................. 37
     8.2.1 European call option .................................... 37
     8.2.2 Asian call option ....................................... 38
     8.2.3 Lookback call option ................................... 39
  8.3 Cauchy problem .............................................. 40

9 Conclusions 44

References 47

List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>European option, Euler-Maruyama scheme. $2 \times 10^6$ sample paths used on each level.</td>
</tr>
<tr>
<td>2</td>
<td>Number of additional samples for a European option using the Euler-Maruyama discretization scheme.</td>
</tr>
<tr>
<td>3</td>
<td>Asian option, Euler-Maruyama scheme. $2 \times 10^6$ sample paths used on each level.</td>
</tr>
<tr>
<td>4</td>
<td>Number of additional samples for an Asian option using the Euler-Maruyama discretization scheme.</td>
</tr>
<tr>
<td>5</td>
<td>Lookback option, Euler-Maruyama scheme. $2 \times 10^6$ sample paths used on each level.</td>
</tr>
<tr>
<td>6</td>
<td>Number of additional samples for a Lookback option using the Euler-Maruyama discretization scheme.</td>
</tr>
<tr>
<td>7</td>
<td>European option, Milstein scheme. $2 \times 10^6$ sample paths used on each level.</td>
</tr>
<tr>
<td>8</td>
<td>Number of additional samples for a European option using the Milstein discretization scheme.</td>
</tr>
<tr>
<td>9</td>
<td>Asian option, Milstein scheme. $2 \times 10^6$ sample paths used on each level.</td>
</tr>
<tr>
<td>10</td>
<td>Number of additional samples for an Asian option using the Milstein discretization scheme.</td>
</tr>
</tbody>
</table>
Lookback option, Milstein scheme. $2 \times 10^6$ sample paths used on each level.  

Number of additional samples for a Lookback option using the Milstein discretization scheme.  

Plots intended to illustrate the convergence rate for the mean value and variance.  

Optimal number of trajectories for the Cauchy problem for the eight different values of $\varepsilon$.  

List of Tables

1. Table showing results from multilevel MC and standard MC for a European option. Note the relatively steep decrease in the value of $\varepsilon$ and the wide range in the number of step sizes used.  
2. Table showing the quotient between the variance for standard MC and multilevel MC simulations for a European options for decreasing $\varepsilon$. Remark that the quotient is roughly equal when the same number of time steps are being used, even though the $\varepsilon$ is different.  
3. Table showing results from multilevel MC and standard MC for an Asian option.  
4. Table showing the quotient between the variance for standard MC and multilevel MC simulations for an Asian option for decreasing $\varepsilon$.  
5. Table showing results from multilevel MC and standard MC for a Lookback option.  
6. Table showing the quotient between the variance for standard MC and multilevel MC simulations for a Lookback option for decreasing values on $\varepsilon$.  
7. Table showing the quotient between the variance for MC and multilevel MC simulations for European options for decreasing $\varepsilon$ when using the Milstein scheme.  
8. Table showing the quotient between the variance for MC and multilevel MC simulations for an Asian option for decreasing $\varepsilon$ when using the Milstein scheme.  
9. Table showing the quotient between the variance for MC and multilevel MC simulations for Lookback options for decreasing $\varepsilon$ when using the Milstein scheme.  
10. Value and error estimate for the Cauchy problem for the multilevel MC case.  
11. Quotient of the two largest error estimates from the multilevel MC simulations for the Cauchy problem.  
12. Value and error estimate for the standard MC case for the Cauchy problem.
1 Project background

The idea with this thesis is to investigate a special type of MC method called multilevel MC. The multilevel MC has been developed by Prof. Mike B. Giles at Oxford University. It differs from standard MC since it uses a multigrid system to solve an equation, instead of just using a single grid. By solving the equation on several grids, each with a different number of time steps, the computational time is reduced. This means that multilevel MC can achieve a lower variance than standard MC for a fixed computational cost. To prove this, a set of examples are used and comparisons are made between the two MC methods. The thesis also investigates how to combine a method for calculating the statistical and discretization errors with the multilevel MC method. The papers used for the calculation of the statistical and discretization errors can be found in [4] and [5]. The aim of this thesis is to combine the multilevel MC with the calculations of discretization error. The calculations will be efficient, thanks to the multilevel MC, and in the same time it becomes possible to calculate the different errors, statistical and discretization, that occurs meaning that it is possible to determine where to spend an increased computational budget as efficiently as possible.

The thesis starts off by introducing the standard MC method, and then introducing relevant theory such as stochastic differential equations, Brownian motion and different discretization schemes. Then the multilevel MC method is presented together with the calculation of the error estimates. The thesis finishes off by presenting a set of examples designed to illustrate and compare the methods. Finally the conclusions are made based on the examples used.

2 Introduction to standard Monte Carlo

MC methods are ways where repeated randomized sampling is used to estimate solutions to certain problems [1]. It can for example be used to estimate the value of an integral or it can estimate ferro magnetism. Now consider the integral of the function \( f(x) \) over the unit interval,

\[
\Gamma = \int_0^1 f(x)dx.
\]  

(1)

The integral may be represented as an expectation, \( E[f(u)] \), where \( u \sim U(0, 1) \). Consider sampling \( N \) independent samples \( u_i \) from \( U(0, 1) \) and use these to calculate an arithmetic mean value of \( f(u_i), i = 1, \ldots, N \). The arithmetic mean value is often used as the MC estimator, and it is written as

\[
\hat{\Gamma} = \frac{1}{N} \sum_{i=1}^{N} f(u_i).
\]  

(2)
Provided that \( f(x) \) is computable over the unit interval, the law of large numbers will ensure that
\[
\hat{\Gamma} \to \Gamma
\]  
with probability 1 as \( N \to \infty \). If \( f(x) \) is square integrable and
\[
\sigma_f^2 = \int_0^1 (f(x) - \Gamma)^2 dx,
\]  
the error in the MC estimate is approximately \( N(0, \sigma_f/\sqrt{N}) \). If \( \sigma_f \) is unknown, the sample standard deviation is used instead,
\[
s_f = \sqrt{\frac{1}{N-1} \sum_{i=1}^{N} (f(u_i) - \hat{\Gamma})^2}.
\]  
The MC error estimate has a convergence rate of \( 1/\sqrt{N} \) and this inverse rate to the square of the number of samples plays a crucial role. To cut the error estimate in half, the number of samples needs to be increased by a factor four. Compared to even simple ways of calculating integrals, such as the trapezoidal rule, the MC estimate has a far worst convergence rate. MC simply cannot compete with integration methods when it comes to one-dimensional problems. However, when the dimension of the problem increases, the MC methods keeps its convergence rate while other methods decrease theirs, making the MC method the obvious choice. This fact makes the MC method a highly desirable method in computational finance since the number of dimensions is often high.

MC methods are used extensively in computational finance to evaluate the expected value of a quantity which is a functional of the solution to a stochastic differential equation (SDE). Suppose we have a SDE with general drift and volatility terms,
\[
dS(t) = a(S,t)dt + b(S,t)dW(t)
\]  
for \( 0 < t < T \) with \( W \) a standard Brownian motion. A strong solution to (6) on an interval \([0, T] \) is an Itô process \( \{S(t), 0 \leq t \leq T\} \), where
\[
S(t) = S(0) + \int_0^t a(S(u), u) du + \int_0^t b(S(u), u) dW(u).
\]  
Equations such as equation (6) are often used to describe the evolution of underlying asset prices, model parameters, interest rates and other financial derivatives.

2.1 Options

An option is a financial instrument that specifies a contract between two parties that ensures the buyer of the option the right, but not the obligation, to engage
in a future transaction \[1\]. The price of the option is the difference between the underlying asset, for example a share or a bond, and the reference price plus a premium based on the time remaining until the expiration of the option. If the option ensures someone to buy something at a specific price, it is known as a call option. An option that ensures someone to sell something at a specified price is referred to as a put option. The options treated in this report are solely call options but the difference in the calculations between the call and put options used in this paper are small.

2.1.1 European Option

A European option is probably the easiest form of options. The European option may only be exercised at the date when the option expires. This means that the date of expiry is the same as the end date on the interval \([0, T]\) on which the option is calculated. In the real world, a European option expires on the Friday prior the third Saturday every month.

2.1.2 Asian Option

An Asian option is a special kind of option where the payoff of the option is completely determined by the arithmetic mean value of the underlying price over the time \([0, T]\). It falls under the category known as the exotic options.

2.1.3 Lookback Option

A Lookback option is, just like the Asian option, a path dependent option from the range of so called exotic options. The payoff depends on the maximum or minimum value of the underlying asset price conditioned on \([0, T]\).

3 Brownian Motion

Brownian motion is the simplest of the stochastic processes called diffusion processes \([1], [11]\) and it is also a Gaussian process. In fact, all other diffusion processes can be described in terms of Brownian motion. A standard one-dimensional Brownian motion on the interval \([0, T]\) is a stochastic process \(\{W(t), 0 \leq t \leq T\}\) where

i. \(W(0) = 0,\)

ii. the mapping \(t \mapsto W(t)\) is, with probability 1, a continuous function on \([0, T],\)

iii. the increments \(\{W(t_1) - W(t_0), W(t_2) - W(t_1), \ldots, W(t_i) - W(t_{i-1})\}\) are independent for any \(i\) and any \(t_i > t_{i-1}\) in \([0, T]\),
iv. $W(t) - W(s) \sim \mathcal{N}(0, t - s)$.

From i. and iv. it is clear that $W(t) \sim \mathcal{N}(0, t)$ for $0 \leq t \leq T$. A stochastic process $X(t)$ is a Brownian motion if

$$\frac{X(t) - \mu t}{\sigma}$$

is a standard Brownian motion with the drift and diffusion constants $\mu$ and $\sigma > 0$. This means it is possible to construct a stochastic process $X(t)$,

$$X(t) = \mu t + \sigma W(t)$$

with a standard Brownian motion $W(t)$ that solves the stochastic differential equation

$$dX(t) = \mu dt + \sigma dW(t).$$

It is also possible to construct a SDE with time varying drift and diffusion coefficients,

$$dX(t) = \mu(t)dt + \sigma(t)dW(t).$$

### 3.1 Geometric Brownian Motion

A stochastic process $X(t)$ is a geometric Brownian motion if $\log(X(t))$ is a Brownian motion with initial value $\log(X(0))$. In order to create a geometric Brownian motion simply exponentiate a Brownian motion. Geometric Brownian motion has the attractive feature that it is never negative which fits well when modelling stock prices since a stock can never attain a negative price. Also, geometric Brownian motion has independent percentage changes,

$$\frac{X(t_i) - X(t_{i-1})}{X(t_{i-1})}$$

rather than independent absolute changes $X(t_i) - X(t_{i-1})$. A stochastic process is said to follow a geometric Brownian motion if it satisfies

$$dX(t) = \mu X(t)dt + \sigma X(t)dW(t)$$

which has the solution

$$X(t) = X(0) \exp \left( \left( \mu - \frac{1}{2}\sigma^2 \right)t + \sigma W(t) \right).$$
4 Stochastic Differential Equations

4.1 Introduction

Consider the following stochastic differential equation [11],

\[
\begin{aligned}
    dX(t) &= a(X(t))dt + b(X(t))dW(t) \\
    X(0) &= x_0.
\end{aligned}
\]  

(15)

\(X(t)\) solves (15) provided

\[
X(t) = x_0 + \int_0^t a(X(s))ds + \int_0^t b(X(s))dW, \quad 0 \leq t \leq T.
\]

(16)

If \(X(t)\) can be described as in (16) it is called an Itô process if \(X(0)\) is \(\mathcal{F}\)-measurable, \(a\) is an \(\mathbb{R}^d\)-valued adapted process satisfying

\[
P\left( \int_0^T |a(t)|dt < \infty \right) = 1
\]

(17)

where \(i = 1, \ldots, d\) and \(b\) is an \(\mathbb{R}^{d \times k}\)-valued adapted process satisfying

\[
P\left( \int_0^T ||b(t)||^2dt < \infty \right) = 1.
\]

(18)

A SDE is said to be linear if \(a(t, X(t)) = \alpha_1(t)X(t) + \alpha_2\) and \(b(t, X(t)) = \beta_1(t)X(t) + \beta_2\) for some constants \(\alpha_1, \alpha_2, \beta_1, \beta_2\). A linear SDE is said to

i. be autonomous if all coefficients are constants;

ii. be homogenous if \(\alpha_2 = 0\) and \(\alpha_2 = 0\);

iii. be linear in the narrow sense if \(\beta_1 = 0\);

iv. have multiplicable noise if \(\beta_2 = 0\).

Equation (19) shows the SDE used in the Black-Scholes model to describe the evolution of a stock price.

\[
\frac{dS(t)}{S(t)} = r(t, S(t))dt + \sigma(t, S(t))dW(t),
\]

(19)

with \(W\) a standard Brownian motion. \(\frac{dS(t)}{S(t)}\) may be thought of as the percentage changes in the stock price as the increments of a Brownian motion. \(r\) is the interest rate, \(\sigma\) is the volatility of the stock price and \(dt\) is the mean rate of return. The risk-neutral dynamics of the stock price occurs when the interest
rate is the same as the mean rate of return. If \( r \) and \( \sigma \) are constants the solution to (19) is written as

\[
S(t) = S(0) \exp \left( (r - \frac{1}{2} \sigma^2) t + \sigma W(t) \right),
\]

(20)

where \( S(0) \) is the stock price at time 0 and the \( W(t) \) is a normally distributed random variable with mean value 0 and variance \( t \). \( W(t) \) can be simulated as \( \sqrt{t} Z \) where \( Z \sim \mathcal{N}(0, 1) \). If \( u < t \), equation (20) can be written as

\[
S(t) = S(u) \exp \left( (r - \frac{1}{2} \sigma^2) (t - u) + \sigma (W(t) - W(u)) \right).
\]

(21)

\( S(t) \) is a log-normally distributed random variable with expected value

\[
E[S(t)] = S(u) \exp(r(t - u))
\]

(22)

and variance

\[
V[S(t)] = S(u)^2 \exp(2r(t - u))(\exp(\sigma^2(t - u)) - 1).
\]

(23)

The stock price of (20) can be simulated using

\[
S(t_{i+1}) = S(t_i) \exp \left( (r(t_i) - \frac{1}{2} \sigma^2(t_i)^2) (t_{i+1} - t_i) + \sigma(t_i) \sqrt{(t_{i+1} - t_i)Z_{i+1}} \right).
\]

(24)

where \( Z_{i+1} \sim \mathcal{N}(0, 1) \).

### 4.2 Ito’s formula

Assume \( X(\cdot) \) is a stochastic process satisfying (16) and has a stochastic differential

\[
dX(t) = r(X(t))dt + \sigma(X(t))dW(t).
\]

(25)

Assume \( u : \mathbb{R} \times [0, T] \rightarrow \mathbb{R} \) is continuous and \( \frac{\partial u}{\partial t}, \frac{\partial u}{\partial x}, \frac{\partial^2 u}{\partial x^2} \) exist and are continuous. Set

\[
Y(t) = u(X(t), t).
\]

(26)

Then \( Y \) has the stochastic differential

\[
dY = \frac{\partial u}{\partial t} dt + \frac{\partial u}{\partial x} dX + \frac{1}{2} \frac{\partial^2 u}{\partial x^2} \sigma^2 dt
\]

\[
= \left( \frac{\partial u}{\partial t} + \frac{\partial u}{\partial x} r + \frac{\partial^2 u}{\partial x^2} \sigma^2 \right) dt + \frac{\partial u}{\partial x} \sigma dW.
\]

(27)

Equation (27) is called Ito’s formula or Ito’s chain rule.
4.3 Solving a SDE

Up until now the solution to equation (19) has been presented but nowhere it has been shown how to get that result. In this section the process of solving this equation will be presented in order to get a deeper understanding of the nature of the SDE. Let's start with a simple example of a SDE,

\[
\begin{cases}
  dX = gX dW \\
  X(0) = 1
\end{cases}
\]  

(28)

where \( g(t) \) is a continuous function. The solution to (28) is

\[
X(t) = \exp \left( -\frac{1}{2} \int_0^t g^2 ds + \int_0^t g dW \right). 
\]

(29)

Note that, by using equation (16),

\[
Y(t) = -\frac{1}{2} \int_0^t g^2 ds + \int_0^t g dW
\]

(30)

can be verified to satisfy

\[
dY(t) = -\frac{1}{2} g^2 dt + gdW.
\]

(31)

Insert \( u(x) = e^x \) into Ito’s lemma to get

\[
dX = \frac{\partial u}{\partial x} dY + \frac{1}{2} \frac{\partial^2 u}{\partial x^2} g^2 dt =
\]

\[
e^Y \left( -\frac{1}{2} g^2 dt + gdW + \frac{1}{2} g^2 dt \right) = gX dW.
\]

(32)

Now consider the following SDE,

\[
\begin{cases}
  dX = r X dt + \sigma X dW \\
  X(0) = x_0
\end{cases}
\]  

(33)

with the solution

\[
X(t) = x_0 \exp \left( \int_0^t \left( r - \frac{1}{2} \sigma^2 \right) ds + \int_0^t \sigma dW \right)
\]

(34)

for any \( t \in [0,T] \). If \( r > 0 \) and \( \sigma \) are constants, Ito’s formula gives

\[
d(\log(X)) = \frac{dX}{X} - \frac{1}{2} \frac{\sigma^2 X^2}{X^2} = \left( \mu - \frac{\sigma^2}{2} \right) dt + \sigma dW.
\]

(35)
and hence
\[ X(t) = x_0 \exp \left( \left( r - \frac{\sigma^2}{2} \right) t + \sigma W(t) \right) \]  
(36)

which could be seen in an earlier example. Note that as long as \( x_0 \) is positive, (36) will always be positive. This is an appealing feature when pricing stocks since they can never attain a negative value. If equation (33) is rewritten as
\[ dX = rXdt + \sigma dW \]  
(37)

it is easy to see that it verifies
\[ X(t) = x_0 + \int_0^t rXds + \int_0^t \sigma XdW \]  
(38)

and because
\[ E \left[ \int_0^t \sigma XdW \right] = 0, \]  
(39)

which gives
\[ E[X(t)] = x_0 + \int_0^t rE[X(s)]ds \]  
(40)

and hence
\[ E[X(t)] = x_0 e^{rt}, t \geq 0. \]  
(41)

This means the expected value of the stock price agrees with the deterministic solution in equation (36) corresponding to \( \sigma = 0 \).

4.4 Discretization schemes

In order to use equation (19) in the MC simulation, a discretization scheme will be needed [1].

4.4.1 Euler-Maruyama scheme

The first scheme to be presented is the easiest one, the Euler-Maruyama scheme. It is written as
\[ S(t_{i+1}) = S(t_i) + r(t_i, S(t_i))(t_{i+1} - t_i) + \sigma(t_i, S(t_i))\sqrt{t_{i+1} - t_i}Z_{i+1} \]  
(42)

where \( Z_{i+1} \sim N(0,1) \). This scheme has a strong convergence order of \( \frac{1}{2} \) and a weak convergence order of 1 even though it can achieve better order of convergence for some cases. For \( r(t, S) = \theta_1 S \) and \( \sigma(t, S) = \theta_2 S \) scheme (42) looks like
\[ S(t_{i+1}) = S(t_i) + \theta_1 S(t_i)(t_{i+1} - t_i) + \theta_2 S(t_i)\sqrt{t_{i+1} - t_i}Z_{i+1}. \]  
(43)
4.4.2 Milstein scheme

One improved scheme, is the Milstein scheme,

\[
S(t_{i+1}) = S(t_i) + r(t_i, S(t_i))(t_{i+1} - t_i) + \sigma(t_i, S(t_i))(W_{i+1} - W_i) \\
+ \frac{1}{2}\sigma(t_i, S(t_i))\sigma_x(t_i, S(t_i)) \left( (W_{i+1} - W_i)^2 - (t_{i+1} - t_i) \right),
\]

or in a more easily read form,

\[
S_{t_{i+1}} = S_t + r\Delta t + \sigma \Delta W + \frac{1}{2}\sigma \sigma_x ((\Delta W)^2 - \Delta t).
\]

The Milstein scheme has strong and weak orders of convergence equal to 1. Now consider the special case where \( r(t, x) = \theta_1 x \) and \( \sigma(t, x) = \theta_2 x \) and thus \( \sigma_x(t, x) = 0 \). In this case the Milstein scheme and the Euler-Maruyama scheme coincides and the Euler-Maruyama scheme will have a strong order of convergence of 1, just as Milstein the scheme.

For \( r(t, S) = \theta_1 S, \sigma(t, S) = \theta_2 S \) and \( \sigma = \theta_2 \) the scheme in (43) looks like

\[
S(t_{i+1}) = S(t_i) + \theta_1 S(t_i)(t_{i+1} - t_i) + \theta_2 S(t_{i+1} - t_i)(W_{i+1} - W_i) \\
+ \frac{1}{2}\theta_2 S(t_i)\theta_2 \left( (W_{i+1} - W_i)^2 - (t_{i+1} - t_i) \right).
\]

This is the form that will be used in the examples below.

5 The multilevel Monte Carlo method

The multilevel method uses a multigrid algorithm in order to reduce the computational complexity when computing an expected value arising from a stochastic differential equation \[2\], \[6\]. The idea is based on using grids with different number of time steps and by using the information from each grid, the computational complexity can be significantly reduced. The multilevel method calculates the value on the finest grid by using corrections from all the other grids and is thereby getting the accuracy associated with the finest grid at a much lower cost, or to put it in another way, it receives a lower variance for a fixed computational cost then standard MC.

Consider MC path simulation on grids with different time steps \( h_l = M^{-l}T, l = 0, \ldots, L \). Let \( \hat{P}_l \) denote the approximation to the payoff \( f(S(t)) \) and let \( \hat{S}_l \) denote the approximation of \( S(t) \) where \( \hat{P}_l \) and \( \hat{S}_l \) are being calculated for a given Brownian path \( W(t) \). The expectation on the finest grid can be calculated using

\[
E[\hat{P}_L] = E[\hat{P}_0] + \sum_{l=1}^{L} E[\hat{P}_l - \hat{P}_{l-1}].
\]

15
Equation (47) clearly shows the idea behind the multilevel method. It calculates the value on the finest grid by using corrections from the coarser grids. A simple estimator for each size \( l \) is the arithmetic mean value, where each level \( l \) uses \( N_l \) samples,

\[
\hat{Y}_l = N_l^{-1} \sum_{i=1}^{N_l} (\hat{P}_l^{(i)} - \hat{P}_{l-1}^{(i)}) ,
\]

(48)

The variance of this estimator is \( V[\hat{Y}_l] = V_l N_l^{-1} \) where \( V_l \) is the variance for a single sample. The combined estimator is

\[
\hat{Y} = \sum_{l=0}^{L} \hat{Y}_l
\]

(49)

and the variance for this combined estimator is

\[
V[\hat{Y}] = \sum_{l=0}^{L} V_l N_l^{-1}.
\]

(50)

The key points here is that \( \hat{P}_l \) and \( \hat{P}_{l-1} \) comes from two discrete approximations with different time steps but the same Brownian path. This is done by creating the Brownian increments for the fine path for the calculation of \( \hat{P}_l^{(i)} \) and then summing these increments into groups of size \( M \) for the calculation of \( \hat{P}_{l-1}^{(i)} \).

The total cost of the calculations of all the grids are approximately proportional to

\[
\sum_{l=0}^{L} h_l^{-1} N_l.
\]

(51)

By treating \( N_l \) as continuous variables the variance is minimized for a fixed computational cost by choosing \( N_l \propto \sqrt{V_l h_l} \). This relation is used in the calculation of an optimal \( N_l \) in equation (61).

The Euler-Maruyama discretization scheme provides \( O(h^{1/2}) \) strong convergence and \( O(h^1) \) weak convergence. This order of convergence holds for both standard MC and multilevel MC. In MC, the mean square error (MSE), when using the arithmetic mean value as the estimator, is

\[
MSE \approx c_1 N^{-1} + c_2 h^2
\]

(52)

for \( c_i > 0, i = 1, 2 \). The first term corresponds to the variance of the mean value and the second term comes from the bias introduced by the Euler-Maruyama or Milstein discretization. In order to get a solution that has a MSE of \( O(\varepsilon^2) \) when using standard MC, \( N = O(\varepsilon^{-2}) \) and \( h = O(\varepsilon) \) is required. This means that the computational complexity is \( O(\varepsilon^{-3}) \). With the multilevel MC however, this computational complexity reduces to \( O(\varepsilon^{-2} (\log \varepsilon)^2) \) when using the Euler-Maruyama scheme and it reduces to \( O(\varepsilon^{-2}) \) if the Milstein scheme is used.
5.0.3 Complexity theorem

The complexity theorem is worded quite generally and it is applicable to a wide variety of financial models. The theorem can be found in [2].

**Theorem 5.1.** Let $P$ denote a functional of the solution of stochastic differential equation (6) for a given Brownian path $W(t)$, and let $\hat{P}_l$ denote the corresponding approximation using a numerical discretization with time step $h_l = M^{-1}T$. If there exist independent estimators $\hat{Y}_l$ based on $N_l$ MC samples, and positive constants $\alpha \geq \frac{1}{2}$, $\beta$, $c_1$, $c_2$, $c_3$ such that

i. $|E[\hat{P}_l - P]| \leq c_1 h_l^\alpha$

ii. $E[\hat{Y}_l] = \begin{cases} E[\hat{P}_0] & l = 0 \\ E[\hat{P}_l - P] & l > 0 \end{cases}$

iii. $V[\hat{Y}_l] \leq c_2 N_l^{-1} h_l^\beta$

iv. $C_l$, the computational complexity of $\hat{Y}_l$, is bounded by

$$C_l \leq c_3 N_l h_l^{-1},$$

(53)

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

$$\hat{Y}_l = \sum_{l=0}^{L} \hat{Y}_l,$$

(54)

has a mean square error with bound

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

(55)

with a computational complexity $C$ with bound

$$C \leq \begin{cases} c_4 \varepsilon^{-2}, & \beta > 1 \\ c_4 \varepsilon^{-2} (\log \varepsilon)^2, & \beta = 1 \\ c_4 \varepsilon^{-2((1-\beta)/\alpha)}, & 0 < \beta < 1. \end{cases}$$

(56)

When taking a closer look at the theorem it becomes clear that the parameter $\beta$ plays an important role since it defines the convergence rate of the variance at each level as $l \to \infty$. The Milstein scheme, with its higher value for $\beta$, can
be used to increase the order of strong convergence so that $V_{l} = \mathcal{O}(h_{l}^{2})$. The computational cost of the multilevel MC is calculated using

$$C = N_{0} + \sum_{l=1}^{L} N_{l} (M^{l} + M^{l-1}).$$

(57)

When $l > 0$ the calculations uses two grids to calculate $\hat{P}_{l} - \hat{P}_{l-1}$ and that is why the term $(M^{l} + M^{l-1})$ is included.

### 5.1 Estimating the remaining bias

The remaining bias due to the discretization needs to be estimated and controlled. A way of doing this is to use the information available in the estimate of the correction $E[\hat{P}_{l} - \hat{P}_{l-1}]$. Combining equation (52) with the behavior of the bias as $l \to \infty$ gives

$$E[P - \hat{P}] \approx c_{1} h_{l},$$

(58)

where $c_{1}$ is a constant. Hence

$$E[\hat{P}_{l} - \hat{P}_{l-1}] \approx (M - 1)c_{1} h_{l} \approx (M - 1)E[P - \hat{P}].$$

(59)

This information will be used as a bound that tries to make sure the bias is less than $\varepsilon/\sqrt{2}$. This bound will be written as a convergence test which determines when the multilevel algorithm will stop. This test looks like

$$\max \left\{ M^{-1} \left| \hat{Y}_{L-1} \right|, \left| \hat{Y}_{L} \right| \right\} < \frac{1}{\sqrt{2}} (M - 1)\varepsilon.$$

(60)

and it will be used in all the following examples.

### 5.2 Choosing an optimal $N_{l}$

The optimal $N_{l}$ is calculated using

$$N_{l} = \left[ 2\varepsilon^{-2} \sqrt{V_{l} h_{l}} \left( \sum_{l=0}^{L} \sqrt{V_{l}/h_{l}} \right) \right].$$

(61)

This value for $N_{l}$ ensures that the variance of the combined estimator becomes less than $\varepsilon^{2}/2$. Equation (60) tries to make sure that the remaining bias falls below $\varepsilon/\sqrt{2}$ and together they try to keep the MSE below $\varepsilon^{2}$. Note also that $N_{l}$ is proportional to $\sqrt{V_{l} h_{l}}$ in order to minimize the variance for a fixed computational cost.
5.3 Numerical algorithm

The numerical algorithm looks like,

1. start with $L = 0$
2. estimate $V_L$ using an initial $N_L = 10^4$ samples
3. define optimal $N_l, l = 0, \ldots, L$ using equation (61)
4. evaluate extra samples at each level as needed for new $N_l$
5. if $L \geq 2$, test for convergence using equation (60)
6. if $L < 2$ or not converged, set $L := L + 1$ and go to 2.

5.4 Multilevel Monte Carlo and the Euler-Maruyama scheme

To demonstrate how the multilevel MC method works, a series of examples using option pricing will be used [2]. The theory behind the options used is presented below and the three option types used will be the same ones described earlier.

5.4.1 European Call Option

The payoff for the European option is $P = \exp(-rT) \max(0, S(T) - K)$ where $K$ is the strike price, $T$ is the exercise date and $r$ is the interest rate.

5.4.2 Asian Call Option

The payoff for the Asian call option is $P = \exp(-rT) \max(0, \bar{S} - K)$ where $\bar{S}$ is calculated using the arithmetic mean

$$\bar{S} = \frac{1}{n_t} \sum_{i=0}^{n_t-1} \tilde{S}_i.$$  \hspace{1cm} (62)

5.4.3 Lookback Call Option

The payoff for the Lookback option is

$$P = \exp(-rT) \left( S(T) - \min_{0 < t < T} S(t) \right).$$  \hspace{1cm} (63)

The minimum value for $S(t)$ is calculated using

$$\tilde{S}_{\min} = \min_n \tilde{S}_n \left( 1 - \beta^* \sigma \sqrt{h_t} \right)$$  \hspace{1cm} (64)

where $\beta^* \approx 0.5826$ is a constant used to make sure that the $O(h_t)$ weak convergence is achieved.
5.5 Multilevel Monte Carlo and the Milstein scheme

When using the Milstein scheme, the estimator construction must be changed in order to correctly respect identity (47) and to avoid undesired bias [3]. The following must hold,

\[ E[\hat{P}_t^f] = E[\hat{P}_t^c] \]  

(65)

where \( \hat{P}_t^f \) corresponds to the fine path and \( \hat{P}_t^c \) corresponds to the coarse path. Relation (65) ensures that the definitions of \( \hat{P}_t \), when estimating \( E[\hat{P}_t - P_{t-1}] \) and \( E[\hat{P}_{t+1} - P_t] \), has the same expectation.

5.5.1 European Call Option

The payoff is calculated in the same way as for the Euler-Maruyama example.

5.5.2 Asian Call Option

The Asian call option considered in the Milstein example has the same payoff as in the case of Euler-Maruyama. \( \hat{S} \) is calculated using

\[ \hat{S} = \int_0^T S(s)ds. \]  

(66)

This can be approximated as

\[ \hat{S}_t = \sum_{n=0}^{n_t-1} \frac{1}{2} \left( \hat{S}_n + \hat{S}_{n+1} \right) h_t \]  

(67)

where \( n_t \) is the number of time steps. This is the same as taking a piecewise linear approximation to \( S_n \) but by approximating the behavior within the time step \([t_i, t_{i+1}]\) as Brownian motion an even better approximation can be achieved. By keeping the drift and volatility constant within the time steps and by using the drift and volatility based on \( S_n \) a Brownian bridge results give

\[ \int_{t_n}^{t_{n+1}} S(t)dt = \frac{1}{2} h (S(t_n) + S(t_{n+1})) + \sigma_n \Delta I_n \]  

(68)

with \( \Delta I_n \) defined as

\[ \Delta I_n = \int_{t_n}^{t_{n+1}} (W(t) - W(t_n))dt - \frac{1}{2} h \Delta W \]  

(69)

where \( \Delta I_n \sim N(0, h^3/12) \). The fine path approximation is therefore

\[ \bar{S} = \frac{1}{T} \sum_{0}^{n_t-1} \left( \frac{1}{2} h (\hat{S}_n + \hat{S}_{n+1}) + \sigma_n \Delta I_n \right). \]  

(70)
This approximation holds for both the fine and the coarse path except that in the latter case, the values for \( \Delta I_n \) are derived from the fine path values.

\[
\int_{t_n}^{t_{n+2h}} (W(t) - W(t_n)) dt - h(W(t_n + 2h) - W(t_n)) \\
= \int_{t_n}^{t_{n+h}} (W(t) - W(t_n)) dt - \frac{1}{2} h(W(t_n + h) - W(t_n)) \\
+ \int_{t_n+h}^{t_{n+2h}} (W(t) - W(t_n + h)) dt - \frac{1}{2} h(W(t_n + 2h) - W(t_n + h)) \\
+ \frac{1}{2} h(W(t_n + h) - W(t_n)) - \frac{1}{2} h(W(t_n + 2h) - W(t_n + h))
\]  

(71)

and hence

\[
\Delta I^c = \Delta I^{f_1} + \Delta I^{f_2} + \frac{1}{2} h (\Delta W^{f_1} - \Delta W^{f_2})
\]  

(72)

where \( \Delta I^c \) is the value for the coarse time step and \( \Delta I^{f_i}, \Delta W^{f_i}, i = 1, 2 \) are the values for the first and second fine time step.

### 5.5.3 Lookback Call Option

The Lookback option used in the Milstein example uses equation (63) for the payoff which is the same equation used in the Euler-Maruyama case.

However, equation (64) used earlier for \( S_{min} \) is no longer correct since the expected value for \( P_i - P_{i-1} \) becomes \( O(h_1^{1/2}) \) and the variance for that expected value becomes \( O(h_1^1) \). This is all fine in the Euler-Maruyama case because this is the best that the discretization scheme can achieve. In the Milstein case however, \( O(h_1^1) \) for the expected value and \( O(h_2^2) \) for the variance is expected and therefore some changes has to be made in order to get that convergence rate. To achieve a better convergence, approximate the behavior within a time step as a Brownian motion conditional on the computed value \( \hat{S}_n \). This will result in a minimum of a Brownian motion on an interval \([t_n, t_{n+1}]\), conditional on the end points, as

\[
\hat{S}_{n,min} = \frac{1}{2} \left( \hat{S}_n - \hat{S}_{n+1} - \sqrt{(\hat{S}_{n+1} - \hat{S}_n)^2 - 2\sigma_n^2 h \log U_n} \right)
\]  

(73)

where \( \sigma \) is the volatility and \( U_n \) is a uniform random variable from \([0, 1]\). This is how the fine path is defined to achieve the global minimum. For the coarse path a slightly different approach is needed. As before Brownian motion is assumed and the value at the midpoint of an interval is given by

\[
\hat{S}_{n+1/2} = \frac{1}{2} \left( \hat{S}_n + \hat{S}_{n+1} - \sigma_n D_n \right)
\]  

(74)

where

\[
D_n = W_{n+1} - 2W_{n+1/2} + W_n = (W_{n+1} - W_{n+1/2}) - (W_{n+1/2} - W_n)
\]  

(75)
where $D_n \sim \mathcal{N}(0, h)$. The minimum over the whole time step is the smaller of the minima for each of the above half-time steps, meaning that

\[
\hat{S}_{n,\text{min}} = \min \left\{ \frac{1}{2} \left( \tilde{S}_n + \tilde{S}_{n+1/2} \right) - \sqrt{\left( \tilde{S}_{n+1/2} - \tilde{S}_n \right)^2 - \sigma_n^2 h \log U_{1,n} , \frac{1}{2} \left( \tilde{S}_{n+1/2} + \tilde{S}_{n+1} \right) - \sqrt{\left( \tilde{S}_{n+1} - \tilde{S}_{n+1/2} \right)^2 - \sigma_n^2 h \log U_{2,n} } \right\}
\]

(76)

As in previous cases, the Brownian increments used for the fine path are also used in the case for the coarse path. Both $U_{1,n}$ and $U_{2,n}$ together with $D_n$ are the same as in the fine path calculations. It is important that the probability distributions for equation (73) and (76) are the same since that is what relation (65) requires.

6 Adaptive weak approximation of stochastic differential equations

As mentioned earlier, the MSE depends on a statistic error and a discretization error. The latter can be improved by using a discretization scheme with higher order of convergence but it can also be improved by using finer grids. A method to control the discretization error is presented and later an example will be used to show that it can be combined with the multilevel MC. This new method will be able to control the discretization error and also achieve a small statistical error [4], [5]. The results will be made using the Euler-Maruyama discretization scheme.

6.1 Introduction

Consider the Cauchy-Dirichlet problem

\[
\begin{aligned}
\partial_t v(t, x) + L v(t, x) &= 0, \quad \text{whenever } (t, x) \in (0, T) \times \mathbb{R}^n \\
v(T, x) &= g(x), \quad \text{whenever } x \in \mathbb{R}^n.
\end{aligned}
\]

(77)

where

\[
L = \frac{1}{2} \sum_{i,j=1}^{n} [\sigma \sigma^\star]_{ij}(t, x) \partial_{ij} + \sum_{i=1}^{n} \mu_i(t, x) \partial_i.
\]

(78)

Note that the notation $\partial_i f$ is an abbreviation for $\frac{\partial f}{\partial x_i}$, $\partial_{ij}$ for $\frac{\partial^2 f}{\partial x_i \partial x_j}$ and so on. A solution to (77) is

\[
u(t, x) = E[g(X(T))|X(t) = x]
\]

(79)

where $X(T)$ solves the SDE

\[
X(t) = x + \int_0^t \mu(s, X(s))ds + \sum_{j=0}^{n} \int_0^t \sigma_j(s, X(s))ds.
\]

(80)
Focus in the section will be on solving the quantity \( u(t, x) \). The discrete Euler-Maruyama approximation of \( X \) will be denoted by \( X^\Delta \), where \( \Delta \) defines a partition \( \{ t_k \}_{k=0}^N \) of the interval \([0, T]\). Let \( 0 = t_0 < t_1 < \cdots < t_{N-1} < t_N = T \) and \( \Delta t_k = t_{k+1} - t_k \), \( k \in \{0, \ldots, N-1\} \). A numerical approximation to \( u(t, x) \) is

\[
u^\Delta(t_k, x) = E(g(X^\Delta(T))|X^\Delta(t_k) = x), \quad k \in \{0, \ldots, N-1\}, \quad x \in \mathbb{R}^n. \tag{81}\]

Multilevel MC method will be used to solve for \( u^\Delta(t_k, x) \) on an interval \([0, T]\).

As noted earlier, the MSE depends on a statistical error and a discretization error. The statistical error can be divided into two parts, \( E^s \) and \( E^{M,s} \). The first error is due to the statistical error that arises when calculating \( u^\Delta(t_k, x) \) and the second error is the statistical error coming from the estimation of the discretization error, \( E^{M,s} \). The estimation of the errors \( E^{s} \) and \( E^{M,s} \) will use standard MC with \( M \) trajectories while \( E^s \) is derived using multilevel MC. In total, the approximation of \( u(0, x) \) can be written as

\[
u(0, x) = u^\Delta(x) + E^s + E^{M,s} + E^{M,s} + R^\Delta. \tag{82}\]

The MC estimator for \( u^\Delta(x) \) will be presented later.

### 6.2 An error estimate

Next an error expansion in a posteriori form will be presented, complete with discrete dual functions, calculation of \( u \) associated to the Cauchy-Dirichlet problem in (77) and one example to present the method.

#### 6.2.1 Error expansion

Consider the Cauchy problem

\[
\begin{align*}
\partial_t v(t, x) + L v(t, x) &= f(x, t), \quad \text{whenever } (t, x) \in (0, T) \times \mathbb{R}^n \\
v(T, x) &= g(x), \quad \text{whenever } x \in \mathbb{R}^n
\end{align*}
\tag{83}
\]

Let \( v_{g,f} \) be a solution to (83) given by the functions \( g, f \). Now let

\[
v_{g,f}^\Delta(t_k, x) = E\left[ g(X^\Delta(T)) - \int_{t_k}^T f(\mu_i, X^\Delta(t)) dt | X^\Delta(t_k) = x \right] \tag{84}
\]

be the approximation to (83) for \( k \in \{0, 1, \ldots, N-1\} \) and let

\[
\Delta_{g,f}^\Delta(t_k, x) = v_{g,f}(t_k, x) - v_{g,f}^\Delta(t_k, x) \tag{85}
\]

The operator \( L \) in (83) can be written as

\[
L = \mu_i(t, x) \partial_i + a_{ij}(t, x) \partial_{ij}. \tag{86}
\]
Let
\[ \epsilon_i^{(k)} = \mu_i(t_{k+1}, X^\Delta(t_{k+1})) - \mu_i(t_k, X^\Delta(t_k)) \]
\[ \epsilon_{ij}^{(k)} = a_{ij}(t_{k+1}, X^\Delta(t_{k+1})) - a_{ij}(t_k, X^\Delta(t_k)) \]

where \( a_{ij}(t, x) = \frac{1}{2} [\sigma \sigma^*]_{ij}(t, x) \). Introduce
\[
\Delta_{g,f}^\Delta(x) = \sum_{k=0}^{N-1} E \left[ \epsilon_i^{(k)} \partial_i v_{g,f} + \epsilon_{ij}^{(k)} \partial_{ij} v_{g,f} \right] \frac{\Delta t_k}{2} + \sum_{k=0}^{N-1} E \left[ f(t_k, X^\Delta(t_k)) - f(t_{k+1}, X^\Delta(t_{k+1})) \right] \frac{\Delta t_k}{2}
\]

where \( v_{g,f}^\Delta = v_{g,f}^\Delta(t_{k+1}, X^\Delta(t_{k+1})) \) and where equation (88) is conditioned on \( X^\Delta(0) = x \). Lemma 4.1 in [4] shows that
\[ \Delta_{g,f}^\Delta(x) - \Delta_{g,f}^\Delta(x) = O(\Delta_N^*) \] (89)

where \( \Delta_N^* \) is the largest time step on \([0, T]\).

6.2.2 Calculation of \( u \)

When calculating \( u = v \) in equation (83) set \( f = 0 \) and \( t = 0 \). This means that
\[ u(x) = u(0, x) = u^\Delta(x) + \Delta_{g,0}^\Delta(x) + R_d^\Delta \]

and similar to equation (88) \( \Delta_{g,0}^\Delta(x) \) becomes
\[ \Delta_{g,0}^\Delta(x) = \sum_{k=0}^{N-1} E \left[ \epsilon_i^{(k)} \phi_i(t_{k+1}) + \epsilon_{ij}^{(k)} \phi_{ij}(t_{k+1}) \right] \frac{\Delta t_k}{2} \] (91)

The last expression in (90) is \( R_d^\Delta = O(\Delta_N^*)^2 \). The estimator used when the standard MC method is applied is
\[ u^{\Delta,M}(x) = \frac{1}{M} \sum_{m=1}^{M} g(X^\Delta(T, \omega_m)) \] (92)

where \( \{\omega_m\}_{m=1}^{M} \) symbolizes the M realizations of the Euler-Maruyama approximation. When the multilevel MC method will be applied, the multilevel estimator in equation (49) will be used. For the Cauchy problem it will be written as
\[ u_{ml}^\Delta(x) = \sum_{l=0}^{L} u_l^\Delta \] (93)
where
\[ u_0^\Delta = \frac{1}{N_0} \sum_{i=1}^{N_0} g \left( X_0^{\Delta,i}(T) \right) \] (94)
whenever \( l = 0 \) and
\[ u_l^\Delta = \frac{1}{N_l} \sum_{i=1}^{N_l} \left[ g \left( X_l^{\Delta,i}(T) \right) - g \left( X_{l-1}^{\Delta,i}(T) \right) \right] \] (95)
whenever \( l > 0 \). \( g \left( X_l^{\Delta,i}(T) \right) \) is the value of the function \( g \) calculated using \( X_l^{\Delta,i}(T) \) where \( X_l^{\Delta,i}(T) \) come from the \( i \)th trajectory on a grid with \( M^l \) number of grid partitions.

The value for \( u(x) \) can be written as
\[ u_{ml}^\Delta(x) + \left( u(x) - u^\Delta(x) \right)_d + \left( u^\Delta(x) - u_{ml}^\Delta(x) \right)_{s,ml} \] (96)
in the case of the multilevel MC and
\[ u^{\Delta,M}(x) + \left( u(x) - u^\Delta(x) \right)_d + \left( u^\Delta(x) - u^{\Delta,M}(x) \right)_s \] (97)
in the case of standard MC. The expression
\[ \left( u(x) - u^\Delta(x) \right)_d \] (98)
is the discretization error \( E_d^\Delta(x) \) and
\[ \left( u^\Delta(x) - u_{ml}^\Delta(x) \right)_{s,ml} \] (99)
is the statistical error \( E_s^\Delta(x) \) for the combined estimator used in the multilevel MC and
\[ \left( u^\Delta(x) - u^{\Delta,M}(x) \right)_s \] (100)
is the statistical error \( E_s^{\Delta,M}(x) \) for the standard MC estimator. Now define, \( \forall k \in \{0, \ldots, N-1\} \) and \( m \in \{1, \ldots, M\} \)
\[ \rho_k(\omega_m) = \frac{\epsilon_i(\omega_m)\psi_i(t_{k+1},\omega_m) + \epsilon_{ij}(\omega_m)\psi_{ij}^{(1)}(t_{k+1},\omega_m)}{2\Delta t_k}. \] (101)
By using (91) and (97) the expression for the error \( E_d^\Delta(x) \) becomes
\[ E_d^{\Delta,M}(x) + E_{d,s}^{\Delta,M}(x) + R_d^\Delta. \] (102)
The error in (102) becomes
\[ E_d^{\Delta,M}(x) = \frac{1}{M} \sum_{k=0}^{N-1} \sum_{m=1}^{M} \rho_k(\omega_m)(\Delta t_k)^2, \] (103)
and
\[ E_{d,s}^M(x) = \sum_{k=0}^{N-1} E \left[ \rho_k(\omega_m) \right] (\Delta t_k)^2 - \frac{1}{M} \sum_{k=0}^{N-1} \sum_{m=1}^M \rho_k(\omega_m)(\Delta t_k)^2. \]  
(104)

The central limit theorem then implies
\[ E_{d,s}^M = \sum_{k=0}^{N-1} \int_{t_k}^{t_{k+1}} I_{k,M} dt \]  
(105)

and when \( M \to \infty \) \( \sqrt{M} I_{k,M} \) converges to a normally distributed random variable with mean value 0 and variance
\[ \sigma_k^2 = \text{Var} \left[ \epsilon_i(k) \psi_i(t_{k+1}) \right] + \text{Var} \left[ \epsilon_{ij}(1) \psi_{ij}(t_{k+1}) \right]. \]  
(106)

### 6.2.3 Discrete dual functions

Recall that
\[ u^\Delta(t_k, x) = E \left[ g(X^\Delta) \mid X^\Delta(t_k) = x \right]. \]  
(107)

Next the dual functions \( \psi_i(t_k) \) and \( \psi_i^{(n)}(t_k) \) associated to \( g, X^\Delta \) and \( u^\Delta \) will be presented. Only the dual functions of the first and second order will be needed in this example. It can be shown [13] that the dual functions for the first and second order becomes
\[ \partial_i u^\Delta(t_k, X^\Delta(t_k)) = E[\psi_i(t_k) | \mathcal{F}_{t_k}], \]  
\[ \partial_{ij} u^\Delta(t_k, X^\Delta(t_k)) = E[\psi_{ij}^{(1)}(t_k) | \mathcal{F}_{t_k}] \]  
(108)

for \( i, j \in \{1, \ldots, n\} \) and \( k \in \{0, \ldots, N\} \). The dual functions of the first \( \psi_i(t_k) \) and second \( \psi_{ij}^{(1)}(t_k) \) order can be calculated by means of certain backwards in time difference equations. Let
\[ c_i(t_k, x) = x_i + \mu_i(t_k, x) \Delta t_k + \sigma_{i\beta}(t_k, x) \Delta W_\beta(t_k) \]  
(109)

denote an Euler-Maruyama discretization scheme for \( i \in \{1, \ldots, n\}, k \in \{0, \ldots, N\} \) and \( x \in \mathbb{R} \). The first order dual function is then recursively defined as
\[ \psi_i(t_N) = \partial_i g(X^\Delta(t_N)) \]  
\[ \psi_i(t_k) = \partial_i c_\beta(t_k, X^\Delta(t_k)) \psi_\beta(t_{k+1}) \]  
(110)

and the second order as
\[ \psi_{ij}^{(1)}(t_N) = \partial_{ij} g(X^\Delta(t_N)) \]  
\[ \psi_{ij}^{(1)}(t_k) = \partial_{ij} c_\beta(t_k, X^\Delta(t_k)) \partial_j c_\gamma(t_k, X^\Delta(t_k)) \psi_{i\gamma}(t_{k+1}) \]  
\[ + \partial_{ij} c_\beta(t_k, X^\Delta(t_k)) \psi_\beta(t_{k+1}), \]  
(111)

both for \( i, j \in \{1, \ldots, n\} \) and \( k \in \{0, \ldots, N - 1\} \).
6.2.4 Statistical error

Consider a random variable $Y$ defined on the probability space $(\Omega, \mathcal{F}, P)$ and let $\{Y(\omega_m)\}_{m=1}^{M}, \omega_m \in \Omega$ denote $M$ independent samples of $Y$. Equation (112) shows the arithmetic mean value, variance and standard deviation respectively.

$$A(M,Y) = \frac{1}{M} \sum_{j=1}^{M} Y(\omega_j)$$  
$$S(M,Y) = (A(M,Y^2) - (A(M,Y))^2)^{1/2}$$  

(112)

If $Y(\omega_m) = g(X^\Delta(T,\omega_m))$ then for a sufficiently large sample the upper bound of the statistical error for the function $g(X^\Delta(T,\omega_m))$ becomes

$$E_s^{\Delta,M} \leq \frac{c_0}{\sqrt{M}} S(M', g(X^\Delta(T,\omega_m)))$$  

(113)

for some number of trajectories $M'$. Statistical error associated to the discretization error then becomes

$$E^{\Delta,M}_{d,s} \leq \frac{c_0}{\sqrt{M}} S \left( M, \sum_{k=0}^{N-1} \rho_k(\omega_m)(\Delta t_k)^2 \right)$$  

(114)

Equation (113) and (114) holds for the standard MC method. The statistical error $E^\Delta_s$ will be calculated using multilevel MC and $E^{\Delta,M}_s$ will be calculated using standard MC. The calculation of $E^{\Delta,s}_{d,s}$ on the other hand will use the standard MC in both cases. For the multilevel MC, the arithmetic mean uses equation (93). The variance of the combined estimator then becomes

$$\mathcal{V} = \sum_{l=0}^{L} N_l^{-1} \mathcal{V}_l \left( N_l, u^\Delta_l \right)$$  

(115)

where $\mathcal{V}_l$ is the variance of a single sample and $u^\Delta_l$ come from equation (93) and are described by equations (94) and (95). Equation (115) is derived from equation (50). The statistical error for the combined estimator then becomes

$$E^\Delta_s = c_0 \sqrt{\mathcal{V}}.$$  

(116)

6.3 Cauchy problem

The following one dimensional problem is used to test the multilevel MC method combined with the calculations of the discretization, discretization statistical and statistical errors. Let $t \in [0,T]$ and $\sigma(t) = \frac{1+t}{10}$. Let the stochastic process $X$ solve the SDE

$$dX(t) = X(t)dt + \sigma(t)dW(t).$$  

(117)
The corresponding differential operator is

\[ L = \frac{1}{2} (\sigma(t))^2 \partial_{11} + x \partial_1 \]  

(118)

and let

\[ u(t, x) = E[(X(T))^2 | X(t) = x]. \]  

(119)

\( u(t, x) \) solves the equation

\[
\begin{cases}
\partial_t u(t, x) + Lu(t, x) = 0, & \text{if } (t, x) \in [0, T] \times \mathbb{R} \\
u(T, x) = x^2, & x \in \mathbb{R}.
\end{cases}
\]  

(120)

By using the Itô calculus the solution to \( u(t, x) \) when \( t \in [0, T] \) and \( x = 1 \) is

\[ u(0, 1) = \frac{81}{80} e^2 - \frac{13}{400} \approx 7.44892. \]  

(121)

This result will later serve as a reference to the results achieved using the adaptive stepping method combined with multilevel MC. The Euler-Maruyama discretization scheme is

\[ X^\Delta(t_{k+1}) = X^\Delta(t_k)(1 + \Delta t_k) + \sigma(t) \Delta W(t_k) \]  

(122)

where \( \Delta t_k = t_{k+1} - t_k \) for \( k \in \{0, 1, \ldots, N - 1\} \) and with the initial condition \( X^\Delta(t_0) = 1 \). To use earlier notations note that (122) becomes

\[ c(t_k, x) = x(1 + \Delta t) + \sigma(t_k) \Delta W(t_k) \]  

(123)

for all \( x \in \mathbb{R}^n \). The first order derivative becomes

\[ \partial_1 c(t_k, x) = 1 + \Delta t_k \]  

(124)

with all higher order derivatives \( \frac{\partial^{(i)}}{\partial x^i}, i = 2, 3, 4 \) equal to zero. The next problem is to calculate the dual functions. Given that \( g(x) = x^2 \) the first and second dual functions, \( \psi \) and \( \psi^{(1)} \), can be calculated as follows,

\[
\begin{align*}
\psi(t_N) &= 2X^\Delta(t_N) \\
\psi(t_k) &= (1 + \Delta t_k) \psi(t_{k+1})
\end{align*}
\]  

(125)

and

\[
\begin{align*}
\psi^{(1)}(t_N) &= 2 \\
\psi^{(1)}(t_k) &= (1 + \Delta t_k)^2 \psi^{(1)}(t_{k+1}).
\end{align*}
\]  

(126)

As explained earlier,

\[ u(0, 1) = u^\Delta_{ml}(x) + E^\Delta_s + E^\Delta_d + E^{\Delta,s}_d + O(\Delta_N^2) \]  

(127)
where \(u^\Delta_{ml}(x)\) is the multilevel estimator given by equation (93). The discretization error is given by

\[
E^\Delta,M_d = \sum_{m=1}^{M} \sum_{k=0}^{N-1} \left[ (X^\Delta,M(t_{k+1}, \omega_m) - X^\Delta,M(t_k, \omega_m)) \psi(t_{k+1}, \omega_m) \right] \frac{\Delta t_k}{2M}
\]

\[+ \sum_{m=1}^{M} \sum_{k=0}^{N-1} \left[ (\sigma(t_{k+1}))^2 - (\sigma(t_k))^2 \right] \psi^{(1)}(t_{k+1}, \omega_m) \frac{\Delta t_k}{4M}. \tag{128}\]

The discretization error and discretization statistical error uses standard MC simulation and estimate with \(M\) trajectories. The discretization statistical error are given by equation (114) and the statistical error is calculated using equation (116).

### 6.4 Algorithm for the Cauchy problem

This section presents the algorithm used to solve the Cauchy problem. The multilevel MC will be combined with calculations for equations (114), (116) and (128). The multilevel MC algorithm is the same as in section 5.3.

1. use multilevel MC to calculate \(u(t, x)\) from equation (93). Use equation (116) to calculate the statistical error.

2. set \(M\) as an initial number of trajectories and set an error limit \(TOL_{d,s}\) for \(E^\Delta,M_d\). If \(E^\Delta,M_d > TOL_{d,s}\), increase \(M\). Continue until \(E^\Delta,M_d < TOL_{d,s}\).

3. calculate the discretization error corresponding to the finest grid using equation (128) and using \(M\) number of trajectories.

The algorithm for the Cauchy problem uses first the multilevel MC and then standard MC for the calculation of \(E^\Delta,M_{d,s}\) and \(E^\Delta,M_d\), just as described in previous sections. The algorithm is a mix between the algorithms described in [2] and [4].

### 7 Software Issues

#### 7.1 C/C++

All of the source code for the multilevel MC was written in C. Since quantitative teams use C/C++ extensively in their numerical work, C is a good choice of computer language. Also, there are useful libraries available in C/C++ such as QuantLib and GNU Scientific Library (GSL) [12]. Although QuantLib was not used in the source code, the GSL library was. The GNU Compiler, GCC, was used as well as GNU plot for plotting the graphs.
7.2 Random Number Generator

The random number generator used was the so called Mersenne Twister developed in 1997. The number generator gets its name from the fact that it is using a Mersenne prime period of $2^{19937} - 1$. A random number generator using the Mersenne Twister is available in the GSL library.

8 Results

Below, the results are presented for the European, Asian and Lookback call options using the Euler-Maruyama scheme and the Milstein scheme and also an example of the Cauchy problem outlined in earlier sections. In all of the cases $M = 2$ was used even though higher refinement numbers of $M$ can be used. All the examples for the option pricing uses $2 \times 10^6$ samples at each level $l$ for the plots of the mean value and variance. These plots are designed to illustrate the convergence rate of the mean value and variance to show that the correct convergence rate associated to each discretization scheme is fulfilled.

All of the option examples uses the SDE

$$dS = rSdt + \sigma SdW$$ (129)

which means that geometric Brownian motion were used. $r = 0.05$, $\sigma = 0.20$, $T = 1$, $S(0) = 1$ and $K = 1$ was used. More information about the parameters in the Cauchy problem can be found in the earlier description of the problem.

8.1 Euler-Maruyama scheme

Three different types of call options, two of them path dependent, are being presented. The simulations uses both the multilevel MC method and the standard MC method. A comparison of the two methods will be made together with other results relevant for the multilevel method.

8.1.1 European call option

Figure 1 shows the variance and mean value for a European call option. The variance and mean value are shown for increasing numbers of $l$ for both a standard MC ($P^{(i)}_l$) and the multilevel MC ($P^{(i)}_l - P^{(i-1)}_l$). The quantity, which is plotted is $\log_M[\text{mean}]$ for the mean value and $\log_M(\text{variance})$ for the variance, $M = 2$. The logarithm is plotted because a slope of $-1$ indicates a convergence rate of $O(h^\beta)$, $\beta = 1$, for $V[P_l - P_{l-1}]$.

Table 1 shows the price of the option for various values of $\varepsilon$ for both multilevel MC and standard MC. The standard MC uses the same number of step sizes seen in the last level in the multilevel MC simulation. The variance of the two
Figure 1: European option, Euler-Maruyama scheme. $2 \times 10^6$ sample paths used on each level.

Table 1: Table showing results from multilevel MC and standard MC for a European option. Note the relatively steep decrease in the value of $\varepsilon$ and the wide range in the number of step sizes used.

<table>
<thead>
<tr>
<th>$\varepsilon$</th>
<th>$2^L$</th>
<th>$\bar{Y}$</th>
<th>$V[\bar{Y}]$</th>
<th>$\bar{Y}_{mc}$</th>
<th>$V[\bar{Y}_{mc}]$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.001</td>
<td>4</td>
<td>0.103442</td>
<td>4.07602e-07</td>
<td>0.102647</td>
<td>5.91368e-07</td>
</tr>
<tr>
<td>0.0005</td>
<td>8</td>
<td>0.103858</td>
<td>1.14515e-07</td>
<td>0.105329</td>
<td>3.95426e-07</td>
</tr>
<tr>
<td>0.00025</td>
<td>16</td>
<td>0.104376</td>
<td>3.11846e-08</td>
<td>0.104226</td>
<td>1.82451e-07</td>
</tr>
<tr>
<td>0.00015</td>
<td>32</td>
<td>0.104375</td>
<td>1.12727e-08</td>
<td>0.104567</td>
<td>1.07132e-07</td>
</tr>
<tr>
<td>0.00005</td>
<td>32</td>
<td>0.104475</td>
<td>1.25040e-09</td>
<td>0.104293</td>
<td>1.17921e-08</td>
</tr>
<tr>
<td>0.000025</td>
<td>64</td>
<td>0.10446</td>
<td>3.13605e-10</td>
<td>0.104328</td>
<td>4.83073e-09</td>
</tr>
</tbody>
</table>

different estimators is also presented in the table. The cost of the calculation in the multilevel case and the standard MC case is the same. This is achieved by first making the calculation with the multilevel MC and then calculate the cost, $C_{ml}$, using equation (57). By using this cost and the step size on the finest level $L$ in the multilevel calculation, the appropriate number of trajectories for standard MC is calculated using

$$N_{mc} = C_{ml}/M^L,$$

where $M^L$ illustrates the number of time steps used at the last level $L$. Table 2 shows the quotient between the variance for standard MC and multilevel MC simulations for a European options for decreasing $\varepsilon$. As $\varepsilon$ decreases, the difference in variance between the multilevel MC method and the standard MC method increases meaning that the multilevel method becomes more attractive compared to the standard method as $\varepsilon \to 0$. For multilevel calculations with a high $\varepsilon$ the additional samples needed for each new level is quite low. When the optimal number of trajectories on each level, given by equation (61), is close to the initial number used to make a first estimation of $V_\\ell$ the gain given by the multilevel
Table 2: Table showing the quotient between the variance for standard MC and multilevel MC simulations for a European options for decreasing $\varepsilon$. Remark that the quotient is roughly equal when the same number of time steps are being used, even though the $\varepsilon$ is different.

<table>
<thead>
<tr>
<th>$\varepsilon$</th>
<th>$2^L$</th>
<th>$V[Y_{mc}] / V[\hat{Y}]$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.001</td>
<td>4</td>
<td>1.4508</td>
</tr>
<tr>
<td>0.0005</td>
<td>8</td>
<td>3.4530</td>
</tr>
<tr>
<td>0.00025</td>
<td>16</td>
<td>5.8507</td>
</tr>
<tr>
<td>0.00015</td>
<td>32</td>
<td>9.5037</td>
</tr>
<tr>
<td>0.00005</td>
<td>32</td>
<td>9.4307</td>
</tr>
<tr>
<td>0.000025</td>
<td>64</td>
<td>15.4039</td>
</tr>
</tbody>
</table>

algorithm is reduced. This is best seen when the number of optimal trajectories are close to the initial number used, especially for the lower levels.

This makes the cost $C_{ml}$, and then $N_{mc}$, relatively high, giving the standard MC calculation a variance close to, or sometimes even better than the multilevel MC. This behavior reduces as $\varepsilon \to 0$ and this means that the variance for the multilevel calculation in comparison with the variance for the standard MC computation improves when $\varepsilon$ is decreased. At the smallest $\varepsilon$, the variance for the multilevel calculation is considerably better then in the standard calculation. The relation $V[\hat{Y}_{mc}] / V[\hat{Y}]$, seen in the last column, shows an increasing trend as $\varepsilon \to 0$.

Figure 2: Number of additional samples for a European option using the Euler-Maruyama discretization scheme.

Figure 2 shows the additional samples proposed by equation (61) for different values on $l$ and for different values on $\varepsilon$. It shows a clear decreasing number of suggested additional samples needed which is expected since $V_l$ and $h_l$ decreases as $l$ increases. The inverse dependence of $\varepsilon^2$ gives higher number on $N_l$ for smaller
values on $\varepsilon$. Equation (61) tries to ensure that the variance of the combined estimator becomes less than $\varepsilon^2/2$. Equation (131) is used as a test to see if this is true,

$$q_\varepsilon = \frac{2V[\hat{Y}]}{\varepsilon^2}.$$  \hspace{1cm} (131)

When $q_\varepsilon < 1$, equation (61) is successful in keeping the variance of the combined estimator below $\varepsilon^2/2$. The results of $q_\varepsilon$ varies between 0.81520 and 1.00354 showing that equation (61) can not always keep the variance below $\varepsilon^2/2$ but that it is very close.

### 8.1.2 Asian call option

Figure 3 shows the mean value and variance for the Asian option and as in the case with the European option there is an approximate $O(h_l)$ convergence rate for the mean value. However, the slope for the variance increases when the level $l$ grows larger and when $l$ is approximately larger than 3 the convergence rate has reached around $-1.5$ showing a relatively high convergence rate. Even though the Euler-Maruyama is suppose to have a $O(h_l)$ convergence rate for the variance it is commonly known that it can achieve better for some specific cases and it might well be that this is one of those cases.

![Figure 3: Asian option, Euler-Maruyama scheme. 2 × 10^6 sample paths used on each level.](image)

In table 3 the results from multilevel MC simulations and standard MC simulations are presented showing a number of step sizes somewhat comparable to the European option calculations. This means that the algorithm stops at roughly the same level $L$. The Asian option does not give the same rate of quotients as the other two options, meaning that the difference of the variance between multilevel MC and standard MC is smaller than for European or Lookback options. This relation can be seen in Table 4. However, when the step size is 64 the quotient
Table 3: Table showing results from multilevel MC and standard MC for an Asian option.

<table>
<thead>
<tr>
<th>ε</th>
<th>$2^L$</th>
<th>$Y$</th>
<th>$V[Y]$</th>
<th>$Y_{mc}$</th>
<th>$V[Y_{mc}]$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.001</td>
<td>8</td>
<td>0.0567139</td>
<td>2.67605e-07</td>
<td>0.057118</td>
<td>2.10693e-07</td>
</tr>
<tr>
<td>0.0005</td>
<td>8</td>
<td>0.0570835</td>
<td>9.50039e-08</td>
<td>0.057604</td>
<td>1.23558e-07</td>
</tr>
<tr>
<td>0.00025</td>
<td>16</td>
<td>0.0572505</td>
<td>2.76729e-08</td>
<td>0.057332</td>
<td>5.64768e-08</td>
</tr>
<tr>
<td>0.00015</td>
<td>32</td>
<td>0.0575246</td>
<td>1.03408e-08</td>
<td>0.057806</td>
<td>3.4827e-08</td>
</tr>
<tr>
<td>0.00005</td>
<td>64</td>
<td>0.0576069</td>
<td>1.24938e-09</td>
<td>0.057554</td>
<td>6.62996e-09</td>
</tr>
<tr>
<td>0.000025</td>
<td>64</td>
<td>0.0576066</td>
<td>3.12548e-10</td>
<td>0.057511</td>
<td>1.65365e-09</td>
</tr>
</tbody>
</table>

Table 4: Table showing the quotient between the variance for standard MC and multilevel MC simulations for an Asian option for decreasing ε.

<table>
<thead>
<tr>
<th>ε</th>
<th>$2^L$</th>
<th>$V[Y_{mc}]/V[Y]$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.001</td>
<td>8</td>
<td>0.78733</td>
</tr>
<tr>
<td>0.0005</td>
<td>8</td>
<td>1.30056</td>
</tr>
<tr>
<td>0.00025</td>
<td>16</td>
<td>2.04087</td>
</tr>
<tr>
<td>0.00015</td>
<td>32</td>
<td>3.09568</td>
</tr>
<tr>
<td>0.00005</td>
<td>64</td>
<td>5.30660</td>
</tr>
<tr>
<td>0.000025</td>
<td>64</td>
<td>5.29087</td>
</tr>
</tbody>
</table>

is around 5 which is still good enough to motivate the usage of the multilevel algorithm.

Figure 4 shows the number of additional samples needed and the behavior of the optimal number of trajectories is similar to the corresponding values for the European option. The results from equation (131) shows that $0.53521 < q < 1.00015$ which means that the variance once again falls below or close to $\varepsilon^2/2$.

### 8.1.3 Lookback call option

The mean value and variance for the Lookback option shown in figure 5 shows the same behavior as both the European and Asian option even though it gets a slightly smaller value for the slope for both the mean, not reaching the −1 or better as the two earlier options managed. The variance achieves −1 as expected. A difference between the Lookback option and the European and Asian options is the number of time steps used in the final level. The number of time steps are significantly higher for the Lookback option, as are the number of trajectories. This increases the computational cost and the cost for the Lookback option is roughly ten times higher than for the other two options, independent of the on $\varepsilon$.

Table 5 shows the price of the option for various values of $\varepsilon$ for both multilevel MC and standard MC. Table 6 shows a clear increase in the quotient when $\varepsilon$ decreases. The steady increase in the quotient is due to the fact that the number
Figure 4: Number of additional samples for an Asian option using the Euler-Maruyama discretization scheme.

Figure 5: Lookback option, Euler-Maruyama scheme. $2 \times 10^6$ sample paths used on each level.

of time steps used for the last level is always increasing, giving it a clear difference in the quotient for each $\varepsilon$. When two simulations with two different values on $\varepsilon$ stops at the same grid size, the quotient is roughly the same. This can be seen in the Table 2 and 4 showing the quotients for the variances for European and Asian options. In the case of Lookback options, the usage of the multilevel algorithm gives significant improvements for the variance of the combined estimator.

Figure 6 shows the number of additional samples needed, showing a similar behavior as previous options. The results from equation (131) shows that $0.62674 < q_\varepsilon < 1.00223$, a result similar to the ones from the two previous options.
Figure 6: Number of additional samples for a Lookback option using the Euler-Maruyama discretization scheme.

<table>
<thead>
<tr>
<th>$\varepsilon$</th>
<th>$2^L$</th>
<th>$\bar{Y}$</th>
<th>$V[\bar{Y}]$</th>
<th>$\bar{Y}_{mc}$</th>
<th>$V[\bar{Y}_{mc}]$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.001</td>
<td>64</td>
<td>0.172615</td>
<td>3.13369e-07</td>
<td>0.172513</td>
<td>6.65588e-07</td>
</tr>
<tr>
<td>0.0005</td>
<td>128</td>
<td>0.172491</td>
<td>9.92586e-08</td>
<td>0.172717</td>
<td>5.40765e-07</td>
</tr>
<tr>
<td>0.00025</td>
<td>256</td>
<td>0.172373</td>
<td>2.87238e-08</td>
<td>0.172174</td>
<td>3.29626e-07</td>
</tr>
<tr>
<td>0.00015</td>
<td>512</td>
<td>0.172291</td>
<td>1.08755e-08</td>
<td>0.172098</td>
<td>2.15038e-07</td>
</tr>
<tr>
<td>0.00005</td>
<td>1024</td>
<td>0.172203</td>
<td>1.25097e-09</td>
<td>0.172461</td>
<td>4.17176e-08</td>
</tr>
<tr>
<td>0.000025</td>
<td>2048</td>
<td>0.172185</td>
<td>3.13198e-10</td>
<td>0.172304</td>
<td>1.73174e-08</td>
</tr>
</tbody>
</table>

Table 5: Table showing results from multilevel MC and standard MC for a Lookback option.

<table>
<thead>
<tr>
<th>$\varepsilon$</th>
<th>$2^L$</th>
<th>$V[\bar{Y}_{mc}]/V[\bar{Y}]$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.001</td>
<td>64</td>
<td>2.1240</td>
</tr>
<tr>
<td>0.0005</td>
<td>128</td>
<td>5.4480</td>
</tr>
<tr>
<td>0.00025</td>
<td>256</td>
<td>11.4757</td>
</tr>
<tr>
<td>0.00015</td>
<td>512</td>
<td>19.7727</td>
</tr>
<tr>
<td>0.00005</td>
<td>1024</td>
<td>33.3482</td>
</tr>
<tr>
<td>0.000025</td>
<td>2048</td>
<td>55.2922</td>
</tr>
</tbody>
</table>

Table 6: Table showing the quotient between the variance for standard MC and multilevel MC simulations for a Lookback option for decreasing values on $\varepsilon$. 
8.2 Milstein scheme

Below, the examples are presented using the same call options as previous, but this time using the Milstein discretization scheme. Similar comparisons and illustrations as previous examples.

8.2.1 European call option

![Figure 7: European option, Milstein scheme. 2 x 10^6 sample paths used on each level.](image)

Figure 9 shows the mean value and variance for the European option. As in the previous examples, which used the Euler-Maruyama discretization scheme, the value of the slope of the variance is an indicator of the convergence rate. The Milstein scheme has a $\beta$ which is equal to $-2$, and that is also approximately the value for the slope of the variance in Figure 7. The convergence of the slope of the mean value is approximately $-1$. This corresponds to a weak order of convergence of one meaning that the convergence rate achieves what could be expected.

Table 7 shows the quotient between the variance of the value from the standard MC simulation and the variance of the combined estimator given in the multilevel case. This table can be compared to Table 2 showing the same relation but for the Euler-Maruyama discretization scheme. The obvious notation is the significantly higher quotient achieved in the Milstein example compared to the Euler-Maruyama example. This means that the effect of using multilevel MC compared to standard MC increases rapidly when using the Milstein discretization instead of the Euler-Maruyama discretization. Still, both discretization schemes often achieves a better variance for a fixed computational cost than standard MC. Note also the large value of the quotient for the lowest value of $\varepsilon$.

Figure 8 shows the number of additional samples needed. When compared to Figure 2 it becomes clear that the multilevel MC that uses the Milstein scheme uses a significantly lower number of trajectories. This is also obvious when
Figure 8: Number of additional samples for a European option using the Milstein discretization scheme.

Table 7: Table showing the quotient between the variance for MC and multilevel MC simulations for European options for decreasing $\varepsilon$ when using the Milstein scheme.

<table>
<thead>
<tr>
<th>$\varepsilon$</th>
<th>$2^L$</th>
<th>$V[Y_{mc}]/V[Y]$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0001</td>
<td>16</td>
<td>2.9303</td>
</tr>
<tr>
<td>0.0005</td>
<td>32</td>
<td>9.6971</td>
</tr>
<tr>
<td>0.00025</td>
<td>64</td>
<td>28.6752</td>
</tr>
<tr>
<td>0.00015</td>
<td>128</td>
<td>64.8733</td>
</tr>
<tr>
<td>0.00005</td>
<td>265</td>
<td>188.6735</td>
</tr>
<tr>
<td>0.000025</td>
<td>1024</td>
<td>719.3832</td>
</tr>
</tbody>
</table>

comparing the computational time where the Milstein example is significantly faster than the Euler-Maruyama example. Finally, equation (131) shows that $0.92067 < q_e < 0.98890$ which means that $V[\hat{Y}] < \varepsilon^2/2$ for all values on $\varepsilon$.

8.2.2 Asian call option

As in the case of the European option using the Milstein scheme, a slope of $-2$ in the graph showing the variance is achieved. Also the mean value shows the same behavior as the one in Figure 7, at least for $L$ larger than one. This concludes that also the Asian option achieves this higher convergence rate for the variance associated with the Milstein scheme. However, the improvement from using the Milstein scheme is not as large as the theory predicts because of the rather high convergence rate in the Euler-Maruyama case for the Asian option. However, the biggest difference between the two discretization schemes is the number of trajectories used, especially for larger levels where the Milstein case uses a significantly lower number of trajectories.
Figure 9: Asian option, Milstein scheme. $2 \times 10^6$ sample paths used on each level.

Table 8: Table showing the quotient between the variance for MC and multilevel MC simulations for an Asian option for decreasing $\varepsilon$ when using the Milstein scheme.

<table>
<thead>
<tr>
<th>$\varepsilon$</th>
<th>$2^L$</th>
<th>$V[\hat{Y}_{mc}]/V[\hat{Y}]$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.001</td>
<td>4</td>
<td>0.54685</td>
</tr>
<tr>
<td>0.0005</td>
<td>4</td>
<td>1.66103</td>
</tr>
<tr>
<td>0.00025</td>
<td>16</td>
<td>5.51116</td>
</tr>
<tr>
<td>0.00015</td>
<td>32</td>
<td>12.61982</td>
</tr>
<tr>
<td>0.00005</td>
<td>128</td>
<td>67.59775</td>
</tr>
<tr>
<td>0.000025</td>
<td>256</td>
<td>159.92721</td>
</tr>
</tbody>
</table>

Table 8 shows a smaller quotient then that of the European option and even one case where the standard MC achieves a better variance than the multilevel MC. As noted earlier the efficiency of the multilevel MC compared to standard MC increases as $\varepsilon \to 0$ and for the Asian option it ends up with a quite large advantage for the two lowest values of $\varepsilon$.

Figure 10 shows the number of additional samples needed. As mentioned earlier, note the reduction in trajectories needed compared to the Asian option that uses the Euler-Maruyama scheme. Equation (131) shows that $0.87929 < q_e < 0.98470$ again showing that $V[\hat{Y}] < \varepsilon^2/2$ for all values on $\varepsilon$.

8.2.3 Lookback call option

Figure 11 shows the mean value and variance for the Lookback option and not the mean value nor the variance reaches the $O(h_l)$ and $O(h_l^2)$ convergence expected even though they come close. The mean value needs two grid refinements before reaching the $O(h_l)$ convergence rate.

Table 9 shows a behavior similar to that of the two other options with an increasing efficiency for lower values of $\varepsilon$. 

39
Figure 10: Number of additional samples for an Asian option using the Milstein discretization scheme.

(a) Mean for Lookback option.  
(b) Variance for Lookback option.

Figure 11: Lookback option, Milstein scheme. $2 \times 10^6$ sample paths used on each level.

Figure 12 shows the number of additional samples needed. It follows the same behavior as the ones for the European and Asian options. Again, the number of trajectories used is decreased compared to the Euler-Maruyama scheme. Equation (131) shows that $0.85524 < q_e < 0.98905$. This means that the Milstein scheme manages to ensure that $V[\hat{Y}] < \varepsilon^2/2$ for all three options, unlike the Euler-Maruyama scheme.

### 8.3 Cauchy problem

An example is presented here where the Cauchy problem, equation (120), is solved using both multilevel MC and standard MC. In both cases the statistical and discretization errors associated with the calculations are calculated and the idea is to use these errors to determine where an increased computational budget
Table 9: Table showing the quotient between the variance for MC and multilevel MC simulations for Lookback options for decreasing $\varepsilon$ when using the Milstein scheme.

<table>
<thead>
<tr>
<th>$\varepsilon$</th>
<th>$2^L$</th>
<th>$V[Y_{mc}] / V[Y]$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.001</td>
<td>4</td>
<td>1.3765</td>
</tr>
<tr>
<td>0.0005</td>
<td>32</td>
<td>5.6646</td>
</tr>
<tr>
<td>0.00025</td>
<td>64</td>
<td>16.1293</td>
</tr>
<tr>
<td>0.00015</td>
<td>128</td>
<td>35.8152</td>
</tr>
<tr>
<td>0.00005</td>
<td>256</td>
<td>105.7842</td>
</tr>
<tr>
<td>0.000025</td>
<td>512</td>
<td>220.2036</td>
</tr>
</tbody>
</table>

Figure 12: Number of additional samples for a Lookback option using the Milstein discretization scheme.

would be most efficiently spent. Together with the multilevel MC, this creates a powerful computational tool achieving a low statistical error considering the spent computational cost together with a good control in the different errors. Algorithm 6.4 is used when solving with multilevel MC.

Equations (93)-(95) and (115)-(116) are all a consequence of the use of multilevel MC and the theory presented in section 5. These equations are used for the modification of the simulations when multilevel MC is used instead of standard MC. It is also these modifications that separates the multilevel MC simulations from the standard MC used in [4].

Table 10 shows the results from multilevel MC simulations for various values on $\varepsilon$ along with the discretization error, statistical error associated with the calculation of the discretization error and also the statistical error associated with the combined estimator. From this table it is easy to see where an improvement is most efficiently spent and it is possible to determine if the additional computer cost should be spent on a finer grid, where a lower discretization error could be achieved, or if a larger set of trajectories should be used in order to decrease the
\begin{table}[h]
\centering
\begin{tabular}{|c|c|c|c|c|c|}
\hline
$\varepsilon$ & $2^L$ & $u^\Delta$ & $E_{d}^{\Delta,M}$ & $E_{d,s}^{\Delta,M}$ & $E_{s}^{\Delta}$ \\
\hline
0.05 & 256 & 7.423347 & 0.02884580 & 0.00041625 & 0.01641830 \\
0.025 & 512 & 7.437778 & 0.0147577 & 0.00020559 & 0.01641840 \\
0.015 & 1024 & 7.439959 & 0.00738702 & 0.00010357 & 0.01281110 \\
0.01 & 2048 & 7.442482 & 0.00369851 & 5.27079e-05 & 0.00927569 \\
0.005 & 4096 & 7.44376 & 0.00182817 & 2.59908e-05 & 0.00559610 \\
0.0025 & 8192 & 7.447451 & 0.00091219 & 1.30007e-05 & 0.00307046 \\
0.001 & 16384 & 7.448162 & 0.00045258 & 6.51591e-06 & 0.00130617 \\
0.0005 & 32768 & 7.449084 & 0.00023281 & 3.26106e-06 & 0.00066765 \\
\hline
\end{tabular}
\caption{Value and error estimate for the Cauchy problem for the multilevel MC case.}
\end{table}

\begin{table}[h]
\centering
\begin{tabular}{|c|c|c|}
\hline
$\varepsilon$ & $2^L$ & $E_{s}^{\Delta} / E_{d}^{\Delta,M}$ \\
\hline
0.05 & 256 & 0.56917 \\
0.025 & 512 & 1.11253 \\
0.015 & 1024 & 1.73427 \\
0.01 & 2048 & 2.50795 \\
0.005 & 4096 & 3.06104 \\
0.0025 & 8192 & 3.36603 \\
0.001 & 16384 & 2.88607 \\
0.0005 & 32768 & 2.86786 \\
\hline
\end{tabular}
\caption{Quotient of the two largest error estimates from the multilevel MC simulations for the Cauchy problem.}
\end{table}

statistical error. The values on $\varepsilon$ has been chosen so that each $\varepsilon$ corresponds to a finest level $L$ giving a particular grid size. The final value of the multilevel simulations clearly converges to the correct value of roughly 7.44892 presented in equation (121).

Table 11 shows the quotient between $E_{s}^{\Delta}$ and $E_{d}^{\Delta,M}$ presented earlier in Table 10. These errors are the two largest and that is why these are the most effective to improve in order to improve the overall reduction in calculation bias. The multilevel MC simulation of which these error estimates are based on uses the same algorithm as in previous examples, meaning that it uses the same formula for additional trajectories on each level and the same test for convergence. The error estimates added in this example simply shows where an additional computational budget would be most efficiently spent. Whichever error is the largest varies and is sometimes the discretization error and sometimes the statistical error. For $\varepsilon = 0.025$ and $\varepsilon = 0.015$ the errors are quite evenly spread but as the value for $\varepsilon$ decreases the statistical error becomes larger compared to the discretization error meaning that an increased computational budget would be best spent on an increased number of trajectories instead of an decreased step size.
Table 12 shows the results from simulations using standard MC for the same values of $\varepsilon$ as in the multilevel simulations. This table acts as a reference to show the improvements given by the multilevel MC over standard MC. As in previous examples calculations using multilevel MC and standard MC uses the same computational cost, where the calculations of the computational cost has been presented earlier. The number of trajectories used in the standard MC simulations, called here $M_0$, are roughly the same for all values of $\varepsilon$, around $3 \times 10^4$. Because the number of trajectories is roughly the same for each value on $\varepsilon$, the values of $E_{s,x}^{\Delta,M'}$ are more or less the same for all simulations. The main difference lies in the number of time steps in each simulation. The error estimate for each grid size is given in Table 10. Note also that the value, $u^{\Delta,M'}$, given by the standard MC simulations vary a lot more than in the multilevel simulation and the accuracy compared to the real value is clearly poorer than that of the multilevel simulations.

Figure 13 shows the mean value and variance for simulation with the multilevel MC and standard MC. The slope of the mean approaches a value of $-1$ which is comparable to results achieved in the option pricing examples using the Euler-Maruyama discretization scheme. The slope for the variance however, shows a value approaching roughly $-2$, giving it an excellent convergence rate considering its use of the Euler-Maruyama scheme. With this convergence rate there is no reason to choose a Milstein discretization scheme instead of the Euler-Maruyama scheme for this example.

Figure 14 shows the optimal number of trajectories for each level. The behavior is the same for all values on $\varepsilon$ and all simulations have a very low optimal number of trajectories on the finest level of the simulation.

<table>
<thead>
<tr>
<th>$\varepsilon$</th>
<th>$2^L$</th>
<th>$u^{\Delta,M'}$</th>
<th>$E_{s,x}^{\Delta,M'}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.05</td>
<td>256</td>
<td>7.421666</td>
<td>0.01496410</td>
</tr>
<tr>
<td>0.025</td>
<td>512</td>
<td>7.435840</td>
<td>0.01499200</td>
</tr>
<tr>
<td>0.015</td>
<td>1024</td>
<td>7.453010</td>
<td>0.01515130</td>
</tr>
<tr>
<td>0.01</td>
<td>2048</td>
<td>7.436608</td>
<td>0.01511520</td>
</tr>
<tr>
<td>0.005</td>
<td>4096</td>
<td>7.436310</td>
<td>0.01501860</td>
</tr>
<tr>
<td>0.0025</td>
<td>8192</td>
<td>7.432457</td>
<td>0.01502180</td>
</tr>
<tr>
<td>0.001</td>
<td>16384</td>
<td>7.456470</td>
<td>0.01493830</td>
</tr>
<tr>
<td>0.0005</td>
<td>32768</td>
<td>7.461537</td>
<td>0.01474200</td>
</tr>
</tbody>
</table>

Table 12: \textit{Value and error estimate for the standard MC case for the Cauchy problem.}
9 Conclusions

The above option pricing examples have shown that the multilevel MC method, with both the Euler-Maruyama and Milstein discretization schemes, achieves better variance for a fixed computational cost than a corresponding standard MC method. However, even though the complexity theorem of the multilevel MC provides a proof that the MSE will be below $\varepsilon^2$, this is not always the case. The examples are the same as those in [2] and [3] and they are simply used to show the efficiency of the multilevel MC method. The new part introduced in this dissertation is the method used on the Cauchy problem.

The main idea with the Cauchy problem is to introduce method of combining the tools given in the section about adaptive weak approximation of stochastic differential equations and the multilevel MC method. The problem shows that it is possible to use both the gain in computational efficiency from the multilevel method.
method and the control of errors given by the adaptive weak approximation. As seen in the Cauchy problem, this knowledge about the errors will give the user the ability to determine where to spend an increased computational budget as efficiently as possible.

The quotients presented in Table 11 show that there are cases where the quotient between the statistical error and the discretization error clearly indicates an uneven relation between the two errors for certain values on $\varepsilon$. The maximum quotient is roughly 3.4, achieved at $\varepsilon = 0.0025$, meaning that the statistical error is roughly 3.4 times larger than the discretization error. Table 11 also shows that even though it is the same problem, just with different numbers of $\varepsilon$, it is not always the same type of error that is dominating.

In the Cauchy problem, the statistical error for the standard MC, shown in table 12, is quite high compared to the statistical error for the multilevel MC. This is because of the fact that both methods use the same computational cost and since the grid sizes are high, the standard MC can not compute as many trajectories as the multilevel MC method. Since the multilevel MC computes most of its trajectories on the coarser grids, the number of trajectories it can compute for the fixed computational cost becomes high. Both methods achieve the same discretization error since the multilevel MC makes calculation on the finest grid as well.

The question regarding choice of discretization scheme does not have a clear answer. When taking a quick look at the results, one may argue that the Milstein scheme should be the obvious choice due to its higher convergence rate. However, to make sure that the convergence rate of $O(h^2)$ is being achieved, the problem sometimes needs to be a bit remade as seen in the earlier examples using the Milstein scheme. Also, the Euler-Maruyama scheme sometimes achieves a convergence rate higher than the $O(h)$ which can be predicted. This higher convergence rate can be seen in the case of the Asian call option and the Cauchy problem. In the latter case, the convergence rate was even as high as $O(h^2)$.

Drawbacks to the multilevel MC method would be that it is not easy to control the number of trajectories since they are being calculated as the algorithm iterates. This can be a problem for cases with a high computational cost per iteration because of the extremely high computational cost that could be demanded using equation (61).

It should be mentioned that the number of grid refinement $M$ was set equal to two for all examples. This number can however be larger then that, and in fact, it can be shown that for a multilevel MC simulation, a value of $M = 7$ actually gives twice the computational efficiency compared to $M = 2$. $M = 2$ was used in order to make plots that are illustrative and that shows the behavior of the multilevel algorithm.

The example with the Cauchy problem shows promising results and illustrates how two advanced methods can be combined in order to make efficient simulations and to have the ability to determine different errors in order to spend an increased
computational budget as efficiently as possible.
References


[12] GSL Library Documentation