Monte Carlo Pricing

April 2020

Binary Options

The following collection of terminal payoffs will be collectively referred to as binary options. The notation in brackets is a short-hand for the option value as determined by the Black-Scholes framework e.g. $BC_{CN,K}$ is the value of a Binary Call (BC) option with a Cash or Nothing (CN) payoff and strike K.

- Cash or Nothing
  - Cash or Nothing Call Option ($BC_{CN,K}$): $P(S_T) = \mathbb{1}(S_T > K)$
  - Cash or Nothing Put Option ($BP_{CN,K}$): $P(S_T) = \mathbb{1}(S_T < K)$

  Put-Call Parity: Holding a CN Call and CN Put with the same strike and time to maturity guarantees a payoff of 1 at expiration and hence the portfolio must be worth $e^{-r(T-t)}$ today.

  $$BC_{CN,K}(S_t, t) + +BP_{CN,K}(S_t, t) = e^{-r(T-t)}$$

- Asset or Nothing
  - Asset or Nothing Call Option ($BC_{AN,K}$): $P(S_T) = S_T \mathbb{1}(S_T > K)$
  - Asset or Nothing Put Option ($BP_{AN,K}$): $P(S_T) = S_T \mathbb{1}(S_T < K)$

  Put-Call Parity: Holding an AN Call and AN Put with the same strike and time to maturity guarantees a payoff of S_T at expiration. Assuming the asset, say stock, has a continuous dividend yield of d, this portfolio must be worth $e^{-d(T-t)} S_t$ today.

  $$BC_{AN,K}(S_t, t) + +BP_{AN,K}(S_t, t) = e^{-d(T-t)} S_t$$

Monte Carlo Pricing

As discussed in the lectures, under certain assumptions, we can estimate the value of a derivative contract on S with terminal payoff function P through the relationship:

$$V(S_t, t) = \mathbb{E}^{\mathbb{Q}}[P(S_T)|\mathcal{F}_t] \tag{1}$$

The contract is assumed to have no early-exercise features here but this is not necessary. This expectation can be approximated by simulating paths of the underlying asset and then forming the corresponding average of the contract payoffs.

Price Process

In the case of a stock, equity index or currency we can model price evolution with a geometric Brownian motion:

$$dS_t = (\mu - d)S_t dt + \sigma S_t dW_t$$

Under the risk-neutral measure this becomes:

$$dS_t = (r - d)S_t dt + \sigma S_t dW_t^{\mathbb{Q}} \tag{2}$$

Through use of Ito's lemma we can solve this SDE to obtain:

$$\begin{aligned} S_T &= S_t e^{(r-d-\sigma^2/2)(T-t)+\sigma(W_T^{\mathbb{Q}} - W_t^{\mathbb{Q}})} \\ &\overset{D}{=} S_t e^{(r-d-\sigma^2/2)(T-t)+\sigma\sqrt{(T-t)}\phi} \end{aligned} \tag{3}$$

The second equality above is in distribution and ϕ has its usual meaning as a standard normal random variable.

Path Generation

Equation (3) above gives an exact form that can be used to sample the asset price at time T. If we are considering options whose values are solely determined by the price of the underlying at expiration then we can directly sample S_T using this relationship and proceed with our expectation estimation. If our contract is path dependent then we would instead split the trajectory of the asset into many discrete steps.

Without loss of generality, we can take $t = 0$ in what follows. Let us split the life of the contract into M time steps of equal length, $\delta t = T/M$, and introduce the notation $t_k := k\,\delta t$. Below I list the three discretisation schemes to be used in this report:

- Lognormal:

$$
\begin{aligned}
S_{t+\delta t} &= S_t e^{(r-d-\sigma^2/2)\delta t+\sigma\sqrt{\delta t}\phi} \\
S_{t_k} &= S_0 e^{(r-d-\sigma^2/2)k\delta t+\sigma\sqrt{\delta t}\sum_{i=1}^{k}\phi_i}
\end{aligned}
$$

- Euler-Maruyama:

$$
\begin{aligned}
S_{t+\delta t} &= S_t(1+(r-d)\delta t+\sigma\sqrt{\delta t}\phi) \\
S_{t_k} &= S_0\prod_{i=1}^{k}(1+(r-d)\delta t+\sigma\sqrt{\delta t}\phi_i)
\end{aligned}
$$

- Milstein:

$$
\begin{aligned}
S_{t+\delta t} &= S_t(1+(r-d)\delta t+\sigma\sqrt{\delta t}\phi+\frac{\sigma^2}{2}(\phi^2-1)\delta t) \\
S_{t_k} &= S_0\prod_{i=1}^{k}(1+(r-d)\delta t+\sigma\sqrt{\delta t}\phi_i+\frac{\sigma^2}{2}(\phi_i^2-1)\delta t)
\end{aligned}
$$

Figures 1 and 2 show a comparison between the three methodologies over a period of 1 year for 12 and 250 time steps, i.e. monthly and daily increments. It can be seen that increasing the sampling frequency decreases the difference between the 3 methods and also that the Milstein method is very close to the Lognormal method. Indeed they agree up to order δt. However, the Euler-Maruyama method can differ more significantly, especially at lower frequencies.

Another point to mention here is that the prices produced by the Euler-Maruyama and Milstein approximations can be negative, which will likely introduce some bias into the payoff calculation. This is not the case for the full lognormal approach. In the case of the Euler-Maruyama scheme, say, if we have

$$
\phi < -(1+(r-d)\delta t)/(\sigma\sqrt{\delta t})
$$

then the next price will have the opposite sign to the current. Examples of cases where this is more likely to cause an issue are when volatility is high or $r-d$, a sort of carry, is negative.

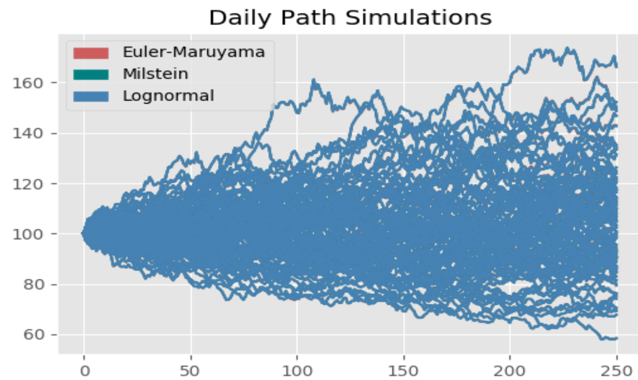

Figure 1: Daily Price Simulations - 250 Time Steps

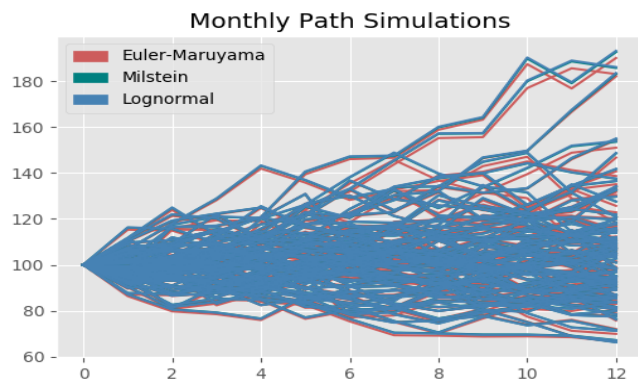

Figure 2: Monthly Price Simulations - 12 Time Steps

Error

The following section is based on [1]. We will estimate the expectation in equation 1 through the formula:

$$\widehat{V}_N := \frac{1}{N}\sum_{i=1}^{N} P(S_T^{(i)}), \tag{4}$$

where $S_T^{(i)}$ denotes the terminal asset value from the i-th path simulation. Since we are drawing independent random numbers, the paths are independent of one another. This means that the random variables $(P(S_T^{(i)}))_i$ are independent and, by construction, identically distributed. It follows that if $P(S_T)$ has finite variance, the Central Limit Theorem (CLT) can be applied to conclude that:

$$\widehat{V}_N \xrightarrow{D} \mathcal{N}\left(\mu, \frac{v^2}{N}\right), \tag{5}$$

where μ and v^2 are the expectation and variance of $P(S_T)$ i.e. $\widehat{V}_N$ is an unbiased estimator and its standard deviation is proportional to $1/\sqrt{N}$. As we are not able to observe v directly, we can instead use the estimate $\hat{v}$ formed from the collection of sample terminal payoffs.

Suppose we want the result of our Monte Carlo pricing to be within some given level of accuracy, ε, of the theoretical contract value with some probability p i.e. we would like $\mathbb{P}(|\widehat{V}_N - V| < \varepsilon) \geq p$. We know that $\frac{\sqrt{N}}{v}(\widehat{V}_N - V) \xrightarrow{D} \mathcal{N}(0,1)$ and hence that:[1]

$$\begin{aligned}\mathbb{P}(|\widehat{V}_N - V| < \varepsilon) &= \mathbb{P}(\frac{\sqrt{N}}{v}|\widehat{V}_N - V| < \frac{\sqrt{N}}{v}\varepsilon) \\ &= \mathbb{P}(|Z| < \frac{\sqrt{N}}{v}\varepsilon) \sim 1 - 2\,\Phi\left(\frac{-\sqrt{N}\varepsilon}{v}\right) \geq p\end{aligned}$$

It follows that to achieve our goal we should take

$$N \geq \hat{v}^2\,\Phi^{-1}((1-p)/2)/\varepsilon^2 \tag{6}$$

First Results

To begin I used an experimental procedure:

- Number of Paths: [100, 1000, 10000, 100000, 1000000]
- Number of Time Steps [1, 5, 10]. For the Euler-Maruyama method, where the number of time steps has a bigger impact, [1, 3, 5, 10, 20, 50]
- Repeat each pricing 50 times to better understand the accuracy of the various methods with different path parameter choices.

I used the three path generation methodologies to price a Cash or Nothing binary call option with the parameters described in the exam outline. The results were

S	K	$T-t$	σ	r
100	100	1	20%	5%

Table 1: Asset Parameters

in line with initial expectations: increasing the number of paths significantly

[1] I use the notation Φ here, instead of N, to denote the CDF of a standard normal random variable to avoid confusion with the path number.

decreased absolute error for all methods. We will see below that the speed of convergence was of the expected order in N, namely $\sim \mathcal{O}(1\sqrt{N})$. The lognormal approach was not noticeably impacted by altering the number of time steps and the same can be said, to a lesser extent, for the Milstein method. The results of the Euler-Maruyama scheme were, however, significantly impacted, producing a consistently biased price when using a low number of time steps.

Accuracy

For convenience, table 2 below gives the theoretical values for vanilla and binary options with the previously discussed set of parameters. The theoretical value

Vanilla Call	Vanilla Put	CN Call	CN Put	AN Call	AN Put
10.45	5.57	0.53	0.42	63.68	36.31

Table 2: Theoretical Option Values

of the Cash or Nothing Call is 0.53 (rounded to 2 decimal places). Table 3 shows the relative error, in percentage points, between Monte Carlo estimates using the Euler-Maruyama and Milstein schemes and this theoretical value. Note that these values were computed by taking the mean of absolute deviations between the Monte Carlo price and the theoretical price over 50 repetitions, i.e. the Monte Carlo process was run 50 times for each combination of number of paths and time steps. The fact that the figures are not strictly decreasing as we proceed right across rows and down columns shows that we may benefit slightly from increasing the number of repetitions beyond 50. However, the general structure looks good and there is a trade-off to be made between accuracy and computation time.

# Time Steps	10^2 Paths	10^3 Paths	10^4 Paths	10^5 Paths	10^6 Paths
1	7.79	7.24	7.02	6.99	6.96
3	7.88	3.07	2.26	2.34	2.32
5	6.49	2.57	1.54	1.40	1.38
10	8.23	1.99	1.04	0.71	0.68
20	7.71	2.34	0.79	0.43	0.33
50	6.33	2.22	0.61	0.21	0.13

Table 3: Euler-Maruyama Mean Relative Error [%]

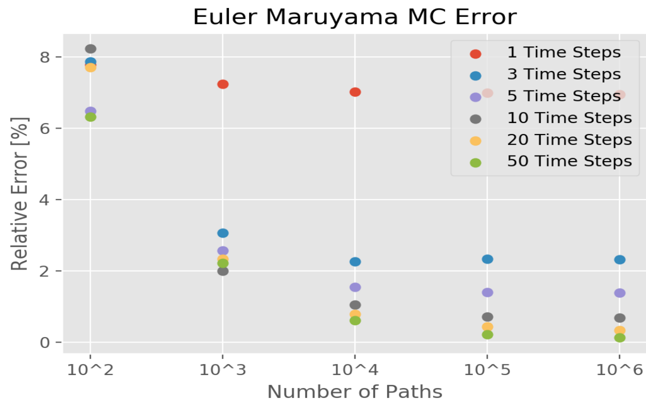

Figure 3: Euler-Maruyama - Mean Relative Error [%]

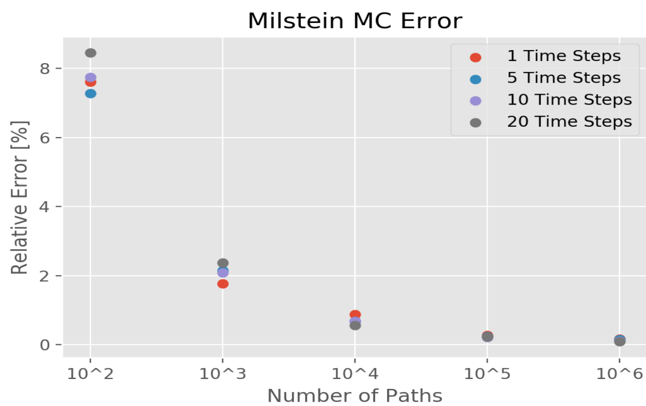

Figure 4: Milstein - Mean Relative Error [%]

Variation of Estimate

Figure 5 shows how the logarithm, taken to base 10, of the standard deviation of absolute error falls very close to lying on a line of slope -0.5 when plotted against path number, as predicted by the $\mathcal{O}(1/\sqrt{N})$ relationship mentioned above. For the lognormal and Milstein schemes we see the same behaviour.

One point of consideration is how accurate we want to be in determining the option value through the Monte Carlo process. If we can assume that the option's tick size is 0.01, which is $\sim 1.88\%$ of the option's value, then we may be happy with an accuracy of, say, 1%. The below table uses the result derived in equation 6 to compute the required number of paths to meet a certain accuracy with 99% probability. The standard deviation of the payoff was estimated to be

Accuracy (%)	Required Number of Paths	Cash Accuracy
2%	4,087	0.0106
1%	16,346	0.0053
0.5%	65,383	0.0027
0.25%	261,531	0.0013
0.10%	1,634,564	0.0005

Table 4: Required Number of Paths for Given Accuracy

~ 0.496, using 10^6 paths of an Euler-Maruyama scheme, which seems intuitive for an ATM contract that pays 0 or 1. These approximate figures could be used to guide the number of paths used, depending on our purpose. Going forward I will use 10^5 paths. Figures 6 and 7 shows how the running price estimate

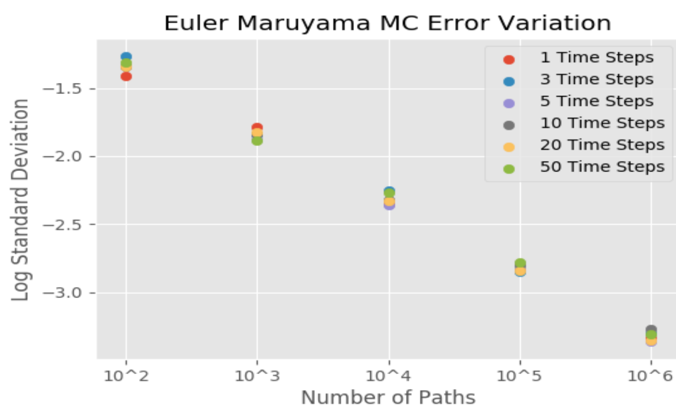

Figure 5: Euler-Maruyama - Log Standard Deviation

converges as the number of sample paths increases in the particular case of the Euler-Maruyama scheme. We clearly identify a bias of overestimation for this particular contract when the step size is large.

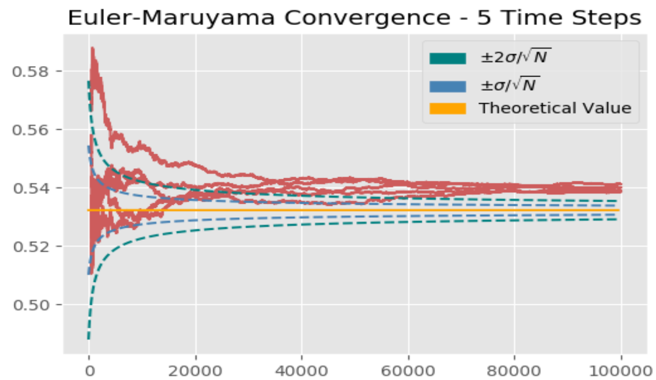

Figure 6: Euler-Maruyama Convergence - 5 Time Steps

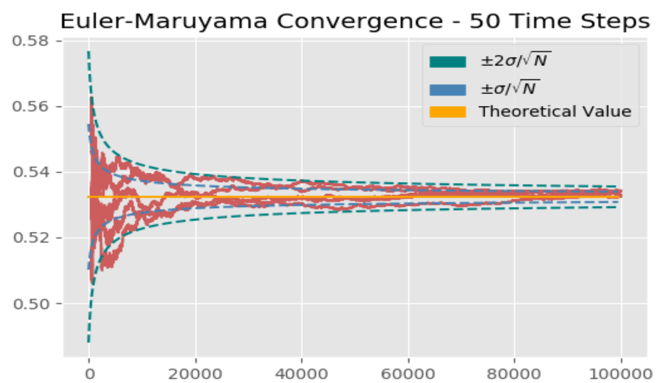

Figure 7: Euler-Maruyama Convergence - 50 Time Steps

Impact of Varying Parameters on Performance

Moneyness

In this section I will examine the impact of changing the 'moneyness', defined here as the ratio S/K, of the option on our ability to approximate its value through Monte Carlo methods. Running the same analysis as above on absolute errors and the standard deviation of the payoff function over a cross-section of asset values shows a clear uni-modal distribution centred on the strike and decaying as the price moves away from the strike, i.e. it is more difficult to accurately price this option close to the strike. Figure 8 shows this result for the standard deviation. For mean error the picture is very similar, but re-scaled.

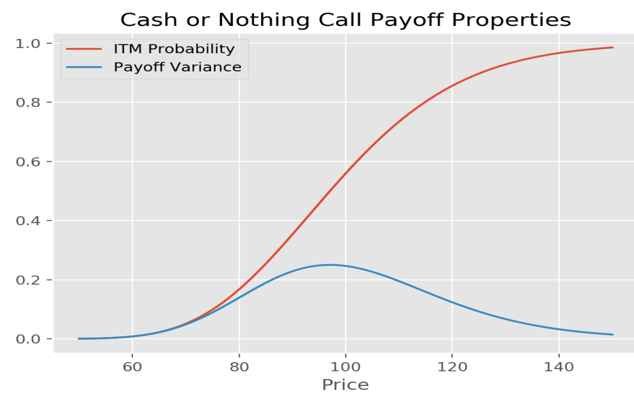

Figure 9: Payoff Standard Deviation By Moneyness

For a cash or nothing option we can calculate the theoretical payoff variance as follows:

$$\begin{aligned}\mathbb{V}[\mathbb{1}_{(S_T>K)}] &= \mathbb{E}[\mathbb{1}_{(S_T>K)}] - (\mathbb{E}[\mathbb{1}_{(S_T>K)}])^2 \\ &= \mathbb{P}(S_T > K)(1 - \mathbb{P}(S_T > K))\end{aligned}$$

Figure 9 shows the theoretical value of this payoff variance plotted against probability of finishing in the money (ITM). This is maximal when the probability of finishing ITM is 50%, which happens at S slightly below 100, i.e. when there is maximal entropy or uncertainty about the payoff.

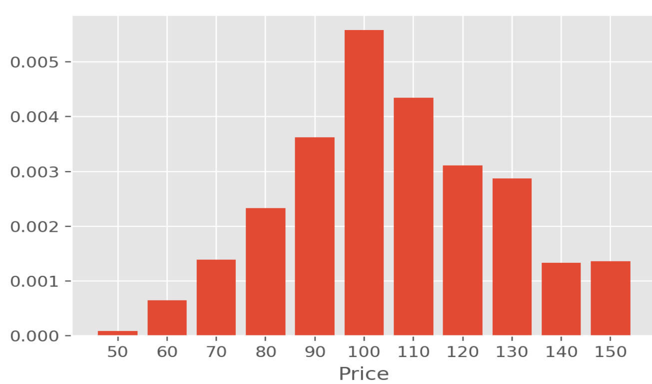

Figure 8: Payoff Standard Deviation By Moneyness

Volatility

The figure below again plots the sample standard deviation of payoffs when the level of asset volatility is shifted for deep OTM, ATM and deep ITM options i.e. with S_t at $60, 100$ and 140, respectively. This one figure is not conclusive, but there does seem to be an increase in payoff uncertainty, and hence more difficulty for pricing using Monte Carlo methods, as the level of volatility increases. This seems to be particularly the case for the OTM option. A possible explanation is that few simulation paths finish ITM, so the price is not altered greatly as volatility increases, but due to the squared nature of variance, those paths that do make it ITM have a disproportionate impact on the payoff uncertainty.

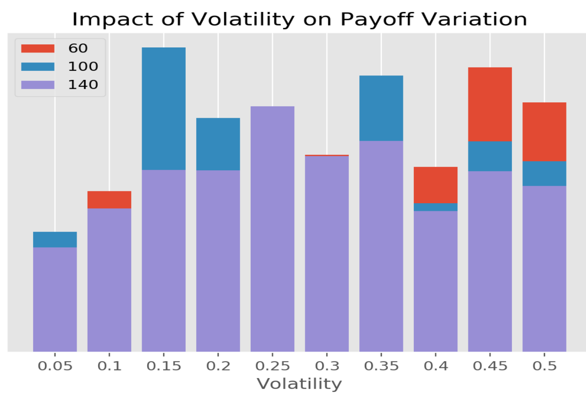

Figure 10: Payoff Standard Deviation By Volatility

Conclusion

This report investigated the use of Monte Carlo methods to price a cash-or-nothing binary call option using the Euler-Maruyama, Milstein and Lognormal schemes. All methods were able to price the option accurately, although time-step size was a concern for the Euler-Maruyama method. As expected, the lognormal approach was generally superior to the Milstein approach, which again was superior to the Euler-Maruyama approach. Theoretical bounds on the number of paths required to meet certain accuracy requirements were provided and empirical calculations to produce concrete numbers were performed. I began to look at the impact varying certain parameters has on our ability to price this particular option. With more time, it would be interesting to look into the related idea of importance sampling and to also examine the impact of time step when pricing a strongly path-dependent option.

References

[1] Peter Jäckel. *Monte Carlo methods in finance*, volume 71. J. Wiley, 2002.