Enhanced Monte Carlo Simulation for Pricing and Hedging Barrier Options

Jiao, Baocheng; Guo, Youli; Wang, Tianhe; Wu, Jianhao

Introduction

Monte Carlo simulation is common for pricing options and assessing their price sensitivities. However, it is susceptible to two drawbacks: slow convergence rates and the time discretization. The latter problem usually leads to extra bias when simulating path-dependent processes such as Barrier options pricing.

Barrier options are a type of financial derivatives whose payoffs hinge on whether the underlying asset's price reaches a predetermined barrier level during the option's duration. They are typically classified as either knock-in or knock-out options. A knock-out option comes into effect only when the underlying asset's price hits the barrier, while a knock-in option is activated in the inverse situation. These options are appealing for their lower costs and accurate price estimation is hence of practical importance.

In our project, we addressed the bias resulting from time discretization in Barrier options pricing by leveraging the reflection principle. We then applied the insights gained from the previous section, combined with smoothing techniques, to analyze the Delta hedge of Barrier options.

Data and Assumptions

We used data of vanilla options based on SPY US Equity to design barrier options from Bloomberg, including:

- Initial price of the equity

- Expected rate of return
- Expected dividend rate
- Vanilla options price
- Implied volatility
- Risk-free interest rate
- Tenor of vanilla options

We assumed the price evolution of SPY US Equity follows the stochastic differential equation:

$$dS(t) = (\mu - q)S(t)dt + \sigma S(t)dW(t),$$

where $S(t)$ is the price of underlying asset at time t, μ is the estimated growth rate, q is the dividend rate, σ is the implied volatility and $W(t)$ is standard Brownian motion. Define the time interval [0, T] and fix a time step Δt the Euler approximation of the price dynamics can be written as:

$$S(t_{i+1}) = S(t_i) + rS(t_i)\Delta t + \sigma S(t_i)\sqrt{\Delta t}N(t_{i+1}),$$

where $N(t_{i+1})$ is a randomly generated standard normal variable.

Discretization Error

The pricing of Barrier option is similar to the pricing of vanilla options with 1 additional barrier constraint. We denoted the first time that the underlying asset price hits the barrier as:

$$\tau_B = \inf\{t > 0: S(t) > B\},$$

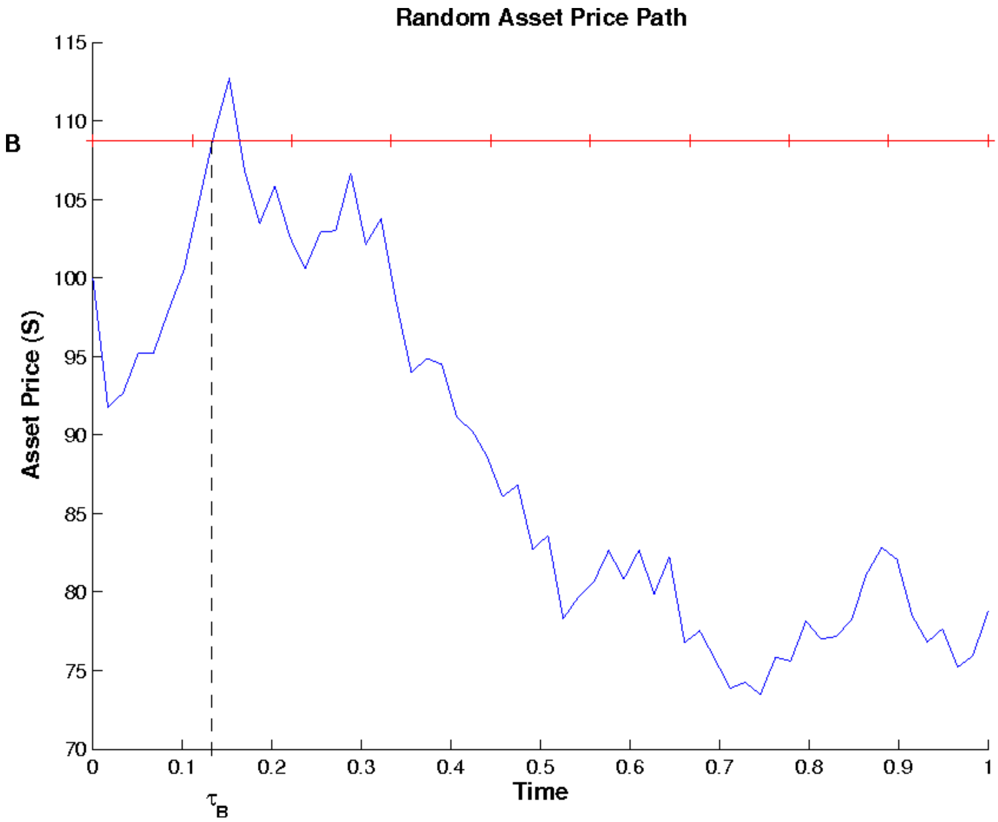
where B is the predetermined barrier value. If the asset does not reach the barrier before expiry, $\tau_B = \infty$. The value of the Barrier option obtained via Mote Carlo simulation at time t can thus be defined as:

$$Knock - In: V_{Knock-In}(t) = e^{-(T-t)r}E\left(V_{vanilla}I_{\{\tau_B \leq T\}}\right)$$

$$Knock - Out: V_{Knock-Out}(t) = e^{-(T-t)r}E\left(V_{vanilla}I_{\{\tau_B > T\}}\right)$$

where r is the risk-free interest rate, T is the expiry time and $V_{vanilla}$ are the simulated payoffs at expiry of vanilla options without a Barrier. Moreover, $I_{\{\tau_B \leq T\}}$ and $I_{\{\tau_B \leq T\}}$ are Heaviside functions:

$$I_{\{\tau_B \leq T\}} = \begin{cases} 1, & \tau_B \leq T \\ 0, & \tau_B > T \end{cases}; I_{\{\tau_B > T\}} = \begin{cases} 1, & \tau_B > T \\ 0, & \tau_B \leq T \end{cases}$$



Single Euler approximation of price evolution with Barrier value and hitting time.

When analyzing Euler approximation steps, if $(S(t_i) - B)(S(t_{i+1}) - B) \leq 0$, we could confirm that at least 1 barrier touching happened between t_i and t_{i+1}; otherwise, it is generally considered that no hit occurred. However, neglecting such instances often introduces additional errors as discrete simulations do not provide complete information. Although the absence of a barrier crossing between adjacent steps may indicate no hit, it does not guarantee that the unaccounted price change between the two steps did not intersect the barrier. Therefore, it is more appropriate to measure the probability of hitting the barrier between two adjacent steps rather than making assertions based solely on the two endpoints.

Correction Process

By assumption, $S(t)$ is log-normally distributed. For convenience, denote $Z(t) = \ln(S(t))$ and $b = \ln(B)$, then:

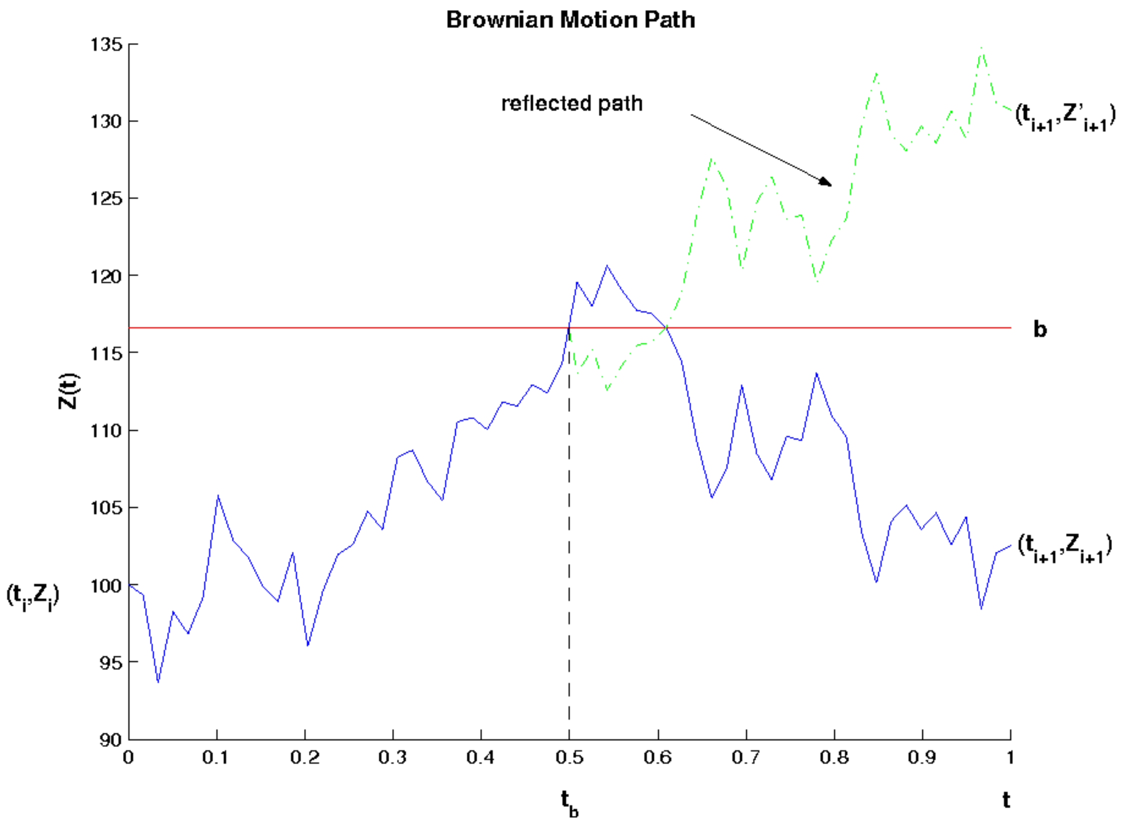
$$dZ(t) = dln(S(t)) = \left(\mu - q - \frac{\sigma^2}{2}\right) + \sigma dW(t) \text{ and } \tau_B = \tau_b$$

Denote $p(Z(t_i), Z(t_{i+1}))$ as the probability that the process starts with $Z(t_i)$ and ends with $Z(t_{i+1})$, then:

$$p(Z(t_i), Z(t_{i+1})) = \frac{1}{\sigma\sqrt{2\pi\Delta t}} \exp\left(-\frac{(Z(t_{i+1}) - Z(t_i) - a\Delta t)^2}{2\sigma^2\Delta t}\right),$$

where a is the drift $\mu - q - \frac{\sigma^2}{2}$. And by the reflection principle, if the hitting happens between $Z(t_i)$ and $Z(t_{i+1})$, we may deduce:

$$p(t_i \le \tau_b \le t_{i+1}, Z(t_i), Z(t_{i+1})) = p(t_i \le \tau_b \le t_{i+1} Z(t_i), 2b - Z(t_{i+1}))$$



Reflection principle application when hitting time happened.

Therefore, $p\left(t_i \le \tau_b \le t_{i+1}, Z(t_i), Z(t_{i+1})\right)$

$$= \frac{e^{\frac{2a\left(b-Z(t_i)\right)}{\sigma^2}}}{\sigma\sqrt{\Delta t}} \Phi\left(-\frac{\left(-2b - Z(t_{i+1}) - Z(t_{i+1}) + a\Delta t\right)}{\sigma\sqrt{\Delta t}}\right),$$

where $\Phi(\mathrm{x})$ is the density function of normal distribution.

The deduction for probability of hitting the barrier between two adjacent steps is hence accomplished:

$$p\left(t_i \le \tau_b \le t_{i+1} \middle| Z(t_i), Z(t_{i+1})\right) = \frac{p\left(t_i \le \tau_b \le t_{i+1}, Z(t_i), Z(t_{i+1})\right)}{p\left(Z(t_i), Z(t_{i+1})\right)}$$

$$= \exp\left(-\frac{2\left(b - Z(t_{i+1})\right)\left(b - Z(t_i)\right)}{\sigma^2 \Delta t}\right),$$

$$p\left(t_i \le \tau_B \le t_{i+1} \middle| S(t_i), S(t_{i+1})\right)$$

$$= \exp\left(-\frac{2\left(ln\left(S(t_{i+1})\right) - ln(B)\right)\left(ln\left(S(t_i)\right) - ln(B)\right)}{\sigma^2 \Delta t}\right)$$

These values also need further control in case they are higher than 1, which does not make sense in reality.

Denote N as the total step in single simulation process. Then $t_N = T$, and the probabilities that hitting does not happens and happens during single simulation are:

$$P\{t_B > T | S(t_0), S(t_1) \dots S(t_N)\} = \Pi_{i=1}^{N-1}\left(1 - p\left(t_i \le \tau_B \le t_{i+1} \middle| S(t_i), S(t_{i+1})\right)\right)$$

$$= \Pi_{i=1}^{N-1}\left(1 - \exp\left(-\frac{2\left(ln\left(S(t_{i+1})\right) - ln(B)\right)\left(ln\left(S(t_i)\right) - ln(B)\right)}{\sigma^2 \Delta t}\right)\right)^+,$$

$$P\{t_B \le T | S(t_0), S(t_1) \dots S(t_N)\} = 1 - P\{t_B > T | S(t_0), S(t_1) \dots S(t_N)\}$$

Utilize the conclusion above, the corrected pricing of Barrier option via simulation is accomplished:

$$Knock - In: V_{knock-in}(t) = e^{-(T-t)r} E\left(V_{vanilla} E\left(I_{\{\tau_B \le T\}} | S(t_0), S(t_1) \dots S(t_N)\right)\right)$$

$$= e^{-(T-t)r} E(V_{vanilla} P\{t_B \le T | S(t_0), S(t_1) \dots S(t_N)\});$$

$$Knock - Out: V_{knock-out}(t) = e^{-(T-t)r} E(V_{vanilla} P\{t_B > T | S(t_0), S(t_1) \dots S(t_N)\})$$

Verification

The real-world prices of the exotic options are usually unavailable. However, theoretically:

$$V_{vanilla}(t) = V_{knock-in}(t, B) + V_{knock-out}(t, B),$$

where t is the tenor and B is the barrier. Thus, we measured the pricing error by fixing a single tenor and changing several barriers:

$$Errors = \sqrt{\frac{\sum_{i=1}^{m}\left(V_{vanilla}(t) - V'_{vanilla}(t, B_i)\right)^2}{m}},$$

where $V'_{vanilla}(t, B) = V_{knock-in}(t, B) + V_{knock-out}(t, B)$ and $V_{vanilla}(t)$ is the real-world vanilla options' price.

We did not modify the strike price; all the options are designed nearly at the money.

Strike: 505	Errors Measurement							
Initial Asset Price: 503.49	Barriers: 510, 530, 550, 600, 650				Barriers: 490, 470, 450, 400, 350			
	Up				Down			
	Call		Put		Call		Put	
Tenors(days)	Crude	Corrected	Crude	Corrected	Crude	Corrected	Crude	Corrected
3	1.438	0.693	65.788	0.429	1.697	0.681	9.515	0.511
14	4.487	0.457	146.377	0.193	5.991	0.371	65.207	0.406
31	6.751	0.451	179.165	0.698	8.889	0.556	102.173	0.371
45	8.252	0.563	202.463	0.579	11.186	0.766	130.235	0.484
66	9.42	0.691	222.847	0.202	13.296	0.731	154.54	0.511
73	9.49	0.386	223.416	0.895	15.214	0.946	155.357	0.338
94	9.831	0.983	238.75	0.943	15.27	1.288	171.931	0.646
106	9.428	0.696	244.064	0.952	17.998	0.606	175.667	0.427
122	11.232	0.835	253.462	0.918	19.459	0.967	187.5	0.863
136	12.303	0.877	258.839	0.763	19.691	1.388	189.586	1.379
Mean	8.586	0.978	214.668	0.682	14.072	1.072	146.796	0.659

Options based on SPY equity with start date 4/16/2024.

Based on the pricing results from our experiment, it can be concluded that the correction method significantly enhances the accuracy of pricing Barrier options using Monte Carlo simulation, especially for the put options.

Delta

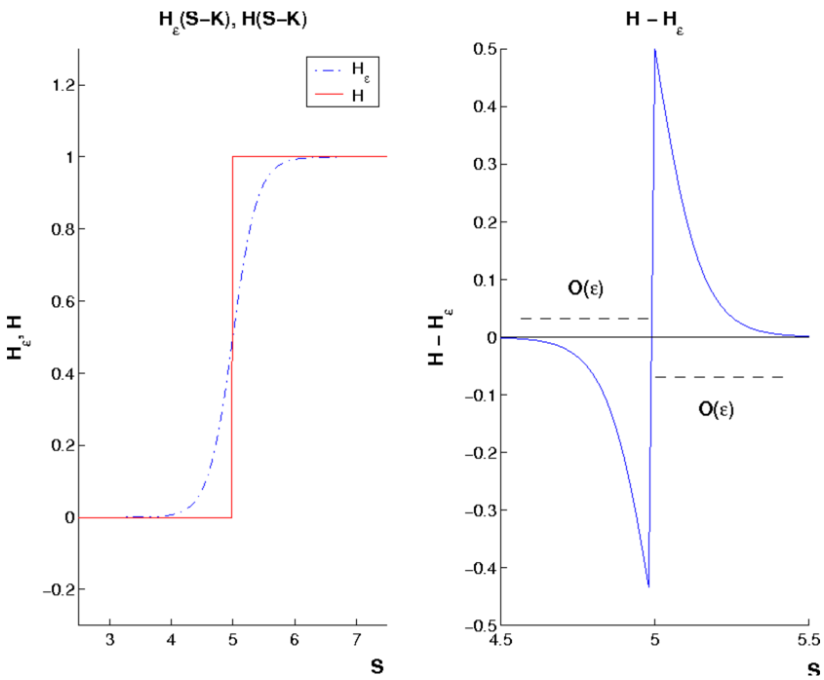
We used another method other than finite difference to obtain the Delta of Barrier options. For simplicity, we introduced this method in the case of down and out call options only as an example. Barrier options are vanilla options with Barrier constraint. Hence, using Monte Carlo simulation, its price at time 0 can be written as:

$$V(0) = e^{-rT} E\big(H(S_{min} - B) R(S(T) - K)\big),$$

where S_{min} is the absolute minimum value during the price evolution process. Moreover, $R(x) = \int_{\infty}^{x} H(x)\,dx$, where $H(x)$ is the Heaviside function:

$$H(x) = \begin{cases} 1, & x > 0 \\ 0, & x \le 0 \end{cases}$$

It is clear that $H(x)$ does not have a derivative at $x = 0$. Therefore, we used the smoothing technique and selected $H_\epsilon(x) = \frac{\tanh\left(\frac{x}{\epsilon}\right)+1}{2}$ as the smooth function. With lower ϵ, the closer $H_\epsilon(x)$ to $H(x)$.



Smoothed and true payoffs when K=5.

Then $H_\epsilon(S_{min} - B)R_\epsilon(S(T) - K)$ is continuous everywhere, therefore:

$$\Delta_\epsilon = \frac{dV(0)}{dS(0)} = e^{-rT}E\left(\frac{d\big(H_\epsilon(S_{min} - B)R_\epsilon(S(T) - K)\big)}{dS(0)}\right)$$

$$= e^{-rT}\int_0^\infty\int_0^\infty e^{-rT}\frac{d\Big(\big(H_\epsilon(S_{min} - B)R_\epsilon(S(T) - K)\big)\Big)}{dS(0)}p\big(S_{min}, S(T)\big)dS(T)dS_{min},$$

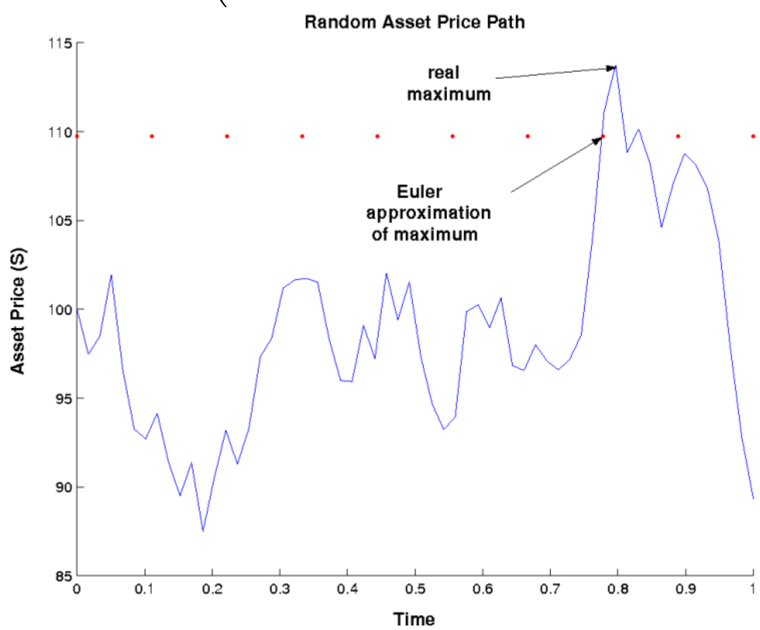
where $p\big(S_{min}, S(T)\big)$ is the joint probability density of S_{min} and $S(T)$.

To obtain the result numerically, we utilized a method that simplified the joint density function, which is related to the true extrema on the simulation path.

True Extrema

We calculated $p\big(t_i \le \tau_B \le t_{i+1} \big| S(t_i), S(t_{i+1})\big)$ in the pricing section. This formula can also be applied to calculate the probabilities associated with the distribution of the true extrema. Denote S_{min}^i as the true minimum in $[S(t_i), S(t_{i+1})]$ and τ_{min} as the first time that the price hits its local extrema over $[t_i, t_{i+1}]$, then this probability can be written as:

$$p\big(t_i \le \tau_{min} \le t_{i+1} \big| S(t_i), S(t_{i+1})\big)$$
$$= \exp\left(-\frac{2\big(ln\big(S(t_{i+1})\big) - ln\big(S_{min}^i\big)\big)\big(ln\big(S(t_i)\big) - ln\big(S_{min}^i\big)\big)}{\sigma^2 T}\right)$$



If we replace $S(t_i)$ and $S(t_{i+1})$ with $S(0)$ and $S(T)$, then this probability is related to the absolute minimum:

$$p\left(t_i \leq \tau_{min} \leq t_{i+1} \middle| S(0), S(T)\right) = \exp\left(-\frac{2\left(ln\left(S(T)\right) - ln(S_{min})\right)\left(ln\left(S(0)\right) - ln(S_{min})\right)}{\sigma^2 T}\right)$$

Replacement

We have obtained the probability regarding S_{min}. Denote U_1 as uniformly distributed variable defined on [0, 1], then we could gain the value of S_{min} with the respect of U_1. Let U_1 replace $p\left(t_i \leq \tau_{min} \leq t_{i+1} \middle| S(0), S(T)\right)$, we could gain:

$$S_{min} = \exp\left(\frac{1}{2}\left(ln\left(S(0)\right) + ln\left(S(T)\right) - \sqrt{\left(ln\left(S(0)\right) - ln\left(S(T)\right)\right)^2 - 2\sigma^2 T ln(U_1)}\right)\right)$$

Moreover, $S(T) = S(0)\exp\left(\left(\mu - q - \frac{\sigma^2}{2}\right)T + \sigma\sqrt{T}\Phi^{-1}(\mathrm{x})\right)$, where $\Phi^{-1}(x)$ is the inverse cumulated distribution function of normal distribution. Denote and U_2 as another independent uniformly distributed variable defined on [0, 1]. Replace $\Phi^{-1}(\mathrm{x})$ with $\Phi^{-1}(U_2)$, then:

$$S(T) = S(0)\exp\left(\left(\mu - q - \frac{\sigma^2}{2}\right)T + \sigma\sqrt{T}\Phi^{-1}(U_1)\right)$$

Furthermore, as $p(U_1, U_2) = 1$:

$$\Delta_\epsilon = e^{-rT}\int_0^1\int_0^1 \frac{d\left(H(S_{min} - B)R(S(T) - K)\right)}{dS(0)} dU_1 dU_2$$

Theoretically, U_1 and U_2 can take all the values in [0, 1]. However, as they are reflected by $\Phi^{-1}(x)$ and $ln(x)$, we constrained their values in (0, 1). Furthermore, $\left(ln\left(S(0)\right) - ln\left(S(T)\right)\right)^2 - 2\sigma^2 T ln(U_2)$ could be smaller than 0, we finally constrained its value be positive.

Hedging

We used our method to obtain Deltas and compared the hedging results with the Black-Scholes Delta hedge. If it is knock-in option, we do not enter the equity market unless it hits the barrier, then we hold the shares of equity indicated by Delta; otherwise, we first hold the shares of equity indicated by Delta and quit the equity market if it hits the barrier. We assessed the deviation between the portfolio's value and the terminal values of the options using absolute error.

	Initial Price: 514.865	End Price: 503.485	Strike: 512				
	Barrier	Rule	Crude Call	Corrected Call	Crude Put	Corrected Put	Hit the Barrier?
Up	517	In	3.067	0.644	7.417	5.346	Yes
		Out	2.934	0.047	11.681	10.374	
	550	In	0	0	1.020	1.020	No
		Out	9.160	0.525	3.863	3.858	
Down	507	In	9.160	1.225	3.923	3.858	Yes
		Out	0	0	1.020	1.020	
	460	In	9.160	0	8.588	3.858	No
		Out	0	0	1.020	1.020	

Generally, the enhanced Delta is a better indicator. Both Deltas do better at hedging call options, the reason could be that the put option in the scenario we chose will make money whether there is a barrier or not. The experiment is constrained by the frequency of information, so the error is greater when the option makes a profit in the last day.

Conclusion

Monte Carlo simulation method, while valuable, inherently lacks the granularity necessary to fully capture continuous processes. In our project, we refined this approach for pricing Barrier options by introducing a method to estimate the probabilities of the evolution process reaching the barrier at each simulation step. Unlike conventional methods that solely rely on crossing points to determine hitting times, our modification significantly enhances effectiveness. Furthermore, it offers valuable insights into accurately measuring the Delta of Barrier options. This improvement not only enhances the precision of pricing but also contributes to a deeper understanding of the underlying dynamics.

References

Justino, D. (2010) *Hedging of Barrier Options.* University Institute of Lisbon.

Kelliher, C. (2022) *Quantative Finance with Python.* New York: CRC Press.

Papatheodorou, B. (2005) *Enhanced Monte Carlo Methods for Pricing and Hedging Exotic Options.* University of Oxford.

Shreve, S. (2000) *Stochastic Calculus for Finance I: The Binomial Asset Pricing Model.* Springer Finance.

Shreve, S. (2000) *Stochastic Calculus for Finance II: Continuous-Time Models.* Springer Finance.