Importance sampling for Monte Carlo simulations with applications to finance

Shawn Ban

April 19, 2008

1 Introduction

In this paper we examine the use of importance sampling algorithms for Monte Carlo simulations in pricing financial derivatives. We first briefly discuss the use of Monte Carlo simulations in finance and how importance sampling can reduce the variance of the estimator used. In Section 2 we construct a Brownian motion and motivate the use of the Ito calculus. In Section 3, we use geometric Brownian motion to model the behavior of an underlying asset and show how to price a derivative on that asset. In Section 4, we show that Girsanov's theorem yields the risk-neutral measure upone which the prices of derivatives are calculuated. In Section 5, we discuss the asymptotically optimal change of measure for importance sampling based on the theory of large deviations. Finally, numerical experiments and comparisons are carried out in Section 6. For illustration, all the derivatives in the numerical examples admit explicit price formulae.

For this paper we assume familiarity with measure-theory probability. For simplicity, we will typically consider only the one-dimensional case, although the analysis can be extended to multi-dimensional settings.

1.1 Monte Carlo simulations

Monte Carlo method is a commonly used simulation algorithm to study the behavior of various physical and mathematical systems. It differs from other numerical methods by being stochastic in nature, usually by using pseudo-random numbers as opposed to deterministic algorithms.

In the field of mathematical finance, the problem of finding the arbitrage-free price of a derivative often comes down to the computation of an integral. In some cases these integrals can be valued analytically, and in other cases they can be valued using numerical integration or solving a partial differential equation, for example the Black-Scholes PDE. However when the number of dimensions (or degrees of freedom) in the problem is large, PDEs and numerical integrals become intractable. In these cases Monte Carlo methods often give better results. For large dimensional integrals, Monte Carlo methods converge to the solution more quickly than numerical integration methods, require less memory and are easier to program. In many cases Monte Carlo methods in fact often offer the only solution.

Monte Carlo methods were introduced to finance in 1977 by Phelim Boyle in his paper "Options: A Monte Carlo Approach" in the *Journal of Financial Economics* and are currently widely used in industry.

1.2 An example of Monte Carlo simulation

Suppose we want to estimate $\theta = E[f(X)]$, where X is a random variable and $f(x)$ is an arbitrary function. We then generate N independent and identically distributed (iid) copies of X, i.e. $X_1, X_2, ..., X_N$. A Monte Carlo estimator is just

$$\hat{\theta} = \frac{1}{N}\sum_{i=1}^{N} f(X_i).$$

Note that the following are true for $\hat{\theta}$:

1. $\hat{\theta}$ is a random variable.

2. $\hat{\theta}$ is an unbiased estimator.

3. The variance of $\hat{\theta}$ decay with rate $1/N$, since

$$\mathrm{Var}(\hat{\theta}) = \frac{1}{N^2}\mathrm{Var}(\sum_{i=1}^{N} f(X_i)) = \frac{1}{N^2}\sum_{i=1}^{N}\mathrm{Var}(f(X_i)) = \frac{1}{N}\mathrm{Var}(f(X)).$$

4. By the Central Limit Theorem,

$$\frac{1}{\sqrt{N}}\sum_{i=1}^{N}[f(X_i) - \theta] \approx N(0, \mathrm{Var}(f(X)))$$

   as $N \to \infty$.

5. $\hat{\theta}$ is approximately normal, that is, $\hat{\theta} \approx N(\theta, \mathrm{Var}(\hat{\theta}))$.

1.3 Importance sampling

Importance sampling (IS) is a variance reduction technique used in Monte Carlo simulations. Certain values of the input random variables in a simulation have more impact on the parameter being estimated than others. If these important values are emphasized by being sampled more frequently, then the estimator variance can be reduced. Hence, the basic method in importance sampling is to choose an alternative simulation distribution through a change of probability measure which encourages the important values. This use of biased distributions will result in a biased estimator if it is applied directly in the simulation. However, the simulation outputs are weighted to correct for the use of the biased distribution, and this ensures that the new IS estimator is unbiased. The weight is given by the likelihood ratio, that is, the Radon-Nikodym derivative of the true underlying distribution with respect to the biased simulation distribution.

In this paper we will refer to Monte Carlo simulation without importance sampling as naive Monte Carlo simulation.

1.4 Principles of importance sampling

Assume X is a random variable with probability density $h(x)$. We want to estimate

$$\theta = E[f(x)] = \int f(x)h(x)dx.$$

Now assume $\bar{X}$ is a random variable with probability density $g(x)$, such that $g(x) = 0$ whenever $h(x) = 0$. Then

$$\theta = \int f(x)\frac{h(x)}{g(x)}g(x)dx$$

We can write:

$$\theta = \int \bar{f}(x)g(x)dx \quad \text{where } \bar{f}(x) = f(x)\frac{h(x)}{g(x)}.$$

In other words,

$$\theta = E\left[f(\bar{X})\frac{h(\bar{X})}{g(\bar{X})}\right] = E[\bar{f}(\bar{X})].$$

The weight $h(x)/g(x)$ is called the likelihood ratio, or Radon-Nikodym derivative.

The implementation of IS algorithm will be as follows:

1. Generate N iid samples of $\bar{X}$ from $g(x)$, i.e. $\bar{X}_1, \bar{X}_2, ..., \bar{X}_N$.
2. Use the estimator

$$\hat{\theta} = \frac{1}{N}\sum_{i=1}^{N} f(\bar{X}_i)\frac{h(\bar{X}_i)}{g(\bar{X}_i)}$$

Note that $\hat{\theta}$ is a random variable and an unbiased estimator of θ. The variance of $\hat{\theta}$ is given by:

$$\mathrm{Var}(\hat{\theta}) = \frac{1}{N}\mathrm{Var}\left[f(\bar{X})\frac{h(\bar{X})}{g(\bar{X})}\right] = \frac{1}{N}\mathrm{Var}[\bar{f}(\bar{X})].$$

By definition of variance,

$$\mathrm{Var}[\bar{f}(\bar{X})] = E[\bar{f}^2(\bar{X})] - (E[\bar{f}(\bar{X})])^2 = E[\bar{f}^2(\bar{X})] - \theta^2.$$

Therefore, given N, in order to minimize the variance of $\hat{\theta}$, it suffices to find the distribution $g(x)$ that minimizes the second moment $E[\bar{f}^2(\bar{X})]$.

However, this minimization problem is not very meaningful if we allow $g(x)$ to be any density function. Indeed, by Jensen's Inequality

$$E[\bar{f}^2(\bar{X})] \geq E^2[\bar{f}(\bar{X})] = \theta^2,$$

with equality if

$$\bar{f}(X) = f(X)\frac{h(X)}{g(X)}$$

is a constant. It follows that if we pick

$$g(x) \doteq \frac{f(x)h(x)}{c}$$

where c is a constant, then $\mathrm{Var}[\bar{f}(\bar{X})] = 0$. Since g is a density, it is easy to see that

$$c = \int f(x)h(x)dx = E[f(X)] = \theta.$$

But of course θ is precisely what we are trying to estimate and unknown. It is for this reason that in practice the change of measure is often selecte from a family of parametrized probability distributions.

In the rest of this paper we will develop a framework to make judicious choices of change of measure, which is the key issue in the implementation of IS. The intuition is to encourage the important regions of the input variables. A good choice can boost the efficiency enormously, while a poor one can make the algorithm less efficient than a naive Monte Carlo simulation.

2 Brownian motion and Ito integral

Brownian motion is a continuous stochastic process $W = \{W(t) : t \geq 0\}$ that has independent and stationary, normally distributed increments. In Section 3, we will use geometric Brownian motion to model the behavior of a stock price. It is therefore worthwhile to examine Brownian motion in detail and discuss the related stochastic calculus. Note that the probability space $(\Omega, \mathcal{F}, \mathbb{P})$ is given and wil not be explicitly mentioned.

2.1 Symmetric random walk

Brownian motion can be understood as the limit of symmetric simple random walk. Let

$$M_n = \sum_{k=1}^{n} X_k, n = 1, 2, ...$$

where $M_0 = 0$ and $\{X_i\}$ are iid with $P(X_i = \pm 1) = 1/2$. At each step, it either increases or decreases by one unit, with each outcome having equal probability. Note that the symmetric simple random walk has independent and stationary increment. Moreover,

$$\mathrm{Var}[M_i - M_j] = |i - j|$$

for all $i, j = 0, 1, 2, \ldots$.

2.2 Scaled symmetric random walk

To approximate a Brownian motion, we speed up time and scale down the step size of a symmetric random walk. We do this by fixing a positive integer n and defining the scaled symmetric random walk

$$W^{(n)}(t) = \frac{1}{\sqrt{n}} M_{[nt]}$$

where $[nt]$ is the integer part of nt. Analogous to the symmetric random walk M, the scaled process $W^{(n)}$ has independent and stationary increments. Furthermore, for any $0 \leq s \leq t$,

$$W^{(n)}(t) - W^{(n)}(s) = \frac{1}{\sqrt{n}}(M_{[nt]} - M_{[ns]}) = \frac{1}{\sqrt{n}} \sum_{k=[ns]+1}^{[nt]} X_k,$$

which is the summation of iid random variable with mean 0 and variance 1. It follows from the Central Limit Theorem that, as $n \to \infty$,

$$W^{(n)}(t) - W^{(n)}(s) \approx N(0, t-s).$$

Therefore, at least formally, let $n \to \infty$ and denote by W the limit of process $W^{(n)}$, we would expect that W satisfies the following properties:

(a) $\mathbb{P}(W(0) = 0) = 1$

(b) For all $0 \leq s \leq t$, the increment $W(t) - W(s)$ follows the normal distribution with mean 0 and variance $t - s$.

(c) The process W has independent increments.

(d) The process W has continuous sample paths.

Any process that satisfies (a)-(d) is said to be a *standard Brownian motion*. The rigorous derivation of the limit process W being a standard Brownian motion is not trivial and beyond the scope of this thesis. It can be found in a number of probability texbooks [**?**] We collect below a few facts about Brownian motion.

Martingale property. W is a martingale, i.e.

$$E[W(t+s)|W(u) : 0 \leq u \leq s] = W(t).$$

Total variation. The total variation of a function f over $[0, T]$ is defined as

$$\sup \sum_{k=1}^{n} |f(t_k) - f(t_{k-1})|$$

where the supremum is taken over all partitions $0 = t_0 \leq t_1 \leq \cdots \leq t_{n-1} \leq t_n = T$. With probability one, the sample paths of Brownian motion have infinite total variation over any time interval of positive length.

Quadratic variation. Brownian motion has finite quadratic variation. Let $S = [t_0, t_1, ..., t_n]$ be a partition of $[0, T]$, i.e. $0 = t_0 \leq t_1 \leq \cdots \leq t_n = T$. The *mesh* of the partition is defined to be $\|S\| = \max_k \{t_{k+1} - t_k\}$. The *quadratic variation* of a function f on an interval $[0, T]$ is

$$\langle f \rangle_T = \lim_{\|S\| \to 0} \sum_{k=1}^{n} |f(t_k) - f(t_{k-1})|^2.$$

For Brownian motion, $\langle W \rangle_T = T$.

2.3 The Ito integral

Let $\mathcal{F} = \{\mathcal{F}_t\}$ be the filtration generated by Brownian motion up to time t, and $X = \{X(t)\}$ a stochastic process adapted to $\mathcal{F}$. Our goal is to define the intergal of X against the standard Brownian motion W

$$I(t) \doteq \int_0^t X(s)dW(s),$$

for all $0 \leq t \leq T$. In the setup of mathematical finance, one can understand the stochastic integral in the following way. Consider that $X(t)$ is the position we take in an asset at time t and $W(t)$ is the price of the asset at time t. We require X to be adapted, since to model financial decisions we can use all the available information up to time t, but cannot look into the future. The stochastic integral then denotes the total net income of the investment strategy X.

The difficulty we face is that Brownian motion paths have unbounded total variation. Therefore, one cannot define the integral in the classical Riemann-Stietjles sense. We will briefly describe the construction of Ito integral. Consider a partition $S = [t_0, t_1, \ldots, t_n]$ with $0 = t_0 \leq t_1 \leq \cdots \leq t_{n-1} \leq t_n = T$. For every $0 \leq t \leq T$, the approximation of the integral on this partition is defined as

$$J(S;t) \doteq \sum_{k=0}^{m-1} X(t_k)[W(t_{k+1}) - W(t_k)] + X(t_m)[W(t) - W(t_m)],$$

where $t_m \leq t < t_{m+1}$. Note that, unlike the definition of Riemann-Stietjles integral, if one replaces $X(t_k)$ by $X(\xi_k)$ where ξ_k is an arbitrary point on interval $[t_k, t_{k+1})$, the summation will *not* converge. We will give more concrete computations on this detail in Section 2.8.

Note that with this definition, $\{J(S;t) : 0 \leq t \leq T\}$ form a martingale. This is easy to see since

$$E[X(t_k)(W(t) - W(t_k))|\mathcal{F}_{t_k}] = X(t_k)E[W(t) - W(t_k)|\mathcal{F}_{t_k}] = 0.$$

The question now is to argue that $J(S;t)$, as $\|S\| \to 0$, has a well defined limit. This can be done rigorously [**?**] with mild conditions on X, and the stochastic integral is just

$$I(t) \doteq \lim_{\|S\| \to 0} J(S;t).$$

2.4 Properties of the Ito integral

From now we will always assume that X is adapted to the filtration $\mathcal{F}$ and

$$E \int_0^T X^2(t)dt < \infty.$$

For those X, the stochastic integral is well defined and has the following properties.

1. Continuity. The paths of $I(t) : 0 \leq t \leq T\}$ are continuous.
2. Adaptivity. For every t, $I(t)$ is $\mathcal{F}_t$-measurable.
3. Linearity. For any constants a and b,
$$\int_0^T [aX(t) + bY(t)]\, dW(t) = a \int_0^T X(t)\, dW(t) + b \int_0^T Y(t)\, dW(t).$$
4. Martingale. The process $\{I(t) : 0 \leq t \leq T\}$ is a martingale with expected value 0.
5. Ito isometry. For every t,
$$E[I^2(t)] = E \int_0^t X^2(s)ds.$$
6. Quadratic variation. The quadratic variation of the process I is
$$\langle I \rangle_t = \int_0^t X^2(s)ds.$$

2.5 Ito's formula

Ito's formula provides a rule to differentiate expressions of the form $f(W(t))$, where $f(x)$ is a twice continuously differentiable function and $W(t)$ is a Brownian motion. In integral form, Ito's formula is

$$f(W(t)) - f(W(0)) = \int_0^t f'(W(s))dW(s) + \frac{1}{2} \int_0^t f''(W(s))ds$$

Note that the two terms on the right hand side are well-defined. The first is an Ito integral, as previously defined. The second is just an ordinary integral with respect to time. It is often convenient to express Ito's formula in the differential form:

$$df(W(t)) = f'(W(t))dW(t) + \frac{1}{2}f''(W(t))dt.$$

If f is a function of both t and x for which the partial derivatives $f_t(t,x)$, $f_x(t,x)$ and $f_{xx}(t,x)$ are defined and continuous, Ito's formula in integral form is given by

$$\begin{aligned} f(T, W(T)) &= f(0, W(0)) + \int_0^T f_t(t, W(t))dt \\ &+ \int_0^T f_x(t, W(t))dW(t) + \frac{1}{2}\int_0^T f_{xx}(t, W(t))dt. \end{aligned}$$

Again it is convenient to consider the differential form:

$$df(t, W(t)) = f_t(t, W(t))dt + f_x(t, W(t))dW(t) + \frac{1}{2}f_{xx}(t, W(t))dt.$$

Remark: Compared with the classical calculus, Ito formula presents an extra term

$$\frac{1}{2}\int_0^T f''(W(t))\, dt.$$

This extra term actually comes from the nonzero quadratic variation of Brownian motion and preserves the martingale property of the Ito integral.

2.6 Changing the endpoints

Recall that stochastic integral I is defined as the limit of the summation $J(S;t)$. We mention that in the summation the value of X is always evaluated at the left-end-point t_k. In this section, we give an example to show that changing the point at which X is evaluated gives a different limit. It is clear from this illustration that the classical Riemann-Stietjles integration is unsuitable for evaluating the integral of a Brownian motion.

Let's consider the case where $X(t) = W(t)$ and $t_k = kT/n$. Consider the following scenarios:

$$
\begin{aligned}
&(1) \quad \sum_{k=0}^{n-1} W(t_k)\left[W(t_{k+1})-W(t_k)\right]; \\
&(2) \quad \sum_{k=0}^{n-1} W(t_{k+1})\left[W(t_{k+1})-W(t_k)\right]; \\
&(3) \quad \sum_{k=0}^{n-1} W((t_k+t_{k+1})/2)\left[W(t_{k+1})-W(t_k)\right].
\end{aligned}
$$

Here we use the the left endpoint, right endpoint and midpoint respectively to evaluate the integral. Recall that if W had sample paths with bounded total variation, then all the approximations would have the same limit as $n \to \infty$.

In this case, however, note what happens when we subtract the first approximation from the second, (2) − (1):

$$
(2)-(1)=\sum_{k=0}^{n-1}\left[W(t_{k+1})-W(t_k)\right]^2=\frac{1}{n}\sum_{k=0}^{n-1}\left[\sqrt{n}(W(t_{k+1})-W(t_k))\right]^2.
$$

Recall that all the increments $\sqrt{n}(W(t_{k+1})-W(t_k))$ are independent and identically normally distributed with mean 0 and variance T. Therefore, as $n \to \infty$, the above summation, by Law of Large Numbers, converges to the expected value of the square of a $N(0,T)$ random variable, which is T. This demonstrates that using the backward difference instead of the forward difference gives a different answer for the integral.

The third approximation (3) is called the Stratonovich integral. One can verify that the difference between the Ito integral and the Stratonovich integral is $T/2$. As an interesting sidenote, the differentiation rule for Stratonovich integral behaves just like the one for classical calculus.

2.7 Using Ito's formula - a worked example

Ito's formula often simplifies the computation of Ito integrals. We'll now compute

$$
I(t)=\int_0^T W(t)\,dW(t)
$$

by both definition and the Ito's formula, and verify that they achieve the same result.

By definition:
Let $t_k = kT/n$ and to ease notation, we use W_k to denote $W(t_k)$. Note that

$$\sum_{k=0}^{n-1} W_k(W_{k+1} - W_k) + \frac{1}{2}\sum_{k=0}^{n-1}(W_{k+1} - W_k)^2 = \frac{1}{2}\sum_{k=0}^{n-1}(W_{k+1}^2 - W_k^2) = \frac{1}{2}W_n^2.$$

Rearranging,

$$J(S;T) = \frac{1}{2}W_n^2 - \frac{1}{2}\sum_{k=0}^{n-1}(W_{k+1} - W_k)^2.$$

As observed in the last section that as $n \to \infty$ the second sumation on the right hand side converges to T, so

$$\int_0^T W(t)dW(t) = \lim_n J(S;T) = \frac{1}{2}W^2(T) - \frac{1}{2}T.$$

By Ito's formula:
Let $f(x) = x^2/2$. Therefore

$$\begin{aligned}
\frac{1}{2}W^2(T) &= f(W(t)) - f(W(0)) \\
&= \int_0^T f'(W(t))dW(t) + \frac{1}{2}\int_0^t f''(W(t))dt \\
&= \int_0^T W(t)dW(t) + \frac{1}{2}T
\end{aligned}$$

Rearranging,

$$\int_0^T W(t)dW(t) = \frac{1}{2}W^2(T) - \frac{1}{2}T.$$

3 Model of mathematical finance

3.1 Geometric Brownian motion model

Assume from now on that the process $W = \{W(t), t \geq 0\}$ is a standard Brownian motion with respect to filtration $\mathcal{F} = \{\mathcal{F}_t\}$. The process $Y(t) = Y(0)e^{aW(t)+bt}$ where a and b are constants is called a *geometric Brownian motion*. Note that $Y(t)$ is a continuous-time strictly positive stochastic process if $Y(0) > 0$.

We use geometric Brownian motion rather than standard Brownian motion to model a financial asset, since the stock price must be strictly positive, whereas standard Brownian motion can take negative values. In addition we are primarily concerned with the rate of return, or the percentage change of the stock price, not its absolute change. In this model we assume that the rates of return of a stock are independent and identically distributed under a normal distribution. In fact one of the main criticisms of the model is that empirically the rates of return do not follow a normal distribution, but any model must strike a balance between reality and tractability. Geometric Brownian motion is thus still widely used for its mathematical tractability, or in other words, because solutions can be derived analytically. For simplicity, we also assume the stock does not pay a dividend.

To model the behavior of a stock, we define the Ito process:

$$Z(t) = \sigma W(t) + \left(\mu - \frac{1}{2}\sigma^2\right) t.$$

Then

$$dZ(t) = \sigma dW(t) + \left(\mu - \frac{1}{2}\sigma^2\right) dt,$$

and

$$dZ(t)dZ(t) = \sigma^2(t)dW(t)dW(t) = \sigma^2 dt.$$

Now consider a stock price process given by

$$S(t) = S(0)e^{Z(t)} = S(0)\exp\left\{\left(\mu - \frac{1}{2}\sigma^2\right) t + \sigma W(t)\right\}.$$

where $S(t)$ is the price of the stock at time t, $S(0)$ is the initial stock price at time $t = 0$, μ is the rate of return of the stock ("drift parameter"), σ is the volatility of the stock, and $W(t)$ is a standard Brownian motion. Here we assume that the drift parameter and the stock's volatility are both constants and the distribution of $S(t)$ is log-normal.

We can write $S(t) = f(Z(t))$, where $f(x) = S(0)e^x$, $f'(x) = S(0)e^x$ and $f''(x) = S(0)e^x$. By Ito's formula,

$$\begin{aligned}
dS(t) &= f'(Z(t))dZ(t) + \frac{1}{2}f''(Z(t))dZ(t)dZ(t) \\
&= S(0)e^{Z(t)}dZ(t) + \frac{1}{2}S(0)e^{Z(t)}dZ(t)dZ(t) \\
&= S(t)\left[\sigma dW(t) + \left(\mu - \frac{1}{2}\sigma^2\right) dt\right] + \frac{1}{2}S(t)\sigma^2 dt \\
&= \mu S(t)dt + \sigma S(t)dW(t).
\end{aligned}$$

Factoring out the $S(t)$ term yields

$$\frac{dS(t)}{S(t)} = \mu dt + \sigma dW(t),$$

the differential form of the geometric Brownian motion model.

3.2 Constructing the Black-Scholes PDE by delta hedging

In order to construct the Black-Scholes partial differential equation, we first consider an investor who at each time t has a portfolio valued at $X(t)$. He invests in a money market account paying a constant rate of interest r and in a stock modeled by the geometric Brownian motion above. At each time t, the investor holds $\triangle(t)$ units of stock. The position $\triangle(t)$ is a process adapted to the filtration associated with the Brownian motion $W(t)$. The remainder of the portfolio, given by $X(t)-\triangle(t)S(t)$, is invested in the money market account.

The change in portfolio value, $dX(t)$, at each time t depends on two factors: the capital gain $\triangle(t)dS(t)$ on the stock position, and the interest earnings $r(X(t)-\triangle(t)S(t))dt$ on the cash position. So

$$\begin{aligned} dX(t) &= \triangle(t)dS(t) + r[X(t) - \triangle(t)S(t)]dt \\ &= \triangle(t)[\mu S(t)dt + \sigma dW(t)] + r[X(t) - \triangle(t)S(t)]dt \\ &= rX(t)dt + (\mu - r)\triangle(t)S(t)dt + \triangle(t)\sigma S(t)dW(t) \end{aligned}$$

We can interpret the term $\mu - r$ as the risk premium the investor receives by investing in stock.

Now consider a European call option that pays $(S(T) - K)^+$ at time T. The strike price K is some positive constant, and the option gives the holder the right to purchase one unit of stock for price K at exercise time T. Intuitively, the value of this call at any time should depend on the time t, the value of the stock price at that time $S(t)$, the model parameters r and σ, and the contractual strike price K.

We let $c(t,x)$ denote the value of the call at time t if the stock price at that time is $S(t) = x$. At the initial time, we do not know the future stock price $S(t)$, and hence we do not know the future option value $c(t, S(t))$ for $t > 0$. Our goal is to determine the continuous function $c(t,x)$ so that we have a formula for future option prices in terms of the future stock price.

We begin by noting that the differential of the value of the call option is

$$\begin{aligned} dc(t,S(t)) &= c_t(t,S(t))dt + c_x(t,S(t))dS(t) + \frac{1}{2}c_{xx}(t,S(t))dS(t)dS(t) \\ &= \left[c_t(t,S(t)) + \mu S(t)c_x(t,S(t)) + \frac{1}{2}\sigma^2 S^2(t)c_{xx}(t,S(t))\right]dt \\ &\quad + \sigma S(t)c_x(t,S(t))dW(t). \end{aligned}$$

We now construct our portfolio $X(t)$ to exactly replicate the payoffs of the European call option so that at each time t it tracks $c(t,S(t))$, assuming its initial value is the initial value of the call, i.e. $X(0) = c(0,S(0))$. In order for this to happen, we require $dX(t) = dc(t,S(t))$ for all t. This requirement arises due to the principle of no arbitrage, that is, the idea if the portfolio exactly replicates the payoffs of the option, we should not be able to concurrently hold a long position in the portfolio and a short position in the option (or vice versa) in order to gain a risk-free profit. Suppose for instance that $dX(t) \neq dc(t,S(t))$, i.e. $dc(t,S(t)) - dX(t) = h(t)dt$. Then by holding $h(t)$ units of the call option and shorting $h(t)$ units of the portfolio at each time t, we can guarantee a positive payoff of $h^2(t)dt$. The only way for no arbitrage to occur is if $h(t) = 0$, i.e. $dX(t) = dc(t,S(t))$ for all t. Recall that

$$dX(t) = rX(t)dt + \triangle(t)(\mu - r)S(t)dt + \triangle(t)\sigma S(t)dW(t)$$

By equating the coefficients of $dW(t)$, we obtain

$$\sigma S(t)c_x(t,S(t)) = \triangle(t)\sigma S(t)$$

$$\triangle(t) = c_x(t,S(t))$$

This is known as the delta-hedging rule. At each time t prior to expiration, the number of shares held by the hedge of the short option position is the partial derivative of the option value with respect to the stock price evaluated at that time t. The quantity $c_x(t,S(t))$ is called the delta of the option, and it is worth noting that this gives a dynamic hedge, i.e. the number of shares held is time dependent and hence continually changing.

Next we equate the coefficients of dt:

$$\begin{aligned} &c_t(t,S(t)) + \mu S(t)c_x(t,S(t)) + \frac{1}{2}\sigma^2 S^2(t)c_{xx}(t,S(t)) \\ &= \ rX(t) + \triangle(t)(\mu - r)S(t). \end{aligned}$$

But from the delta-hedging rule, we have $\triangle(t) = c_x(t,S(t))$. We have also assumed that the initial values of the portfolio and option are equal, i.e. $X(0) = c(0, S(0))$. If the investor dynamically uses the hedge $\triangle(t) = c_x(t, S(t))$, then he guarantees that $dX(t) = dc(t, S(t))$ for all t. It follows that $X(t) = c(t, S(t))$. Accounting for this information (also replace $S(t)$ by x) yields

$$c_t(t, x) + rxc_x(t, x) + \frac{1}{2}\sigma^2 x^2 c_{xx}(t, x) = rc(t, x),$$

which is known as the Black-Scholes partial differential equation. Note that the term containing the risk premium $\mu - r$ has been cancelled out in the equation, showing that the value of the call option is only dependent on the volatility of the underlying stock, not its rate of return. We now seek a continuous function $c(t, S(t))$ which is the solution to this equation and satisfies the terminal condition $c(T, x) = (x - K)^+$. The equation can be solved using partial differential equation methods. For our purposes, however, it is more instructive to show that the value of the call option can be expressed as the discounted expected payoff of the option under the risk-neutral measure. We will demonstrate this in the next section.

4 Girsanov's Theorem and risk-neutral measure

4.1 Radon-Nikodym derivative and Girsanov's Theorem

In this section we show that the risk-neutral measure can be derived from Girsanov's theorem, and that the Black-Scholes formula can be derived and solved by considering the value of a call option as a discounted expected payoff of the option under the risk-neutral measure. We begin with a probability space $(\Omega, \mathcal{F}, \mathbb{P})$ and a strictly positive random variable Z such that $E[Z] = 1$. We now define a new probability measure, $\tilde{\mathbb{P}}$:

$$\tilde{\mathbb{P}}(A) = \int_A Z(u) d\mathbb{P}(u) \quad \forall A \in \mathcal{F}.$$

We call Z the Radon-Nikodym derivative or likelihood ratio of $\tilde{\mathbb{P}}$ with respect to $\mathbb{P}$, and write

$$Z = \frac{d\tilde{\mathbb{P}}}{d\mathbb{P}}$$

Z is therefore analogous to a ratio of the two probability densities. Any random variable X now has two expectations, one under the original probability measure P and another under the new probability measure $\tilde{\mathbb{P}}$. We call these $E[X]$ and $\tilde{E}[X]$ respectively. These are related by:

$$\tilde{E}[X] = E[XZ].$$

Since $P(Z > 0) = 1$, $\mathbb{P}$ and $\tilde{\mathbb{P}}$ agree which sets have probability zero and are absolutely continuous with respect to each other. We can write:

$$E[X] = \tilde{E}\left[\frac{X}{Z}\right]$$

In this section we will perform a change of measure in order to change the drift of a Ito process. Throughout, we will assume that $\{W(t), 0 \leq t \leq T\}$ is a standard Brownian motion on a probability space $(\Omega, \mathcal{F}, \mathbb{P})$ and let $\{\mathcal{F}(t), 0 \leq t \leq T\}$ be the filtration for this Brownian motion.

We begin by stating Girsanov's theorem:

Girsanov's theorem. Let $\{\theta(t), 0 \leq t \leq T\}$ be a bounded adapted process. Define

$$\begin{aligned} Z(t) &= \exp\left[-\int_0^t X(u)dW(u) - \frac{1}{2}\int_0^t \theta^2(u)du\right], \\ \tilde{W}(t) &= W(t) + \int_0^t \theta(u)du, \end{aligned}$$

and define a new probability measure by

$$\tilde{\mathbb{P}}(A) = \int_A Z(T)d\mathbb{P} \quad \forall A \in \mathcal{F}.$$

Then under $\tilde{\mathbb{P}}$, the process $\{\tilde{W}(t),\ 0 \leq t \leq T\}$ is a standard Brownian motion. Note that the probability measures $\mathbb{P}$ and $\tilde{\mathbb{P}}$ are absolutely continuous with respect to each other.

In addition to Girsanov's Theorem, our asset pricing model will rely on the Martingale Representation Theorem:

Martingale Representation Theorem. Let $\{M(t),\ 0 \leq t \leq T\}$ be a martingale with respect to the filtration $\mathcal{F}$, that is, for every t, $M(t)$ is $\mathcal{F}(t)$-measurable and for $0 \leq s \leq t \leq T$, $E[M(t)|\mathcal{F}(s)] = M(s)$. Then there is an adapted process $\pi(t)$ such that

$$M(t) = M(0) + \int_0^t \pi(u)dW(u).$$

The theorem states that when the filtration is the one generated by a Brownian motion, then every martingale with respect to this filtration can be written as an initial condition plus an Ito integral with respect to the Brownian motion. The relevance of this to hedging is that the only source of uncertainty in our model is the Brownian motion. We now need to set up an asset price model in which $\mathbb{P}$ is the actual probability measure and $\tilde{\mathbb{P}}$ is the risk-neutral probability measure.

4.2 Risk-neutral measure

In our pricing model we are particularly concerned with the discounted stock price. To illustrate the main idea, we can assume that the drift parameter μ, the continuously compounded interest rate r and the volatility σ are constant. The discounted stock price process is given by

$$e^{-rt}S(t) = S(0)\exp\left[\int_0^t \sigma(u)dW(u) + \int_0^t \left(\mu - r - \frac{1}{2}\sigma^2\right)du\right]$$

and its differential is

$$\begin{aligned} de^{-rt}S(t) &= e^{-rt}[(\mu - r)S(t)dt + \sigma S(t)dW(t)] \\ &= e^{-rt}\sigma S(t)\left[\frac{\mu - r}{\sigma}dt + dW(t)\right] \\ &= e^{-rt}\sigma S(t)[\theta dt + dW(t)] \end{aligned}$$

where we call $\theta = (\mu - r)/\sigma$ the market price of risk.

We now introduce the probability measure $\tilde{\mathbb{P}}$ as defined in Girsanov's theorem:

$$\tilde{\mathbb{P}}(A) = \int_A \exp\left[-\int_0^t \theta dW(u) - \frac{1}{2}\int_0^t \theta^2 du\right] d\mathbb{P} \quad \forall A \in \mathcal{F}$$

which uses the market price of risk θ. Then under $\tilde{\mathbb{P}}$, the process $\tilde{W}(t)$ is a Brownian motion, where $\tilde{W}(t)$ is given by

$$\tilde{W}(t) = W(t) + \int_0^t \theta du.$$

Under the new Brownian motion, we rewrite the differential as

$$de^{-rt}S(t) = e^{-rt}\sigma S(t)d\tilde{W}(t)$$

In integral form, our discounted stock price is:

$$e^{-rt}S(t) = S(0) + \int_0^t e^{-ru}\sigma S(u)d\tilde{W}(u)$$

Note that under $\tilde{\mathbb{P}}$, the new probability measure defined by Girsanov's theorem, the discounted stock price is a martingale. For this reason $\tilde{\mathbb{P}}$ is called the risk-neutral probability measure.

By making the substitution $dW(t) = -\theta dt + d\tilde{W}(t)$, we can express the differential of the stock price in terms of the new Brownian motion:

$$dS(t) = rS(t)dt + \sigma S(t)d\tilde{W}(t)$$

Solving this yields

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

Note that the drift parameter of the stock μ has been accounted for by using the market price of risk θ and therefore does not feature explicitly in our model. The expected rate of return of the stock is simply r under the probability measure $\tilde{\mathbb{P}}$. This is in line with what we might expect of a risk-neutral measure.

We now consider our investor who begins with initial capital $X(0)$ and at each time t holds $\triangle^*(t)$ shares of stock, investing or borrowing at the interest rate r in order to finance this. The differential of the discounted portfolio value is given by

$$\begin{aligned} d(e^{-rt}X(t)) &= e^{-rt}\triangle^*(t)\sigma S(t)[\theta dt + dW(t)] \\ &= \triangle^*(t)d(e^{-rt}S(t)) \\ &= e^{-rt}\triangle^*(t)\sigma S(t)d\tilde{W}(t) \end{aligned}$$

The investor therefore has two investment options, a money market account with rate of return r, or a stock with expected rate of return r under $\tilde{\mathbb{P}}$. Regardless of how he invests, the mean rate of return for his portfolio will be r under $\tilde{\mathbb{P}}$, and hence the discounted value of his portfolio, $e^{-rt}X(t)$, will be a martingale.

4.3 Arbitrage-free pricing

Similar to Section 3.2, the most important element in the valuation of this option via arbitrage pricing is a replicating portfolio. We construct a self-financing portfolio with the same payoff at time T as the option.

We begin by defining the discounted value process $\{e^{-rt}V(t),\ 0 \le t \le T\}$ as the expected value of the discounted payoff, given the information up to that point, i.e.

$$e^{-rt}V(t) = \tilde{E}[e^{-rT}(S(T)-K)^+|\mathcal{F}(t)]$$

Note that at time T, $V(T)$ is simply the payoff of the option, $(S(T)-K)^+$. We can verify that the discounted value process is a $\widetilde{\mathbb{P}}$-martingale, since by iterated conditioning,

$$\begin{aligned}\tilde{E}[e^{-rT}(S(T)-K)^+|\mathcal{F}(t)] &= \tilde{E}[\tilde{E}[e^{-rT}(S(T)-K)^+|\mathcal{F}(t)]|\mathcal{F}(s)] \\ &= \tilde{E}[e^{-rT}(S(T)-K)^+|\mathcal{F}(s)].\end{aligned}$$

By the Martingale Representation Theorem, $e^{-rt}V(t)$ therefore has a representation as

$$e^{-rt}V(t) = V(0) + \int_0^t \pi(u)d\tilde{W}(u)$$

for some process π.

Let c be the price of the European call option at time 0. Therefore at time 0, the seller of a European call option receives c. He invests the entire amount c in a portfolio $X(t)$ with strategy $\triangle^*$. Then the discounted value of the portfolio is

$$e^{-rt}X(t) = c + \int_0^t e^{-ru}\triangle^*(u)\sigma S(u)d\tilde{W}(u).$$

By choosing the number of shares to hold, $\triangle^*(t)$, to satisfy

$$e^{-rt}\triangle^*(t)\sigma S(t) = \pi(t)$$

or in other words

$$\triangle^*(t) = \frac{e^{rt}\pi(t)}{\sigma S(t)}$$

the seller can create a replicating portfolio for the call option. At the time to expiration T, the seller's position in the portfolio is $X(T)$. However, he has to make a payoff of $(S(T)-K)^+$. His total wealth, Z, at time T is therefore:

$$\begin{aligned}Z &= X(T) - (S(T)-K)^+ \\ &= e^{rT}(e^{-rT}X(T) - e^{-rT}V(T)) \\ &= e^{rT}(c - V(0)).\end{aligned}$$

To summarize, the seller has a payoff of 0 at time 0, and a guaranteed, deterministic payoff of $e^{-rT}(c - V(0))$ at time T. In order for no arbitrage to occur, his payoff at time T must be equal to 0. The only way this is possible is if the price of the call option, c, is equal to the initial value of the value process, $V(0)$. The price of the call option can therefore be expressed as the expected value of the discounted payoff, given the information up to that point, i.e.

$$c = \tilde{E}[e^{-rT}(S(T) - K)^+ | \mathcal{F}(0)]$$

At this stage it appears that this argument using the Martingale Representation Theorem is not constructive. We have justified the use of the risk-neutral pricing formula, since it guarantees that a process $\pi(t)$ exists and hence a hedging strategy $\triangle^*(t)$ exists, but we have not provided a method for finding $\pi(t)$. In fact, we can derive $\pi(t)$ by use of Ito's formula, and we will do this in section 4.5.

4.4 Deriving the Black-Scholes formula

Recall that the discounted price of the option is given by

$$e^{-rt}V(t) = \tilde{E}[e^{-rT}(S(T) - K)^+ | \mathcal{F}(t)]$$

which can be rearranged to yield the price of the option at time t:

$$V(t) = \tilde{E}[e^{-r(T-t)}(S(T) - K)^+ | \mathcal{F}(t)]$$

Since geometric Brownian motion is a Markov process, this expression depends on the stock price $S(t)$ and on the time t at which the conditional expectation is computed, but not on the stock price prior to time t. In other words, there is a function $c(t, x)$ such that

$$c(t, S(t)) = \tilde{E}[e^{-r(T-t)}(S(T) - K)^+ | \mathcal{F}(t)].$$

Letting $\tau = T - t$, the time to expiration, we can write

$$\begin{aligned} S(T) &= S(t) \exp\left[\left(r - \frac{1}{2}\sigma^2\right)\tau + \sigma(\tilde{W}(T) - \tilde{W}(t))\right] \\ &= S(t) \exp\left[\left(r - \frac{1}{2}\sigma^2\right)\tau - \sigma\sqrt{\tau}Y\right] \end{aligned}$$

where Y is the standard normal random variable

$$Y = -\frac{\tilde{W}(T) - \tilde{W}(t)}{\sqrt{T - t}}$$

We can therefore express the value of our call option as

$$\begin{aligned} c(t,x) &= \tilde{E}\left[e^{-r\tau}\left(x\exp\left\{\left(r-\frac{1}{2}\sigma^2\right)\tau-\sigma\sqrt{\tau}Y\right\}-K\right)^+\right] \\ &= \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{+\infty} e^{-r\tau}\left(x\exp\left\{\left(r-\frac{1}{2}\sigma^2\right)\tau-\sigma\sqrt{\tau}y\right\}-K\right)^+ e^{-\frac{1}{2}y^2}dy \end{aligned}$$

Let

$$d_1=\frac{1}{\sigma\sqrt{\tau}}\left[\log\frac{x}{K}+\left(r-\frac{1}{2}\sigma^2\right)\tau\right].$$

One can verify that the payoff is positive if and only if $y<d_1$. Therefore

$$\begin{aligned} c(t,x) &= \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{d_1} e^{-r\tau}\left(x\exp\left\{\left(r-\frac{1}{2}\sigma^2\right)\tau-\sigma\sqrt{\tau}y\right\}-K\right)e^{-\frac{1}{2}y^2}dy \\ &= \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{d_1} x\exp\left\{-\frac{1}{2}y^2-\sigma\sqrt{\tau}y-\frac{1}{2}\sigma^2\tau\right\}dy \\ &\quad -\frac{1}{\sqrt{2\pi}}\int_{-\infty}^{d_1} e^{-r\tau}Ke^{-\frac{1}{2}y^2}dy \\ &= \frac{x}{\sqrt{2\pi}}\int_{-\infty}^{d_1}\exp\left\{-\frac{1}{2}(y+\sigma\sqrt{\tau})^2\right\}dy-e^{-r\tau}K\Phi(d_1) \\ &= \frac{x}{\sqrt{2\pi}}\int_{-\infty}^{d_1+\sigma\sqrt{\tau}}\exp\left\{-\frac{1}{2}z^2\right\}dz-e^{-r\tau}K\Phi(d_1) \\ &= x\Phi(d_2)-e^{-r\tau}K\Phi(d1) \end{aligned}$$

where $d_2=d_1+\sigma\sqrt{\tau}$ and Φ is the standard normal cumulative distribution function. We have now found an explicit solution for the value of the call option by expressing it as the discounted expected payoff of the option under the risk-neutral measure. It can in fact be verified that this is the solution to the Black-Scholes partial differential equation derived in section 3.2.

4.5 Derivation of π(t)

It now remains to find an explicit derivation of $\pi(t)$ in order for the seller to construct a dynamic hedge. Note that

$$e^{-rt}V(t)=V(0)+\int_0^t\pi(u)d\tilde{W}(u)$$

But $V(t)=c(t,S(t))$, and whence

$$d(e^{-rt}c(t,S(t)))=\pi(t)d\tilde{W}(t)$$

or equivalently (by Ito formula)

$$\pi(t) = e^{-rt} c_x(t, S(t)) \sigma S(t).$$

We have assumed that σ, r, t and $S(t)$ are known to the seller, while $c_x(t, S(t))$ can be evaluated by means of the Black-Scholes formula. By using this choice of $\pi(t)$, the seller can construct a dynamic hedge for the portfolio in section 4.3.

5 The embedding model and large deviations

The simulation of the option prices are closely related to large deviations theory and large deviations can often shed invaluable insight in the design of importance sampling schemes [**?**]. To fix ideas, consider an option with expiration time T and payoff

$$e^{-F(W)}.$$

Here $F : C[0,T] \to \mathbb{R}$ is a function with minor regularity conditions. Note that many options in the Black-Scholes model have payoffs of this form: call options, put options, barrier options, Asian options, and so on.

We can embed this problem into a larger model where we assume there are a sequence of payoffs of form

$$e^{-F(\sqrt{\varepsilon}W)/\varepsilon}.$$

The original payoff corresponds to the case where $\varepsilon = 1$. The advantage of this embedding is that now we can consider the limit asymptotics as $\varepsilon \to 0$, where it is known that large deviation asymptotics often suggest an efficient, or asymptotically optimal, change of measure for importance sampling. Some may think that the asymptotic consideration may not be suitable for $\varepsilon = 1$. However there are a few reason to adopt this approach: (1) the magnitude of ε is always relative; (2) large deviations capture some important characteristics of the stochastic system that is at least useful to give some guidelines of picking changes of measure for fixed ε; (3) This embedding has been successfully applied to option pricing [**?**].

The basic large deviations result yields that, under some regularity conditions on F,

$$-\lim_{\varepsilon \to 0} \varepsilon \log E\left[e^{-F(\sqrt{\varepsilon}W)/\varepsilon}\right] = \inf\left\{F(\phi) + I(\phi)\right\}, \tag{1}$$

where the infimum is taken over all absolutely continuous functions and

$$I(\phi) \doteq \frac{1}{2}\int_0^T [\dot{\phi}(t)]^2\,dt.$$

One way to understand this large deviations result is to formally write

$$\mathbb{P}(\sqrt{\varepsilon}W \approx \phi) \approx e^{-I(\phi)/\varepsilon}\,d\phi,$$

which is clearly consistent with the large deviation limit (**??**).

Now let us consider the importance sampling change of measure. The Girsanov theorem asserts that under the change of measure $\mathbb{Q}$ where

$$\frac{d\mathbb{Q}}{d\mathbb{P}} = \exp\left\{\frac{1}{\sqrt{\varepsilon}}\int_0^T \dot{\varphi}(t)\,dW_t - \frac{1}{2\varepsilon}\int_0^T [\dot{\varphi}(t)]^2\,dt\right\},$$

the process

$$\bar{W}_t \doteq W_t - \frac{1}{\sqrt{\varepsilon}}\varphi(t)$$

is a standard Brownian motion. Note that $\varphi(t)$ can be random, as long as it is adapted to the filtration generated by W. Under the probability measure $\mathbb{Q}$, the second moment of the importance sampling estimator is

$$E^{\mathbb{Q}}\left[e^{-2F(\sqrt{\varepsilon}W)/\varepsilon}\left(\frac{d\mathbb{P}}{d\mathbb{Q}}\right)^2\right] = E^{\mathbb{P}}\left[e^{-2F(\sqrt{\varepsilon}W)/\varepsilon}\frac{d\mathbb{P}}{d\mathbb{Q}}\right]$$

Plugging the formula of $d\mathbb{P}/d\mathbb{Q}$, the second moment equals

$$E^{\mathbb{P}}\left[\exp\left\{-\frac{2}{\varepsilon}F(\sqrt{\varepsilon}W) + \frac{1}{2\varepsilon}\int_0^T [\dot{\varphi}(t)]^2\,dt - \frac{1}{\sqrt{\varepsilon}}\int_0^T \dot{\varphi}(t)\,dW_t\right\}\right].$$

Now considers a new probability $\bar{\mathbb{Q}}$ defined by the Girsanov theorem

$$\frac{d\bar{\mathbb{Q}}}{d\mathbb{P}} = \exp\left\{-\frac{1}{\sqrt{\varepsilon}}\int_0^T \dot{\varphi}(t)\,dW_t - \frac{1}{2\varepsilon}\int_0^T [\dot{\varphi}(t)]^2\,dt\right\},$$

that is, under $\bar{\mathbb{Q}}$, the process

$$B_t \doteq W_t + \varphi(t)$$

is a standard Brownian motion. Therefore the second moment of the importance sampling estimator can be written as

$$E^{\bar{\mathbb{Q}}}\left[\exp\left\{-\frac{2}{\varepsilon}F(\sqrt{\varepsilon}W) + \frac{1}{\varepsilon}\int_0^T [\dot{\varphi}(t)]^2\,dt\right\}\right].$$

We can use the said large deviation principle to obtain the limit

$$\begin{aligned}
&\lim_{\varepsilon\to 0} \varepsilon \log E^{\mathbb{Q}}\left[e^{-2F(\sqrt{\varepsilon}W)/\varepsilon}\left(\frac{d\mathbb{P}}{d\mathbb{Q}}\right)^2\right] \\
&= \quad -\inf_{\phi}\left[2F(\phi-\varphi) - \int_0^T [\dot{\varphi}(t)]^2\,dt + \int_0^T \frac{1}{2}[\dot{\phi}(t)]^2\,dt\right] \\
&= \quad -\inf_{\phi}\left[2F(\phi) - \int_0^T [\dot{\varphi}(t)]^2\,dt + \int_0^T \frac{1}{2}[\dot{\phi}(t)+\dot{\varphi}(t)]^2\,dt\right] \\
&= \quad -\inf_{\phi}\left[2F(\phi) + \int_0^T [\dot{\phi}(t)]^2\,dt - \int_0^T \frac{1}{2}[\dot{\phi}(t)-\dot{\varphi}(t)]^2\,dt\right].
\end{aligned}$$

Therefore the problem of finding a good change of measure amounts to finding the solution to the following minimax problem (note that due to the negative sign the minimizing φ becomes the maximizing player)

$$\sup_{\varphi}\inf_{\phi}\left[2F(\phi) + \int_0^T [\dot{\phi}(t)]^2\,dt - \int_0^T \frac{1}{2}[\dot{\phi}(t)-\dot{\varphi}(t)]^2\,dt\right]. \tag{2}$$

If one can switch the order of sup and inf in the above minimax problem, it is easy to see that the optimal φ, denoted by φ^*, can be characterized by

$$\varphi^* = \phi^* \tag{3}$$

where ϕ^* is the minimizer of the problem

$$\inf_{\phi}\left[2F(\phi) + \int_0^T [\dot{\phi}(t)]^2\,dt\right]. \tag{4}$$

This consideration is indeed used predominantly in most of the study of importance sampling schemes. However, the assumption of the exchangability is a big IF. It is well known that when F is convex, this assumption does hold. If this is the case, the change of measure defined by φ^* from (**??**) and (**??**) can be shown to be optimal in a suitable asymptotic sense. Some of the examples in the thesis satisfy this assumption, for which we use the change of measure defined by ϕ^*. For illustration, we give an example where it is a call option and

$$F(\phi) = -\log\left[e^{\sigma\phi(T)} - K\right]^+.$$

with $K > 0$. To find the minimizer in (**??**), we should note that ϕ^* must be a straight line, thanks to the convexity of function $x \mapsto x^2$. Therefore, we are looking for

$$\phi^*(t) = b^* t$$

where b^* is the minimizer in the expression

$$\inf_{\phi}\left[-2\log\left[e^{b\sigma T}-K\right]^{+}+b^2T\right].$$

It is not difficult to show that b^* is the unique solution to the algebraic equation

$$(\sigma-b)e^{b\sigma T}+bK=0$$

on interval $[(\log K)/(b\sigma),\infty)$. More often than not, the function F is not convex and therefore by simply switching sup and inf will not work. There are many ways to circumspect this. But no matter how one does it, the change of measure will end up to be state dependent. One way to do it is to consider solving this minimizing problem (**??**) at every simulation step, and the resulting state dependent change of measure will be asymptotically optimal. To describe the idea, consider the special case where F is the minimum of a finite collection of convex functions, say

$$F=F_1\wedge F_2\wedge\cdots\wedge F_m$$

where F_i are all convex functions. Now suppose we solve the expanded minimizing problem (**??**) with F replaced by F_i for each i

$$V_i(t,x)\doteq\inf_{\phi}\left[2F_i(x+\phi)+\int_t^T[\dot{\phi}(t)]^2\,dt\middle|\,\phi(t)=0\right]$$

Here x can be thought of as the scaled current state of the Brownian motion W at time t, that is, $x=\sqrt{\varepsilon}W_t$. The algorithm will proceed as follows. At each (t,x), let i^* be the minimizer of $V_1(t,x)\wedge V_2(t,x)\wedge\cdots\wedge V_m(t,x)$. Then use the change of measure defined by ϕ_i^*. There are other ways to construct simpler optimal schemes, but it requires advanced knowledge of partial differential equations.

6 Comparison of importance sampling with naive Monte Carlo simulation

To demonstrate the efficacy of importance sampling algorithms, we price several options by both naive Monte Carlo simulation and with importance sampling, and compare the prices and standard errors in both cases. In order to establish a benchmark, we use options whose prices admit analytical formula.

All codes implementing the pricing models in this section are included in the appendices in section 7. These can be run on Matlab v7.4 or higher, or GNU Octave.

6.1 European vanilla call option

We start with the simplest case, and consider a European vanilla call option that has a payoff of 1 if it's in the money at exercise time, and a payoff of 0 if it's out of the money.

Analytic price:

$$\begin{aligned}
\text{Price} &= E[e^{-rT}1_{\{S_T \geq K\}}] \\
&= e^{-rT}\ P(S_T \geq K) \\
&= e^{-rT}\ P(S_0 e^{(r-\frac{1}{2}\sigma^2)T+\sigma W_T} \geq K) \\
&= e^{-rT}\ P\left(\frac{W_T}{\sqrt{T}} \geq \frac{\log\frac{K}{S} - (r-\frac{1}{2}\sigma^2)T}{\sigma\sqrt{T}}\right)
\end{aligned}$$

But following from the definition of Brownian motion, $W_T/sqrtT$ is a standard normal random variable. So

$$\text{Price} = e^{-rt}\left[1 - \Phi\left(\frac{\log\frac{K}{S} - (r-\frac{1}{2}\sigma^2)T}{\sigma\sqrt{T}}\right)\right]$$

where Φ is the cumulative density function for a standard normal distribution.

Naive simulation:

Our algorithm is as follows:

1. Simulate a standard Brownian motion W to find W_T, the final value of the Brownian motion. Do this by dividing the time interval into n time steps, such that each discrete movement is normally distributed, $N(0,\ T/n)$. W_T is then just the cumulative sum of the discrete movements.

2. Solve for $S_T = S_0 e^{(r-\frac{1}{2}\sigma^2)T+\sigma W_T}$.

3. If $S_T \geq K$, payoff = 1; otherwise, payoff = 0.

4. Repeat steps 1-3 N time and store the N payoffs.

5. Discount the payoffs by multiplying by a discount factor of e^{-rT}.

6. Price = $\frac{\sum \text{discounted payoff}}{N}$

7. Standard error = $\frac{\text{std(discounted payoffs)}}{\sqrt{N}}$

Importance sampling:

We need to find a new probability measure, $\mathbb{Q}$, that will minimize the variance in the limit. We first need to find the optimal drift, which we call u^*. An importance sampling algorithm is as follows:

1. Set $S_T = K$ and solve for u^* by
$$K = Se^{(r+\sigma u^* - \frac{1}{2}\sigma^2)T}$$
Rearranging,
$$u^* = \frac{1}{\sigma T}\log\frac{K}{S} + \frac{\sigma}{2} - \frac{r}{\sigma}.$$

2. The process $\bar{W}_t = W_t - u^* t$ is a Brownian motion under the new measure $\mathbb{Q}$. Again, simulate Brownian motion $\bar{W}$ by dividing the time interval into n time steps, then each discrete movement is $N(0,\ T/n)$. At each time step, we need to add a drift term of u^*T/n. The cumulative sum of the discrete movements will give us W_T.

3. Solve for $S_T = S_0 e^{(r-0.5\sigma^2)T+\sigma W_T}$.

4. We now multiply the payoff by our IS estimator, which is the Radon-Nikodym derivative as given by Girsanov's Theorem. The IS estimator is given by: $e^{-u^* W_T + 0.5(u^*)^2 T}$.

5. Repeat steps 1-4 N times and store the N payoffs.

6. Discount the payoffs by multiplying by a discount factor of e^{-rT}.

7. Price $= \frac{\sum \text{discounted payoff}}{N}$

8. Standard error $= \dfrac{\text{std(discounted payoffs)}}{\sqrt{N}}$

The following results were obtained by holding constant the parameters of $S_0 = 20$, $r = 0.05$, $\sigma = 0.15$, $T = 1$ and varying the strike price K. For each simulation we used 50000 iterations and 1000 timesteps for the simulation of Brownian motion.

K	21	24	30	35	40
true price	0.45023	0.16099	6.89 x 10^{-3}	2.45 x 10^{-4}	6.11 x 10^{-6}
MC price	0.44761	0.16040	6.93 x 10^{-3}	2.66 x 10^{-5}	1.90 x 10^{-4}
MC error	2.12 x 10^{-3}	1.59 x 10^{-3}	3.62 x 10^{-4}	7.12 x 10^{-5}	1.90 x 10^{-5}
IS price	0.44938	0.16047	6.88 x 10^{-3}	2.48 x 10^{-4}	6.13 x 10^{-6}
IS error	2.02 x 10^{-3}	8.60 x 10^{-4}	5.14 x 10^{-5}	2.18 x 10^{-6}	6.10 x 10^{-8}

By comparing the standard errors, we see that the IS algorithm outperforms the naive Monte Carlo method in each of these cases. Note that as the strike price K is increased, the event that the option expires in the money becomes rarer, and the outperformance of the IS algorithm over the naive Monte Carlo method increases, as we might expect.

It is also noteworthy that in the extreme case where the strike price is twice the initial stock price ($K = 40$), the naive Monte Carlo simulation gives an answer that is inaccurate by an order of magnitude with a standard error as large as the answer itself, while the IS algorithm continues to provide a reasonable approximation to the true value.

6.2 Standard European call option

We now price a slightly more complicated option, the standard European call option. The main feature of this option is that at expiration, it pays off $\max[S_T - K, 0]$.

Analytic price:
Under the Black-Scholes framework, the option price is given by:

$$C = S\phi(d1) - Ke^{-rT}\phi(d2)$$

where

$$\begin{aligned} d1 &= \frac{\log(S/K) + (r + \sigma^2/2)T}{\sigma\sqrt{T}} \\ \text{and} \quad d2 &= d1 - \sigma\sqrt{T}. \end{aligned}$$

Naive Monte Carlo simulation:
The algorithm is the same as it was for the vanilla option, with the exception of the payoff:

1. Simulate W_t, standard Brownian motion, to find W_T.
2. Solve for $S_T = S_0 e^{(r - \frac{1}{2}\sigma^2)T + \sigma W_T}$.
3. If $S_T \geq K$, payoff $= S_T - K$; otherwise, payoff $= 0$.
4. Repeat steps 1-3 N times and store the N payoffs.
5. Discount the payoffs by multiplying by a discount factor of e^{-rT}.

6. Price $= \frac{\sum \text{discounted payoff}}{N}$

7. Standard error $= \frac{\text{std(discounted payoffs)}}{\sqrt{N}}$

Importance sampling:
To find the optimal drift, we first minimize the cost function

$$\frac{1}{2}\int_0^T (\frac{d\phi}{dt})^2(s)ds + F(\phi_T)$$

where

$$F(x) = -log(S_0 e^{(r-\frac{1}{2}\sigma^2)T+uT\sigma} - K)^+.$$

Using Holder's inequality, we can prove that the integral $\int_0^T \frac{1}{2}(\frac{d\phi}{dt})^2(s)ds$ is smaller for a straight line than for any curve. This implies that the minimizing value of ϕ is a straight line, or $\frac{d\phi}{dt}$ is a constant, which we call u. We now find u to minimize

$$f(u) = \frac{1}{2}u^2T - \log(S_0 e^{(r-\frac{1}{2}\sigma^2)T+uT\sigma} - K)^+$$

Since the payoff is positive only when $S_T > K$ and 0 otherwise, we know u has to be such that $S_0 e^{(r-\frac{1}{2}\sigma^2)T+uT\sigma} - K > 0$, implying we consider only values of u that yield positive payoffs. We find the minimizing value of u by equating the derivative of $f(u)$ to 0:

$$f'(u) = uT - \frac{T\sigma S_0 e^{(r-\frac{1}{2}\sigma^2)T+uT\sigma}}{S_0 e^{(r-\frac{1}{2}\sigma^2)T+uT\sigma} - K}$$

Solving for $f'(u) = 0$ by bisection methods gives the minimization value of u, which we call u^*. Note that we need to verify that function is monotone and that there is only one root in the interval.

1. With u^* in hand, we again simulate $\bar{W}_t$ and add on u^*T at each time step to get W_T.

2. Solve for $S_T = S_0 e^{(r-0.5\sigma^2)T+\sigma W_T}$.

3. We now need to multiply our payoff by our IS estimator, the Radon-Nikodym derivative as given by Girsanov's Theorem. The IS estimator is given by: $e^{-u^* W_T + 0.5(u^*)^2 T}$.

4. Repeat steps 1-3 N times and store the N payoffs.

5. Discount the payoffs by multiplying by a discount factor of e^{-rT}.

6. Price = $\frac{\sum \text{discounted payoff}}{N}$

7. Standard error = $\frac{\text{std(discounted payoffs)}}{\sqrt{N}}$

6.3 Results

The following results were obtained by holding constant the parameters of $S_0 = 20$, $r = 0.05$, $\sigma = 0.15$, $T = 1$ and varying the strike price K. For each simulation we used 50000 iterations and 1000 timesteps for the simulation of Brownian motion.

H	21	24	27	30	35
true price	1.2071	0.33200	0.067148	0.010650	3.3794 x 10^{-4}
MC price	1.1851	0.33570	0.066218	0.011494	1.8044 x 10^{-4}
MC error	8.52 x 10^{-3}	4.64 x 10^{-3}	1.96 x 10^{-3}	8.16 x 10^{-4}	7.08 x 10^{-5}
IS price	1.2076	0.33295	0.067462	0.010596	3.3692 x 10^{-4}
IS error	2.87 x 10^{-3}	1.10 x 10^{-3}	2.76 x 10^{-4}	5.11 x 10^{-5}	1.95 x 10^{-6}

By comparing the standard errors, we see that the IS algorithm outperforms the naive Monte Carlo method in each of these cases. Note that as the strike price K is increased, the event that the option expires in the money becomes rarer, and the outperformance of the IS algorithm over the naive Monte Carlo method increases, as we might expect. At the extreme value of $K = 35$, the Monte Carlo method gives a completely inaccurate answer that is half the value of the true price, while the IS algorithm continues to yield a reasonable approximation.

6.4 Knock-in European barrier put option

We now price a knock-in European barrier put option by naive Monte Carlo simulation and two different importance sampling schemes. The key feature of this option is that it is only activated when the stock hits a pre-agreed 'barrier' value, H, which is greater than the initial stock price, S_0. If the stock does not breach the barrier before the time of expiration, T, the option expires worthless. If the barrier is hit before expiration, the option pays off $\max[K - S_T, 0]$.

Analytic price: The analytic price is given by [**?**]

$$\text{Price} = -S_0 \left(\frac{H}{S}\right)^{2\lambda} \Phi(-y) + Ke^{-rT} \left(\frac{H}{S}\right) 2\lambda - 2\Phi(-y + \sigma\sqrt{T}),$$

where

$$\begin{aligned} \lambda &= \frac{r+\sigma^2/2}{\sigma^2} \\ \text{and} \quad y &= \frac{\log(\frac{H^2}{S_0 K})}{\sigma\sqrt{T}} \end{aligned}$$

Naive Monte Carlo simulation:
The algorithm is as follows:

1. Simulate a standard Brownian motion W_t.
2. Compute S_t at each time step by the formula $S_t = S_0 e^{(r-0.5\sigma^2)t+\sigma W_t}$.
3. If $\max(S_t) \geq H$ and $K \geq S_T$, payoff $= K - S_T$; otherwise payoff $= 0$.
4. Repeat steps 1-3 N times and store the N payoffs.
5. Discount the payoffs by multiplying by a discount factor of e^{-rT}.
6. Price $= \frac{\sum \text{discounted payoff}}{N}$
7. Standard error $= \frac{\text{std(discounted payoffs)}}{\sqrt{N}}$

Importance sampling scheme 1 (fixed-time): We implement two separate importance sampling schemes and compare their efficiency. The first scheme uses one change of measure until a fixed time t^* before using a second change of measure. For this reason we will refer to it as a fixed-time scheme. We first need to find values of t^* and k^* that minimize the following cost function:

$$\frac{t^*}{2\sigma^*}\left[\frac{\log(H/S_0)}{t^*} - r + \frac{\sigma^2}{2}\right]^2 + \frac{T-t^*}{2\sigma^*}\left[\frac{\log(k^*/H)}{T-t^*} - r + \frac{\sigma^2}{2}\right]^2 - \log(K-k^*)$$

with the following restrictions: $0 \leq t^* \leq T$ and $0 \leq k^* \leq K$. The intuition for this is that the optimal trajectory is a two-piece straight line which hits the barrier H at time t^* and hit price k^* at the expiration time T. We can evaluate this function numerically to obtain the minimizing values of t^* and k^*. The algorithm is as follows:

1. For time 0 until t^*, find the optimal drift, $u_1^* = \frac{1}{\sigma t^*}\log\frac{H}{S_0} + \frac{1}{2}\sigma - \frac{r}{\sigma}$.
2. With u_1^* in hand, we again simulate W_t from time 0 until t^* and add on $u_1^* T$. This gives us the new Brownian motion, which we call $\tilde{W}_t$.

3. At each time step, compute the stock price $S_t = S_0 e^{(r-\frac{1}{2}\sigma^2)t+\sigma\tilde{W}_t}$. Let S_{t*} be the stock price at time t^*.

4. Now for time t^* to T, compute the optimal drift, $u_2^* = \frac{1}{\sigma(T-t^*)} \log \frac{k^*}{S_{t*}} + \frac{1}{2}\sigma - \frac{r}{\sigma}$.

5. Again, simulate a standard Brownian motion W_t from time t^* and add on u_2^* at each time step. This gives us the new Brownian motion, which we call $\hat{W}_t$.

6. At each time step, compute the stock price $S_t = S_{t*} e^{(r-\frac{1}{2}\sigma^2)t+\sigma\hat{W}_t}$.

7. If $\max(S_t) \geq H$ and $K \geq S_T$, payoff $= K - S_T$; otherwise, payoff $= 0$.

8. Repeat steps 2-7 N times and store the N payoffs.

9. We need to multiply our payoffs by the IS estimator, as given by Girsanov's Theorem. The IS estimator is the product of the two IS estimators for each portion of the simulation:

$$e^{-u_1^*\tilde{W}_{t*}+0.5(u_1^*)^2T} \ . \ e^{-u_2^*\hat{W}_T+0.5(u_2^*)^2T}.$$

10. Discount the payoffs by multiplying by a discount factor of e^{-rT}.

11. Price $= \frac{\sum \text{discounted payoff}}{N}$

12. Standard error $= \dfrac{\text{std(discounted payoffs)}}{\sqrt{N}}$

Importance sampling scheme 2 (state-dependence):

In the second importance sampling scheme, we simulate the stock price using one change of measure until the barrier is hit, then use a second change of measure until the barrier is hit. If the barrier is not hit, we do not use a second change of measure and the option expires worthless. The time when we change the measure is therefore a function of the stock price, or the state of the system, so this is a state-dependent scheme. The algorithm is as follows:

1. As in the previous IS scheme, find the initial optimal drift, $u_1^* = \frac{1}{\sigma t^*} \log \frac{H}{S_0} + \frac{1}{2}\sigma - \frac{r}{\sigma}$.

2. We again simulate a Brownian motion W_t under the u_1^* measure, calling this $\tilde{W}_t$. At each time step, compute the stock price, $S_t = S_0 e^{(r-\frac{1}{2}\sigma^2)t+\sigma\tilde{W}_t}$.

3. Continue the simulation as long as $S_t < H$ and $t < T$. If $t \geq T$ is reached before $S_t \geq H$, end the simulation. The payoff of the option is 0, since the barrier has not been hit.

4. If S_t hits H before time T is reached, call the time reached $\tilde{t}$ and let $\hat{t} = T - \tilde{t}$, the time left to expiration. Let $\tilde{S}_t$ be the stock price at $\tilde{t}$.

5. Compute the new optimal drift, $u_3^* = \frac{1}{\sigma \hat{t}} \log \frac{k^*}{\tilde{S}_t} + \frac{1}{2}\sigma - \frac{r}{\sigma}$.

6. Simulate a new Brownian motion with u_3^* until T. Call this new Brownian motion $\hat{W}_t$.

7. At time T, if $K \geq S_T$, payoff $= K - S_T$; otherwise, payoff $= 0$.

8. Repeat steps 2-7 N times and store the N payoffs.

9. We need to multiply our payoffs by the IS estimator, as given by Girsanov's Theorem. The IS estimator is the product of the two IS estimators for each portion of the simulation:

10. Discount the payoffs by multiplying by a discount factor of e^{-rT}.

11. Price $= \frac{\sum \text{discounted payoff}}{N}$

12. Standard error $= \frac{\text{std(discounted payoffs)}}{\sqrt{N}}$

6.5 Results

The following results were obtained by holding constant the parameters of $S_0 = 20$, $r = 0.05$, $\sigma = 0.15$, $T = 1$, $K = 21$ and varying the barrier price H. For each simulation we used 20000 iterations and 1000 timesteps for the simulation of Brownian motion.

H	22	22.5	23	23.5	24
true price	0.21762	0.12346	0.066246	0.033682	0.016258
MC price	0.21083	0.11984	0.064282	0.029487	0.017658
MC error	9.16×10^{-3}	4.96×10^{-3}	3.43×10^{-3}	2.25×10^{-3}	1.72×10^{-3}
IS1 price	0.21379	0.12071	0.065332	0.031749	0.015827
IS1 error	7.53×10^{-3}	4.31×10^{-3}	2.80×10^{-3}	2.12×10^{-3}	9.42×10^{-4}
IS2 price	0.21603	0.12117	0.065967	0.032929	0.016058
IS2 error	2.55×10^{-3}	1.04×10^{-3}	5.88×10^{-4}	3.08×10^{-4}	1.56×10^{-4}

By comparing the standard errors, we note that both IS schemes outperform the naive Monte Carlo method. As the barrier H increases, the event that the stock hits the barrier and the option expires in the money becomes rarer, and the outperformance of the IS schemes over the naive Monte Carlo method increases.

We also note that IS2, the state dependent scheme is more efficient than IS1, the fixed-time scheme, in some cases giving standard errors an order of magnitude smaller than in IS1. However, any outperformance of this algorithm must be weighed against its longer runtime. Since it is a state dependent algorithm, it needs to verify at each time step if the stock has hit the barrier, requiring a costly `while`-loop.

7 Appendices

This section contains the codes for the implementation of all pricing models in this paper. All programs will run on Matlab v7.4 or GNU Octave.

7.1 European vanilla call option

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% This program prices a vanilla call option
% by both naive Monte Carlo simulation
% and Monte Carlo simulation with importance sampling.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clear all
close all
tic
% Declaration of variables:
%%%%%%%%%%%%%%%%%%%%%%%%%%%%
N = input('Enter number of iterations: ');
s0 = input('Enter initial stock price: ');
strike = input('Enter strike price: ');
r = input('Enter interest rate: ');
sigma = input('Enter volatility: ');
time = 1; %For simplicity sake, assume 1 year.
timesteps = 1000; % Also for simplicity, 1000 time steps.
naive_payoff = zeros(N,1); % Our payoff matrix
naive_stock_matrix = zeros(N,1); % Our stock matrix
IS_payoff = zeros(N,1); % IS payoff matrix
IS_stock_matrix = zeros(N,1); % IS stock matrix
u_star = (1/(sigma*time))*(log(strike/s0)) + 0.5*sigma - (r/sigma);
true_value = exp(-r*time)*(1-normcdf((log(strike/s0)-
((r-0.5*sigma*sigma)*time))/(sigma*sqrt(time)),0,1));
% Naive Monte Carlo pricer
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    for i = 1:N %Simulate 10000 draws
    % Simulation of Brownian motion from N(0, time/timesteps):
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
        W = [0; cumsum(randn(timesteps,1).*sqrt(time/timesteps))];
        W_final = W(timesteps+1); % Final value of Brownian motion
        % Final stock price:
        S_final = s0 * exp ((r - 0.5 * (sigma^2))*time + sigma * W_final);
        if S_final >= strike
             naive_payoff(i,1) = 1*exp(-r*time);
        else naive_payoff(i,1) = 0;
        end
    naive_stock_matrix(i,1) = S_final;
    end
    naive_price = sum(naive_payoff)/N;
    naive_error = std(naive_payoff)/sqrt(N);
% Importance Sampling pricer
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    for i = 1:N %Simulate 10000 draws
% Simulation of Brownian motion from N(0, time/timesteps):
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    ISW = [0; cumsum(randn(timesteps,1).*sqrt(time/timesteps)
     + u_star*(time/timesteps))];
```

```
    ISW_final = ISW(timesteps+1); % Final value of Brownian motion
    % Final stock price:
    IS_final = s0 * exp ((r - 0.5 * (sigma^2))*time + sigma * ISW_final);
    if IS_final >= strike
         IS_payoff(i,1) = 1 * exp (-u_star*ISW_final + 0.5*u_star*u_star*time)
          *exp(-r*time);
    else IS_payoff(i,1) = 0;
    end
    IS_stock_matrix(i,1) = IS_final;
    end
    IS_price = sum(IS_payoff)/N;
    IS_error = std(IS_payoff)/sqrt(N);
disp(sprintf('True value is                %12.9f',true_value));
disp(sprintf('Naive Monte Carlo price is   %12.9f',naive_price));
disp(sprintf('Importance sampling price is %12.9f',IS_price));
disp(sprintf('Naive Monte Carlo error is   %12.9f',naive_error));
disp(sprintf('Importance sampling error is %12.9f',IS_error));
toc
```

7.2 Standard European call option

```
%==============================================================
% This program prices a standard European call option
% by both naive Monte Carlo simulation
% and importance sampling.
%==============================================================
clear all
close all
tic
% Declaration of variables:
%%%%%%%%%%%%%%%%%%%%%%%%%%
N = 10000; % number of iterations
s0 = 20; % Initial stock price
strike = 21; % Needs to be high.
r = 0.05; % Interest rate
sigma = 0.15; % Volatility.
time = 1; %For simplicity sake, assume 1 year.
timesteps = 1000; % Also for simplicity, 1000 time steps.
naive_payoff = zeros(N,1); % Our payoff matrix
naive_stock_matrix = zeros(N,1); % Our stock matrix
IS_payoff = zeros(N,1); % IS payoff matrix
IS_stock_matrix = zeros(N,1); % IS stock matrix
d1 = (log(s0/strike) + (r+0.5*sigma*sigma)*time)/(sigma*sqrt(time));
d2 = d1 - (sigma*sqrt(time));
true_value = s0*normcdf(d1,0,1) - strike*exp(-r*time)*normcdf(d2,0,1); %Black-Scholes
%%%%%%%%%%%%%%%%%
% Naive Monte Carlo Case
%%%%%%%%%%%%%%%%%
for i = 1:N
% Simulation of Brownian motion from N(0, time/timesteps):
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    W = [0; cumsum(randn(timesteps,1).*sqrt(time/timesteps))];
    W_final = W(timesteps+1); % Final value of Brownian motion
    % Final stock price:
    S_final = s0 * exp ((r - 0.5 * (sigma^2))*time + sigma * W_final);
    if S_final >= strike
         naive_payoff(i,1) = S_final - strike;
```

```
    else naive_payoff(i,1) = 0;
    end
naive_stock_matrix(i,1) = S_final;
end
naive_price = sum(naive_payoff)/N;
naive_error = std(naive_payoff)/sqrt(N);
%%%%%%%%%%%%%%%%%%%%
% Importance Sampling Case
%%%%%%%%%%%%%%%%%%%%
x = linspace(0,2);
z= (log(strike/s0)-(r-0.5*sigma*sigma)*time)/(sigma*time);
x = x+z;
y = 0.5*(x.^2)*time
- log(s0*exp((r-0.5*sigma*sigma)*time + x * time * sigma) - strike);
plot(x,y)
figure
y_prime = x*time - ((s0*exp((r-0.5*sigma*sigma) + x*sigma*time)*time*sigma)/
(s0*exp((r-0.5*sigma*sigma) + x*sigma*time) - strike));
plot(x,y_prime)
% We have to minimize this function.
u_star = Bisection('f_prime',z,z+1);
for i = 1:N
% Simulation of Brownian motion from N(0, time/timesteps):
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    ISW = [0; cumsum(randn(timesteps,1).*sqrt(time/timesteps)
     + u_star*(time/timesteps))];
    ISW_final = ISW(timesteps+1); % Final value of Brownian motion
    % Final stock price:
    IS_final = s0 * exp ((r - 0.5 * (sigma^2))*time + sigma * ISW_final);
    if IS_final >= strike
         IS_payoff(i,1) = (IS_final - strike) * exp (-u_star*ISW_final
          + 0.5*u_star*u_star*time);
    else IS_payoff(i,1) = 0;
    end
IS_stock_matrix(i,1) = IS_final;
end
IS_price = sum(IS_payoff)/N;
IS_error = std(IS_payoff)/sqrt(N);
disp(sprintf('u* is %2.9f',u_star));
disp(sprintf('True value is                %12.9f',true_value));
disp(sprintf('Naive Monte Carlo price is   %12.9f',naive_price));
disp(sprintf('Importance sampling price is %12.9f',IS_price));
disp(sprintf('Naive Monte Carlo error is   %12.9f',naive_error));
disp(sprintf('Importance sampling error is %12.9f',IS_error));
toc
```

7.3 European knock-in barrier put option

```
%==============================================================
% This program prices a knock in barrier put option
% by naive Monte Carlo simulation and importance sampling.
%
% It investigates which of the two cases: (i) fixed time; or
% (ii) state dependence gives a better IS scheme.
%
%==============================================================
clear all
```

```
close all
tic
% Declaration of variables:
%%%%%%%%%%%%%%%%%%%%%%%%%%%%
N = 50000; % number of iterations
s0 = 20; % Initial stock price.
H = 24;
K = 21; % Needs to be high.
r = 0.05; % Interest rate
sigma = 0.15; % Volatility.
T = 1; %For simplicity sake, assume 1 year.
timesteps = 1000; % Also for simplicity, 1000 time steps.
delta_t = T/timesteps;
lambda = (r+0.5*sigma^2)/(sigma^2);
y = log(H^2/(s0*K))/(sigma*sqrt(T)) + lambda*sigma*sqrt(T);
P = -s0*(H/s0)^(2*lambda)*normcdf(-y,0,1)
+ K*exp(-r*T)*(H/s0)^(2*lambda-2)
*normcdf(-y+sigma*sqrt(T),0,1);
naive_payoff = zeros(N,1); % Our payoff matrix
IS1_payoff = zeros(N,1);
IS2_payoff = zeros(N,1);% Our payoff matrix
%%%%%%%%%%%%%%%%%%%
% Naive Monte Carlo Case
%%%%%%%%%%%%%%%%%%%
for i = 1:N
% Simulation of Brownian motion from N(0, delta_t):
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    S = zeros(timesteps+1,1);
    S(1) = s0;
    W = [0; cumsum(randn(timesteps,1).*sqrt(delta_t))];
    for j = 2:(timesteps+1)
        S(j) = s0 * exp ((r - 0.5 * (sigma^2))*((j-1)*(delta_t)) + sigma * W(j));
    end
    S_final = S(timesteps+1);
    if ((max(S) >= H) & (K >= S_final))
        naive_payoff(i) = (K-S_final)*exp(-r*T);
    else
        naive_payoff(i) = 0;
    end
end
naive_price = sum(naive_payoff)/N;
naive_error = std(naive_payoff)/sqrt(N);
%%%%%%%%%%%%%%%%%%%%%%
% Importance sampling setup
%
%%%%%%%%%%%%%%%%%%%%%%
J = 300; %Number of points in matrix
t=linspace(0.01,T-0.01, J); %time, this shows up on the row index.
k=linspace(0.01,K-0.01, J); %target stock price, this shows up on the column index.
Y = zeros(J,J);
for i = 1:J;
    for j = 1:J;
        Y(i,j) = (t(i)/(2*sigma^2))*((log(H)-log(s0))/t(i) - r + 0.5*sigma^2)^2
         + ((T-t(i))/(2*sigma^2))*((log(k(j))-log(H))/(T-t(i))
- r + 0.5*sigma^2)^2 - log(K-k(j));
    end
```

```
end
[a,b] = min(Y);
[c,d] = min(a);
k_star = k(d); %These are the values that minimize Y
t_star = t(b(d)); %These are the values that minimize Y
t_bar = T - t_star;
u_star1 = (1/(sigma*t_star))*(log(H/s0)) + 0.5*sigma - (r/sigma);
u_star1new = (1/(sigma*t_star))*(log(H/s0));
steps1 = floor((t_star/T)*timesteps);
steps2 = timesteps - steps1;
% Importance Sampling scheme 1: fixed time
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    for i = 1:N
% Simulation of Brownian motion from time 0 to t_star:
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    ISW1 = [0; cumsum(randn(steps1,1).*sqrt(delta_t) + u_star1new*(delta_t))];
    S1 = zeros(timesteps+1,1);
    S1(1) = s0;
    for j = 2:(steps1+1)
        S1(j) = s0 * exp ((r - 0.5 * (sigma^2))*(j-1)*(delta_t) + sigma * ISW1(j));
    end
    ISW1_final = ISW1(steps1+1);
    S_star = S1(steps1+1); %stock price at t_star
    %Now we simulate from t_star to T (call this t_bar for ease):
    u_star2 = (1/(sigma*t_bar))*(log(k_star/H)) + 0.5*sigma - (r/sigma);
    u_star2new = (1/(sigma*t_bar))*(log(k_star/H));
    ISW2 = [0; cumsum(randn(steps2,1).*sqrt(delta_t) + u_star2new*(delta_t))];
    ISW2_final = ISW2(steps2+1);% - ISW1_final;
    for j = 2:(steps2+1)
        S1(steps1+j) = S_star * exp ((r - 0.5 * (sigma^2))*(j-1)*(delta_t)
          + sigma * ISW2(j));
    end
    IS1_final = S1(timesteps+1); %final stock price
    if ((max(S1) >= H) & (K >= IS1_final))
        IS1_payoff(i) = exp(-r*T)*(K - IS1_final)*
         exp(-u_star1*ISW1_final + 0.5*u_star1*u_star1*t_star)
     *exp(-u_star2*ISW2_final + 0.5*u_star2*u_star2*t_bar);
    else
        IS1_payoff(i) = 0;
    end
    end
IS1_price = sum(IS1_payoff)/N;
IS1_error = std(IS1_payoff)/sqrt(N);
% Importance Sampling scheme 2: state dependence
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
for i = 1:N
% Simulation of Brownian motion from time 0 until barrier hit (t_tilde):
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    S2 = s0;
    t_count = 0;
    ISW3_final = 0;
    while ((S2 < H) & (t_count < timesteps))
        ISW3 = randn(1).*sqrt(delta_t) + u_star1new*(delta_t);
        S2 = S2 * exp ((r - 0.5 * (sigma^2))*(delta_t) + sigma * ISW3);
        t_count = t_count + 1;
        ISW3_final = ISW3_final + ISW3;
```

```
    end
    if t_count >= timesteps
        IS2_final = S2;
        IS2_payoff(i,1) = 0;
    else
       t_tilde = (t_count/timesteps) * T;
       t_hat = T - t_tilde;
       t_left = timesteps - t_count;
       u_star3 = (1/(sigma*t_hat))*(log(k_star/S2)) + 0.5*sigma - (r/sigma);
       u_star3new = (1/(sigma*t_hat))*(log(k_star/S2));
       ISW4 = [0; cumsum(randn(t_left,1).*sqrt(delta_t) + u_star3new*(delta_t))];
       ISW4_final = ISW4(t_left + 1);
       IS2_final =  S2 * exp ((r - 0.5 * (sigma^2))*(t_hat) + sigma * ISW4_final);
       if K >= IS2_final
            IS2_payoff(i) = exp(-r*T)*(K - IS2_final)*
             exp(-u_star1*ISW3_final + 0.5*u_star1*u_star1*t_tilde)*
exp(-u_star3*ISW4_final + 0.5*u_star3*u_star3*t_hat);
       else
            IS2_payoff(i) = 0;
       end
    end
end
IS2_price = sum(IS2_payoff)/N;
IS2_error = std(IS2_payoff)/sqrt(N);
disp('                                                 ');
disp('----------------------------------------------');
disp(sprintf('k_star is   %12.9f',k_star));
disp(sprintf('t_star is %12.9f',t_star));
disp(sprintf('Analytic price is              %12.9f',P));
disp(sprintf('Naive Monte Carlo price is     %12.9f',naive_price));
disp(sprintf('Importance sampling 1 price is %12.9f',IS1_price));
disp(sprintf('Importance sampling 2 price is %12.9f',IS2_price));
disp(sprintf('Naive Monte Carlo error is     %12.9f',naive_error));
disp(sprintf('Importance sampling 1 error is %12.9f',IS1_error));
disp(sprintf('Importance sampling 2 error is %12.9f',IS2_error));
toc
```

References

[1] P. Dupuis and H. Wang (2004).Importance Sampling, Large Deviations, and Differential Games. *Stoch. and Stoch. Reports.* **76**, 481–508.

[2] P. Glasserman and P. Heidelberger and P. Shahabuddin (1999), Asymptotically optimal importance sampling and stratification for pricing path-dependent options, *Mathematical Finance* **9**, 117–152.

[3] J. Hull (1989). *Options, Futures, and Other Derivatives*, Prentice Hall, New Jersey.

[4] I. Karatzas and S. E. Shreve (1988). *Brownian Motion and Stochastic Calculus*, Springer-Verlag, New York.