Monte Carlo Simulation

4.1 INTRODUCTION

Many complex derivatives exist for which analytical formulae are not possible. Monte Carlo simulation (first used by Boyle, 1977) provides a simple and flexible method for valuing these types of instruments. It can deal easily with multiple random factors; for example options on multiple assets, random volatility or random interest rates. Monte Carlo simulation also allows the incorporation of more realistic asset price processes, such as jumps in asset prices, and more realistic market conditions such as the discrete fixing of exotic path-dependent options. It can also give insights into the effectiveness of a hedge. However, Monte Carlo simulation is computationally inefficient in its basic form. In this chapter, after first introducing the Monte Carlo simulation methods, we show how to improve its efficiency using control variates and quasi-random numbers (deterministic sequences). We also describe in detail how Monte Carlo simulation can be used to value complex path-dependent options. The interested reader is recommended to study Ripley (1987) for a general discussion of stochastic simulation.

4.2 VALUATION BY SIMULATION

In Chapter 1 it was shown that the value of an option is the risk-neutral expectation of its discounted pay-off. We can obtain an estimate of this expectation by computing the average of a large number of discounted payoffs. Consider a European-style option which pays $C_T$ at the maturity date $T$. Firstly, we simulate the risk-neutral processes for the state variables from their values today, time zero to the maturity date $T$ and compute the pay-off of the contingent claim, $C_{T,j}$ for this simulation $(j)$. Then we discount this pay-off using the simulated short-term interest rate sequence:

$$C_{0,j} = \exp \left( - \int_0^T r_u \, du \right) C_{T,j} \quad (4.1)$$

In the case of constant interest rates equation (4.1) simplifies to

$$C_{0,j} = \exp(-rT)C_{T,j}$$

The simulations are repeated many (say $M$) times and the average of all the outcomes is taken

$$\hat{C}_0 = \frac{1}{M} \sum_{j=1}^{M} C_{0,j} \quad (4.2)$$
Monte Carlo simulation

where $\hat{C}_0$ is an estimate of the true value of the option $C_0$, but with an error due to the fact that it is an average of randomly generated samples and so is itself random. A measure of the error is the standard deviation of $\hat{C}_0$ which is called the standard error SE(.) and can be estimated as the sample standard deviation of $C_{0,j}$ divided by the square root of the number of samples (see Hines and Montgomery, 1980, for an introduction to probability and statistics).

$$SE(\hat{C}_0) = \frac{SD(C_{0,j})}{\sqrt{M}}$$

where

$$SD(C_{0,j}) = \sqrt{\frac{1}{M-1} \sum_{j=1}^{M} (C_{0,j} - \hat{C}_0)^2}$$

Let us look at a simple, specific example in detail: a standard European call option in the Black–Scholes world. Here interest rates are constant and so, as noted above, the discounting term in (4.1) becomes $\exp(-rT)$. This is the same for all simulations and so can be taken out of equation (4.1) and applied once to the average obtained using equation (4.2). In order to implement Monte Carlo simulation we need to simulate the geometric Brownian motion (GBM) process for the underlying asset:

$$dS_t = (\mu - \delta)S_t \, dt + \sigma S_t \, dz_t$$

The best way to simulate a variable following GBM is via the process for the natural logarithm of the variable which follows arithmetic Brownian motion and is normally distributed. Let $x_t = \ln(S_t)$ then we have

$$dx_t = \nu \, dt + \sigma \, dz_t, \quad \nu = \mu - \delta - \frac{1}{2}\sigma^2$$

Equation (4.5) can be discretised by changing the infinitesimals $dx$, $dt$ and $dz$ into small changes $\Delta x$, $\Delta t$ and $\Delta z$:

$$\Delta x = \nu \Delta t + \sigma \Delta z$$

This representation involves no approximation because it is actually the solution of the SDE (4.5) which we can write as

$$x_{t+\Delta t} = x_t + \nu \Delta t + \sigma (z_{t+\Delta t} - z_t)$$

In terms of the asset price $S$ we have

$$S_{t+\Delta t} = S_t \exp(\nu \Delta t + \sigma (z_{t+\Delta t} - z_t))$$

where $z_t$ would normally be defined as being equal to zero. The random increment $z_{t+\Delta t} - z_t$ has mean zero and a variance of $\Delta t$, it can therefore be simulated by random samples of $\sqrt{\Delta t} \varepsilon$, where $\varepsilon$ is a sample from a standard normal distribution. Equation (4.8) therefore provides a way of simulating values of $S_t$. We divide the time period over which we wish to simulate $S_t$, in this case $(0, T)$, into $N$ intervals such that $\Delta t = T/N$. We can
Figure 4.1 illustrates a set of $M = 100$ simulated paths using (4.9) repeatedly with typical parameter values for a stock: $S = 100$, $\sigma = 20$ per cent, $r = 6$ per cent, $T = 1$ year, $N = 365$. For each simulated path we compute the pay-off of the call option $\max(0, S_T - K)$. To obtain the estimate of the call price we simply take the discounted average of these simulated pay-offs

$$\hat{C}_0 = \exp(-rT) \frac{1}{M} \sum_{j=1}^{M} \max(0, S_{T,j} - K)$$

(4.10)

Note that for this simple example, since we have the solution of the underlying SDE (equation (4.7)), we can generate the samples of $S_T$ directly without simulating the entire path as shown in Figure 4.1. This is not the case in general, as we will see later; normally we can only obtain an approximate discretisation of the SDE which must be simulated with relatively small time steps. Figure 4.2 gives a pseudo-code implementation of the Monte Carlo valuation of a European call option.

Once again, note that to compute the European call option estimate under GBM we can set $N = 1$, but this is not the case in general.

**Example: Pricing a European Call Option by Monte Carlo Simulation**

We price a one-year maturity, at-the-money European call option with the current asset price at 100 and volatility of 20 per cent. The continuously compounded interest rate is assumed to be 6 per cent per annum, the asset pays a continuous dividend yield of 3 per cent per annum. The simulation has 10 time steps and 100 simulations; $K = 100$, $T = 1$
MONTE CARLO SIMULATION

FIGURE 4.2 Pseudo-code for Monte Carlo Valuation of a European Call Option in a Black–Scholes World

```plaintext
initialise_parameters { K, T, S, sig, r, div, N, M }

{ precompute constants }

dt = T/N
nu = (r - div - 0.5 * sig^2) * dt
sigdt = sig * sqrt(dt)
lnS = ln(S)
sum_CT = 0
sum_CT2 = 0

for j = 1 to M do { for each simulation }

lnSt = lnS

for i = 1 to N do { for each time step }

  e = standard_normal_sample
  lnSt = lnSt + nu * dt + sigdt * e { evolve the stock price }
  next i

  ST = exp(lnSt)
  CT = max( 0, ST - K )
  sum_CT = sum_CT + CT
  sum_CT2 = sum_CT2 + CT * CT

next j

call_value = sum_CT / M * exp(-r * T)
SD = sqrt( (sum_CT2 - sum_CT * sum_CT / M) * exp(-2 * r * T) / (M - 1) )
SE = SD / sqrt(M)
```

year, S = 100, σ = 0.2, r = 0.06, δ = 0.03, N = 10, M = 100. Figure 4.3 illustrates the numerical results, the simulated paths of ln(S_t)(i = 1,...,10) are only shown for j = 1,...,5 and j = 95,...,100. The corresponding standard normal random numbers ε are shown in the table below the table of ln(S_t) values in Figure 4.3.

Firstly, the constants; Δt (dt), νΔ(t(nu)*dt), σ√Δt(siga), and ln(S)(lnS) are precomputed:

\[ Δt = \frac{T}{N} = \frac{1}{10} = 0.1 \]

\[ nu = (r - δ - \frac{1}{2} σ^2)Δt = (0.06 - 0.03 - 0.5 \times 0.2^2) \times 0.1 = 0.001 \]

\[ sig = σ\sqrt{Δt} = 0.2\sqrt{0.1} = 0.0632 \]

\[ lnS = ln(S) - X cosθ \]

Simulation
### Figure 4.3 Numerical Example for Monte Carlo Valuation of a European Call Option in a Black–Scholes World

<table>
<thead>
<tr>
<th>K</th>
<th>T</th>
<th>S</th>
<th>sig</th>
<th>r</th>
<th>div</th>
<th>N</th>
<th>M</th>
<th>sum_CT</th>
<th>sum_CT2</th>
<th>ED</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>1</td>
<td>100</td>
<td>0.2</td>
<td>0.06</td>
<td>0.03</td>
<td>10</td>
<td>100</td>
<td>996.49</td>
<td>26610.7</td>
<td>12.22467</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>dt</th>
<th>nrdt</th>
<th>sigstd</th>
<th>lnS</th>
<th>call_value</th>
<th>SE</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>0.0010</td>
<td>0.0832</td>
<td>4.6562</td>
<td>5.3849</td>
<td>1.2225</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>lnSt</th>
<th>j</th>
<th>i</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>ST</th>
<th>CT</th>
<th>CTCT</th>
</tr>
</thead>
</table>

|      | 99 | 4.6052 | 4.5634 | 4.5947 | 4.6552 | 4.6864 | 4.5528 | 4.5570 | 4.5953 | 4.5915 | 4.5915 | 4.6065 | 91.98 | 0.0000 | 0.00 |

<table>
<thead>
<tr>
<th>e</th>
<th>j</th>
<th>i</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td></td>
<td>-0.0497</td>
<td>0.3425</td>
<td>0.7442</td>
<td>-0.3723</td>
<td>0.2277</td>
<td>-1.6708</td>
<td>0.3709</td>
<td>-0.5581</td>
<td>-1.0924</td>
<td>2.6422</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td></td>
<td>1.2660</td>
<td>-0.1946</td>
<td>-2.1717</td>
<td>-1.2990</td>
<td>-0.3266</td>
<td>0.5444</td>
<td>0.4699</td>
<td>-0.3771</td>
<td>-0.5831</td>
<td>0.5763</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td></td>
<td>0.5618</td>
<td>-0.4677</td>
<td>0.0476</td>
<td>0.9110</td>
<td>0.7402</td>
<td>0.4104</td>
<td>-0.1541</td>
<td>0.1510</td>
<td>-0.3833</td>
<td>1.1032</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td></td>
<td>0.1999</td>
<td>0.1557</td>
<td>-1.8976</td>
<td>0.4551</td>
<td>-0.9496</td>
<td>0.0264</td>
<td>0.2076</td>
<td>0.1225</td>
<td>-0.0347</td>
<td>-0.7592</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td></td>
<td>1.1781</td>
<td>0.2555</td>
<td>-0.2584</td>
<td>0.3978</td>
<td>0.1129</td>
<td>0.8560</td>
<td>0.4924</td>
<td>0.5929</td>
<td>-1.0130</td>
<td>0.5489</td>
</tr>
</tbody>
</table>

|    | 95 | 0.0236 | 0.2533 | 1.0649 | -1.3689 | 1.3267 | -2.1535 | -0.5969 | 2.0413 | 0.3444 | -0.6672 |
|    | 96 | 0.4256 | 0.4411 | 0.8249 | 0.6971 | 0.7571 | 1.1791 | -1.0113 | -0.8750 | 0.1782 | 0.3261 |
|    | 97 | -0.5256 | 2.0068 | -0.5362 | 1.6003 | 0.8954 | 2.0709 | -2.1525 | 0.9144 | 0.3343 | 0.6075 |
|    | 98 | -0.2396 | 0.3012 | 1.2777 | -0.7271 | 1.0051 | 1.4231 | 1.7281 | 1.2042 | -0.0647 | 0.0465 |
|    | 99 | -0.6757 | -0.0463 | -0.8418 | 0.5661 | 0.0544 | 0.5256 | -0.7383 | 0.2339 | -0.2779 | -1.3456 |
|    | 100| -1.3793 | -0.0259 | -0.9368 | 1.0383 | 2.4134 | -0.9981 | -0.3623 | -0.0339 | 1.7945 | 0.2250 |
Then for each time step \( i = 1 \) to \( N \), where \( N = 10 \), \( \ln(S_t) \) is simulated. For example for \( j = 1 \) and \( i = 1 \) (dropping the \( i \) and \( j \) subscripts):

\[
\ln(S_t) = \ln(S_i) + \nu dt + \sigma \text{std} \times \varepsilon
\]

\[
\ln(S_i) = 4.6052 + 0.001 + 0.0632 \times (-0.0497) = 4.6030
\]

At \( i = 10 \)

\[
S_T = \exp(\ln(S_t)) = \exp(4.6521) = 104.81
\]

\[
C_T = \max(0, S_T - K) = \max(0, 104.81 - 100) = 4.8070
\]

The sum of the values of \( C_T \) and the squares of the values of \( C_T \) are accumulated:

\[
\sum_{j=1}^{M} C_{T,j} = 996.488 \text{ (sum.CT)} \quad \text{and} \quad \sum_{j=1}^{M} (C_{T,j})^2 = 26610.7 \text{(sum.CT2)}
\]

The estimate of the option value \( \hat{C}_0 \) (call.value) is then given by

\[
\hat{C}_0 = 996.488/100 \times \exp(-0.06 \times 1) = 9.3846
\]

The standard deviation (SD) is given by

\[
\text{SD} = \sqrt{\frac{\sum_{j=1}^{M} (C_{T,j})^2 - \frac{1}{M} \left( \sum_{j=1}^{M} C_{T,j} \right)^2 \times \exp(-2\sigma T)}{M-1}} = \frac{\sqrt{26610.73 - \frac{1}{100} (996.488)^2 \times \exp(-2 \times 0.06 \times 1)}}{100 - 1} = 12.2246
\]

and so the standard error (SE) is

\[
\text{SE} = \frac{\text{SD}}{\sqrt{M}} = \frac{12.2246}{10} = 1.22246
\]

Unfortunately, in order to get an acceptably accurate estimate of the option price a very large number of simulations has to be performed, typically in the order of millions (\( M > 1000000 \)). This problem can be dealt with by using variance reduction methods. These methods work on exactly the same principle as that of hedging an option position, that is that the pay-off of a hedged portfolio will have a much smaller variability than an unhedged pay-off. This corresponds to the variance (or equivalently standard error) of a simulated hedge portfolio being much smaller than that of the unhedged pay-off. We will stress this interpretation throughout this chapter.

### 4.3 Antithetic Variates and Variance Reduction