Design and Risk Management of an S&P 500-Linked Snowball Auto-callable: A Comparative Analysis Using Monte Carlo Simulation and PDE Method

Chenyan Zheng[1,†,*], Hanxi Qin[2,†], Jiani Han[3,†]
{3230104254@zju.edu.cn[1], sqin14@smith.edu[2], jh796@duke.edu[3]}

School of Economics, Zhejiang University, Hangzhou, 310058, China[1]
Mathematical Sciences Department; Economics Department, Smith College, Northampton, MA 01063, US[2]
Applied mathematics and computational science, Duke Kunshan University, Kunshan, 215311, China[3]
*corresponding author
†co-first authors

Abstract. This paper discusses the design of an S&P 500-linked Snowball Auto-callable, which aims to enrich the derivatives market. It is essential to ensure effective risk management in light of the increased complexity of the market during the COVID-19 crisis. Considering that options serve as a key financial instrument for hedging, we evaluate our product using two pricing methods - PDEs and Monte Carlo simulations. Additionally, we analyze internal and external risks, offering both investors and issuers hedge strategies and recommendations.

Keywords: Snowball Autocallable, pricing, Monte Carlo Simulation, PDE, Sensitivity analysis

1 Introduction

In recent years, snowball autocallables have garnered significant attention within the financial industry, not only because they are a novel derivative combining features of options, but also due to their mutually beneficial payoff structure for both brokers and investors. Essentially, snowball autocallables are bizarre options with obstacle terms, allowing investors to collect option premiums. The product's performance is linked to underlying assets such as indices, individual stocks, or commodities, with the triggering of knock-in and knock-out events determined by pre-set barrier levels.

The introduction of snowball autocallables has proven valuable in revitalizing the volatile stock markets, particularly in the wake of the pandemic, providing enhanced benefits for both investors and brokers while increasing market liquidity. This is one of the primary motivations for this study on snowball autocallables. Through a comprehensive literature review, it was found that snowball pricing is generally approached via two methods: the derivation through Partial Differential Equations (PDE) and simulation-based pricing using Monte Carlo methods.

CONF-CIAP 2025, January 17-23, Eskisehir, Turkey

DOI 10.4108/eai.17-1-2025.2355240

This paper will first explain the mathematical derivation of both methods and explore their feasibility in programming implementation, ultimately obtaining the pricing results. By comparing the convergence and differences in the final pricing outcomes, this research aims to verify the accuracy of each method and evaluate the potential areas for improvement. After developing the snowball autocallable product, this research will also conduct thorough risk management assessments. This includes evaluating factors such as knock-in and knock-out levels, market trends, liquidity risk, and the impact of credit risk on the product's performance. And this paper roughly introduces the application of Delta hedging strategy in snowball autocallable risk hedging. By conducting these tests, this research aims to ensure that the application scenarios of the snowball autocallable are more aligned with the dynamics of real-world financial markets. Through this approach, the product can be better positioned to meet the practical demands and risks present in contemporary market environments. Additionally, through the literature review, this research has observed that snowball autocallables are more prevalent in the Chinese market, often linked to large-cap indices. Thus, this research will also explore whether these methods are equally applicable to major indices in the U.S. market.

While many brokers have already developed sophisticated pricing systems for snowball autocallables, often fine-tuning parameters manually for greater pricing precision, this paper seeks to provide a deeper understanding of snowball products for both investors and brokers alike.On the basis of designing the structured financial product to meet the investors and issuers of the product, this paper adopts quantitative and qualitative research methods to price and risk-analyze the product, confirm the feasibility of the product, enrich the product varieties in the current financial market, and provide more choices for investors and issuers.

2 Literature Review

There is no analytical solution for option pricing, so numerical methods such as binomial trees, finite differences, and Monte Carlo simulations are a practical alternative.

Cox, Ross, and Rubinstein introduced the binomial tree method in 1979 for simplifying the pricing process by converting continuous-time problems into discrete-time problems [1]. The algorithm is useful for both European and American options, but when multiple variables are involved, it becomes inefficient, resulting in a slow convergence. Finite difference methods reduce the complexity of differential equations by turning them into discrete algebraic equations that enable faster and more efficient pricing. Schwartz began using the finite difference method to approximate the exact solution of partial differential equations in 1997 [2]. Since then, the application of the finite difference method in finance has been expanded.

Monte Carlo method is a method of stochastic simulation, which is based on probability and statistical theory. This method consists of simulating many different paths and averaging the final value of the option. In 1997, Boyle introduced Monte Carlo simulations as a probabilistic method for estimating the value of European options [3]. Even though it is versatile, large simulations may require a significant amount of computational power.

In summary, the three key numerical methods in option pricing are binomial trees, finite differences, and Monte Carlo. The use of binary trees is simple, but they require a significant amount of computational power. Finite differences are more efficient for complex options and Monte Carlo can be used when handling stochastic scenarios, but they require a considerable amount of computational power.

3 Introduction of the Snowball Auto-Callable and Our Product

3.1 Product Design and Key Elements

This paper discusses the factors that make up a standard Snowball structured auto-callable, including the knock-in and knock-out levels, volatility, as well as risk-free rate. Detailed specifications for the S&P 500 index-linked Annual Auto-Callable Notes due June 1, 2025, have been developed based on the product's profit structure. There is a detailed product overview shown on the left in Figure 1. Additionally, a clear annual yield graph is generated on the right in Figure 1.

Term	**Term note**
Security	Standard Snowball AutoCallable Derivative
Maturity	12M
Underlying Asset	S&P 500 Index
Knock-In Barrier Level	S0*85% (Observation frequency: Daily)
Knock-Out Call Level	S0*103% (Observation frequency: Monthly)
Coupon rate	20%

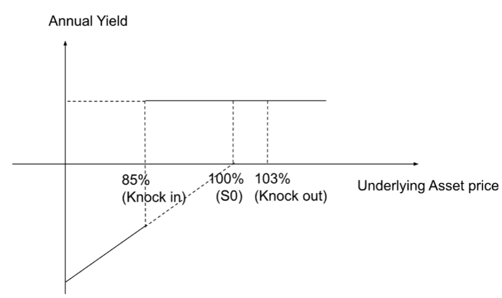


Fig. 1. Product overview(on the left) and Impact of Underlying Asset Price on Annual Yield(on the right)

3.2 payment at maturity

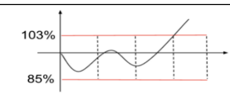
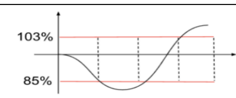
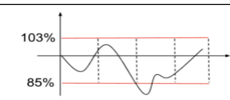
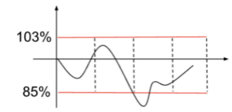
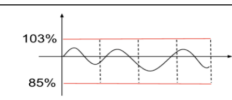
	Scenario		Maturity Payoff	
1	Underlying price knocks out (reach the 103% level)	Price never reach the knock in level and knock out	Principal + coupon (3 weeks)	
2		Price dips below 85%, then bounce back and knock out		
3	Underlying price knocks in (Drop below 85% level)	Underlying price drops below 85% and bounce back (St>S0)	Principal	
4		Underlying price drops below 85% and not bounce back (St<S0)	Nominal principal loss (if St=90%*S0, lose 10%*principal)	
5	Underlying price never drops below 85% and never reaches 103%	Price fluctuates but stays in between 85% and 103%	Principal + coupon (1 month)	

Fig. 2. Payment at maturity of snowball structured derivative

The calculation of the Snowball AutoCallable payment is based on five scenarios listed below in the graph (Figure 2). This section will go through all the scenarios to provide a better understanding of the Snowball interest payment.

1. Scenario 1 (Knock-out but no Knock-in): if the underlying asset knocks out and never knocks in before, then the investors get the coupon payment (that is the coupon interest rate times the principal investor originally paid.

2. Scenario 2 (Knock-out and Knock-in): if the underlying asset knocks in before knock-out, then the investors get the coupon payment as well, which is the same as the Scenario 1

3. Scenario 3 (Knock-in but no Knock-out with Bounce back): if the underlying asset hits the knock-in ground but the asset price at maturity is higher than at the beginning, the investors get the original principle. In this case, investors gain no interest and no loss.

4. Scenario 4 (Knock-in but no Knock-out without Bounce-back): if the underlying asset knocks in and does not knock out until the maturity date, then the investor gets the loss of principal. For example, if the price of the underlying asset drops by 20%, the investor's loss at maturity of the Snowball product will be 20% of the principal.

5. Scenario 5 (No Knock-in and No Knock-out): if the underlying asset does not breach either the knock-out or knock-in conditions, the investor will receive the full principal along with the coupon interest. (Our product does not include dividend coupons, although most products in the market do have such provisions.)

4 Data

4.1 Assumption

The efficient market hypothesis states that The history is fully reflected in the present price, which does not hold any further information; Markets respond immediately to any new information about an asset price [4].

The two assumptions above suggest asset prices change according to a Markov process, which means that the price of an asset will be determined by its current price alone, not by previous prices. This model assumes that underlying asset prices are determined by a stochastic process driven by Brownian motion. A stochastic differential equation (SDE) can be used to model the price dynamics of the underlying asset as follows:

$$dS_t = \mu S_t dt + \sigma S_t dZ_t$$

Here, S_t represents the price of the underlying asset at time t, μ is the expected rate of return, σ is the volatility, and dZ_t is the increment of a standard Brownian motion, representing the random fluctuations in the asset price.

This paper prices the annual snowball auto-callable due June 1, 2025, linked to the S&P 500 Index. According to Black-Scholes, the path-dependent snowball auto-callable assumes the underlying asset follows an SDE [5]:

$$dS_t = \mu S_t dt + \sigma S_t dB_t$$

Where S_t is the stock price at time t, r is the risk-free interest rate, σ is the volatility of the underlying asset, and $dB_t = \varepsilon\sqrt{dt}, \varepsilon \sim N(0,1)$ represents the standard normal distribution.

4.2 Volatility

Volatility describes the degree of volatility of a financial asset and is a measure of the uncertainty of an asset's returns. In general, the higher the volatility, the more volatile the financial assets and the greater the uncertainty of asset returns. As an important factor in the Black-Scholes model option pricing,we choose Garch volatility instead of directly using volatility in the previous year. This indicates the aggregation and tailing of the volatility. We calculate and use volatility of 18%.

4.3 Risk-free Rate

A key characteristic of the Black–Scholes–Merton differential equation is that it is independent of any variables influenced by the risk preferences of investors (Hull, 2012). The only variables present in the equation are the current stock price, time, stock price volatility, and the risk-free rate of interest. The pricing process assumes a risk-neutral framework, in which derivative prices are unaffected by investors' subjective risk preferences. In this setting, the return on all assets is equal to the risk-free rate. As a result, under risk-neutral conditions, all cash flows can be discounted using the risk-free rate.

For this analysis, the risk-free rate is derived from the average 10-year Treasury yield, measured over the period from June 1, 2023, to June 1, 2024, resulting in a risk-free rate of 4.25%.

5 Methodology

5.1 Garch model

Due to the heteroscedasticity and volatility clustering effects of financial product yield sequences, Engle proposed the Autoregressive Conditional Heteroscedasticity (ARCH) model(ENGLE R,1982). [6] The Garch model is an extension of the ARCH model to describe the variance structure in time-series data more accurately by introducing higher-order terms of past variance. This paper uses the Garch(1,1) model to forecast the volatility of S&P 500 in the duration of this product. The formula is as below,

$$\sigma_n^2 = \gamma V_L + \alpha r_{n-1}^2 + \beta \sigma_{n-1}^2$$

in which

$$\gamma + \alpha + \beta = 1$$

We choose the closing price of S&P 500 in the last 10 years and first create a yield chart as follows.

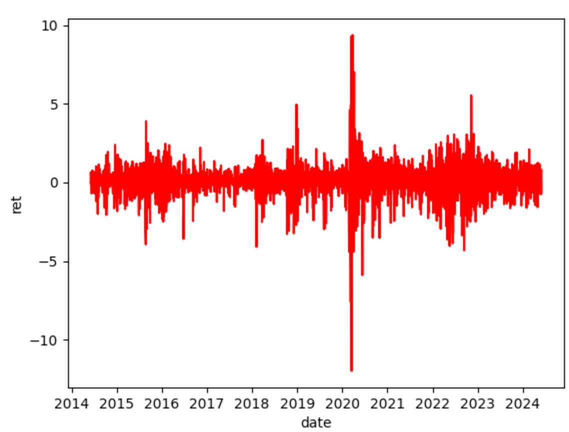


Fig. 3. Yield chart

Figure 3 shows that there is an obvious aggregation effect. By processing data and calculating LB, it can be known that the sequence is not white noise and can be modeled.Finally, we calculate the future volatility-18%.

5.2 PDE Method

5.2.1 Finite Difference Method

Under the Black-Scholes model, the stochastic differential equation (SDE) for the underlying asset (e.g., stock price) is as follows:

$$\frac{dS(t)}{S(t)} = rdt + \sigma dB(t)$$

The Black-Scholes partial differential equation can be derived from Ito's Lemma based on the assumption that there is no arbitrage in the financial market, as shown below [7]:

$$\frac{\partial f}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 f}{\partial S^2} + rS\frac{\partial f}{\partial S} - rf(S,t) = 0$$

The term $f(S(t),t)$ can be interpreted as any portfolio of investments that is determined only by S and t. Although the equation to solve remains the same, boundary conditions vary based on the derivative product's structure. These are the three boundary conditions mentioned here [8]:

$$f(S,t) \quad \text{as} \quad t \to T$$
$$f(S,t) \quad \text{as} \quad S \to 0$$
$$f(S,t) \quad \text{as} \quad S \to \infty$$

Rather than using the numerical method to calculate pricing, the grid search method is needed to calculate the price from back to front. The time and price steps must be selected before the grid can be constructed.

The expiration time is set at $T = 1$ year, which corresponds to 252 trading days, and the time interval is divided into $N = 252$ steps, with each step corresponding to $\Delta t = \frac{T}{N} = \frac{1}{252}$.

The price is divided into $M = 800$ price steps. The maximum asset price $S_{\max}$ is $S_{\max} = 4 * K$, where K is the knock out level. The price step is $\Delta S = \frac{S_{\max}}{M} = 26.39$, which is approximately 0.5% of the initial price. This ensures that the price grid is fine enough to identify the dynamics of the option.

The following (800×252) grid (figure 4) maintains a balance between precision and performance, ensuring that the price of an option is accurately represented over time. Gray points represent neighboring points involved in the spatial and time differencing schemes. Red points represent current function values at specific prices and times. The Crank-Nicolson method uses the information from the gray points to compute the value at the red point, combining both spatial and time differences to update the solution.

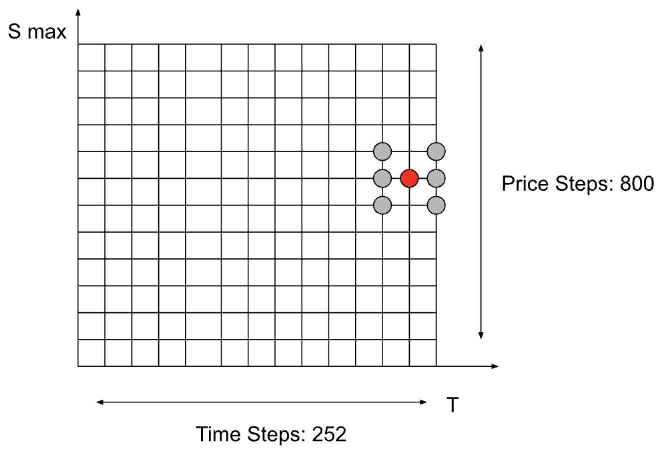


Fig. 4. Grid of PDE

To perform this analysis, it is necessary to obtain all the points on the vertical coordinate at $t = 0$, i.e., the price of the derivative at the price of S of different indices at the time of initial pricing. Accordingly, the PDE method is a method of deriving the initial price backwards from the boundary conditions.

This paper employs the Crank-Nicolson finite difference method, which combines the explicit and implicit time-stepping schemes by averaging them. The finite difference method in both the time dimension and the spatial dimensions, which employs central difference method with a parameter value of $\theta = 0.5$, can be expressed by the following formula [9]:

$$\frac{\partial V([\theta i + (1-\theta)(i+1)], j)}{\partial t} \approx \frac{V(i+1, j) - V(i, j)}{\Delta t}$$

In spatial dimensions:

$$\frac{\partial V}{\partial S} = \frac{\partial V(i, j)}{\partial S} \approx \frac{V(i, j+1) - V(i, j-1)}{2\Delta S}$$

$$\frac{\partial^2 V}{\partial S^2} = \frac{\partial^2 V(i, j)}{\partial S^2} \approx \frac{V(i, j+1) - 2V(i, j) + V(i, j-1)}{(\Delta S)^2}$$

By substituting the finite difference approximations into the PDE, we obtain the following:

$$\frac{V(i+1, j) - V(i, j)}{\Delta t} + rj\frac{V(i, j+1) - V(i, j-1)}{2\Delta S} + \frac{1}{2}\sigma^2 j^2 \frac{V(i, j+1) - 2V(i, j) + V(i, j-1)}{(\Delta S)^2} - rV(i, j) = 0$$

Multiplying by Δt and rearranging terms, we obtain:

$$V(i+1, j) = a_j V(i, j-1) + b_j V(i, j) + c_j V(i, j+1)$$

Where the coefficients are defined as:

$$a_j = -\frac{1}{4}rj\Delta t + \frac{1}{4}\sigma^2 j^2 \Delta t$$

$$b_j = -\frac{r\Delta t}{2} - \frac{\sigma^2 j^2 \Delta t}{2}$$

$$c_j = \frac{1}{4}rj\Delta t + \frac{1}{4}\sigma^2 j^2 \Delta t$$

For all j, we now have the following system of equations:

$$\begin{pmatrix} 1-b_1 & -c_1 & 0 & \cdots & 0 \\ -a_2 & 1-b_2 & -c_2 & \ddots & 0 \\ 0 & -a_3 & 1-b_3 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & -a_{M-1} & 1-b_{M-1} \end{pmatrix} \begin{pmatrix} V(i,1) \\ V(i,2) \\ \vdots \\ V(i,M-2) \\ V(i,M-1) \end{pmatrix}$$

$$= \begin{pmatrix} 1+b_1 & c_1 & 0 & \cdots & 0 \\ a_2 & 1+b_2 & c_2 & \ddots & 0 \\ 0 & a_3 & 1+b_3 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & a_{M-1} & 1+b_{M-1} \end{pmatrix} \begin{pmatrix} V(i+1,1) \\ V(i+1,2) \\ \vdots \\ V(i+1,M-2) \\ V(i+1,M-1) \end{pmatrix} + \begin{pmatrix} a_1(V(i,0)+V(i+1,0)) \\ 0 \\ \vdots \\ 0 \\ c_{M-1}(V(i,M)+V(i+1,M)) \end{pmatrix}$$

The matrix on the left-hand side contains the coefficients a_j, b_j, and c_j from the discretized equation, which denotes as M1. The vector on the left-hand side contains the values of $V(i,j)$, the option prices at time step i, which denotes as b. The vector on the right-hand side contains the option prices at the next time step $i+1$, adjusted for the boundary conditions $V(i,0)$ and $V(i,M)$, which denotes M2. Alternatively, the system can be expressed as follows:

$$M_1 \cdot V_i = M_2 V_{i+1} + b$$

This formulation allows us to solve for V_i at each time step. By iterating backward through time, starting from the final condition, we can eventually determine the value of V at time $t = 0$, which represents the price of the option at the current time.

5.2.2 Boundary condition for each PDE

This paper divides the five scenarios of payment of maturity into three categories.

- One-touch Up(OTU)
  Option that are knocked out directly, or those that are knocked out following a knock-in event, fall into the first category. On an observation date, if the knock-out level is breached, the option terminates. The payoff is $R_1 \times ti$, where R_1 is the one-month coupon payment. At a lower boundary, the asset price reaches 0 resulting in a worthless option. When an option expires without triggering a payout, a right boundary condition applies, and the terminal payoff is zero. Figure 5 showcases the three boundary conditions for OTU [10].

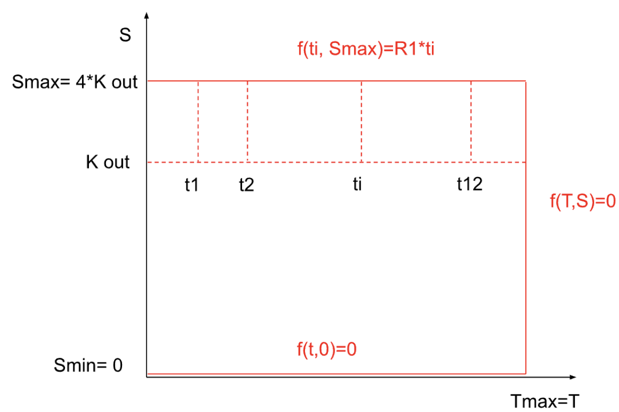


Fig. 5. PDE of OTU

- Double No-touch(DNT)
  A second type of knock-out option involves two barriers, specifically an up-out and a down-out. As long as the asset remains within the knock-in and knock-out levels on the observation date, the holder receives the full coupon payment. In the event the price reaches zero or exceeds S_max, the option is knocked out, and no payout is made. The full coupon is paid if the price stays between the barriers. In the case of S = 0, the option's value is always zero, meaning that it is worthless. Figure 6 showcases the three boundary conditions for DNT [10].

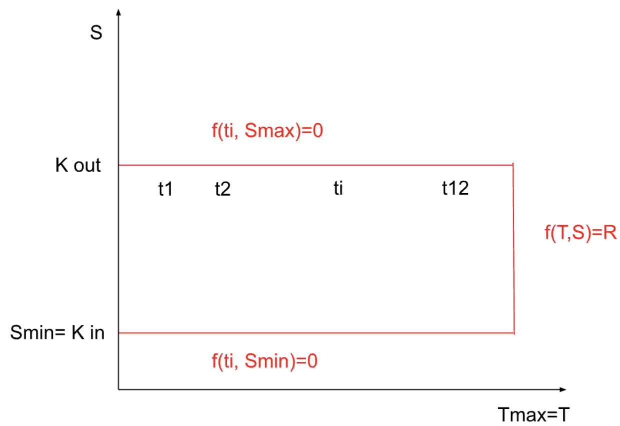


Fig. 6. PDE of DNT

- Double Knock-out Put(DKOP) - Up and Out Put(UOP)
  A third type of knock-in involves an up-out and a down-in, which is equivalent to selling a put option with an up-out and a down-in. When the knock-in component is replaced by standard knock-out options, the pricing is simplified, while maintaining the same payoff and return. A strategy based on this result consists of selling an Up-Out put option while purchasing an Up-Out and Down-Out put option, with the same return as the third category. Boundary conditions are set for both selling the Up-Out Put and buying the Up-Out and Down-Out Put.
  The PDE for the DKOP is similar to the PDE for the second category, with the main difference being that, at maturity, the payoff is that of a put option. On the right is a representation of the PDE for the UOP. When the index is knocked out, the payoff is zero, but if it is not, it becomes the payoff of a put option. The bottom boundary reflects the value of the put option if the asset's price reaches zero, which is $S_0 * e^{-r(T-t)}$. Figure 7 showcases the three boundary conditions for DKOP and UOP [10].

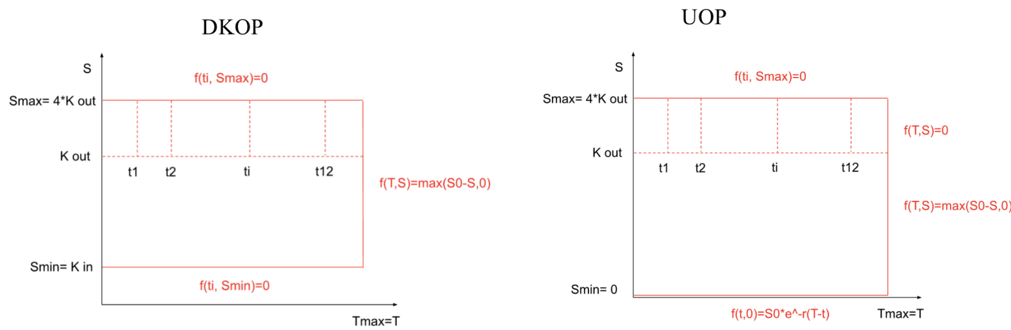


Fig. 7. PDE of DKOP(on the left) and UOP(on the right)

5.3 Monte Carlo Method

5.3.1 Overview

Monte Carlo Simulation (MCS) is extensively employed for pricing financial derivatives globally. Renowned for its simplicity and precision, MCS effectively handles complex pricing scenarios that arise in the financial markets. Unlike the traditional Black-Scholes model, MCS excels in addressing intricate derivative structures, offering a closer approximation of option prices in a more efficient timeframe [11]. In the subsequent section on MCS, this paper will present the mathematical foundations that underscore its applicability, establish parameters to demonstrate its efficacy in pricing Snowball Autocallables, and finally, evaluate the accuracy of the results by assessing convergence and estimating potential errors.

5.3.2 Monte Carlo Introduction

Monte Carlo Simulation refers to the simulation of the independent and random event several times to generate an expected statistical probability. It avoids problems that are too complicated to analyze by pure numerical analysis or mathematical induction. The mathematical proof of the validation for the Monte-Carlo simulation is as follows.

To construct a Monte-Carlo Simulation, it first needs a density function $\psi(x)$. Suppose that there is a set of possible events D. For every $x \in D$, it follows a density function $\psi(x)$. Then, the probability of the event could be formulated as,

$$P_r[x \in D] = \int_D \psi(x)dx,$$

where the sum of all the possibilities of the event should be equal to 1,

$$\int \psi_D(x) f(x) dx = 1.$$

Since already have the density function, the expected value (expectation) of the $f(x)$ concerning $\psi(x)$ could be calculated,

$$\mathbb{E}_{\psi}[f] = \mathbb{E}(f) = \int_D \psi(x) f(x) dx.$$

Also, for the variance, which is based on the calculation of the expectation,

$$\mathbb{V}[f] = \mathbb{E}[(f - \mathbb{E}(f))^2].$$

The variance can also be written as, by the existing theorem,

$$\mathbb{V}[f] = \mathbb{E}[f^2] - \mathbb{E}[f]^2.$$

With the variance and expectation, the covariance and correlation can be expressed as,

$$Cov[f,g] = \mathbb{E}[f,g] - \mathbb{E}[f]\mathbb{E}[g],$$

$$Corr[f,g] = \frac{Cov[f,g]}{\sqrt{\mathbb{V}[f]\mathbb{V}[g]}},$$

where the $Corr[f,g] \in [-1,1]$.

After examining the probability for every independent event, this paper sets the "Running Times" and "Running Sum" of the simulation. The "Running Times" refers to how many times should the simulation process repeat and the "Running Sum" refers to the sum of the result generated by each time of simulation. After achieving the "Running Times", this paper takes the average of the "Running Sum" [12]. The process can be expressed by the mathematical formula below.

$$\bar{V_N} = \frac{1}{N} \sum_{i=1}^{N} f(x_i),$$

where $\bar{V_N}$ stands for the "Running Average" and N is the "Running Times". $f(x_i)$ is the probability generated by each time where i stands for the order in the running times.

For the Snowball AutoCallable, as the paper introduced in the previous chapters, it is different from the traditional option pricing since it has knock-in and knock-out scenarios which complicates the payoff calculation and the expiration dates. This method could simulate and generate every path of the stock prices so that the interest of the Snowball AutoCallable can be calculated based on the known situation. For example, if the stock price is higher than the knock-out level and has never knocked in before this time spot, then the put option suspends and investors get the coupon interest in advance. Instead of setting various time and price boundaries in the traditional Black-Scholes model to tackle the situation like this, MCS provides a flexible and easy way to option pricing. Successful examples like Schwartz and Torous (1989) who used this model to calculate the mortgaged-back securities, and Boyle et al. (1997) who used this method to price the American option prove the feasibility and high-precision of the Monte-Carlo Simulation [13].

5.3.3 Brownian Motion introduction

This paper uses the Monte-Carlo to simulate the stock price for the following one-year S&P500 trend by Geometric Brownian motion model. The Geometric Brownian motion (GBM) refers to the continuous time-depending on the stochastic process where the logarithm of the randomly floating quantity follows the Brownian motion (also known as the Weiner process) [5], and a stochastic process represents a system that evolves in a probabilistic manner [5].

$$dS_t = \mu S_t dt + \sigma S_t dW_t$$

- S_t: the stock price at time
- μ: the drift term, representing the expected return of the stock
- σ: the volatility of the stock, indicating the degree of variation in returns
- dW: the increment of a Wiener process (or Brownian motion), capturing the random shocks to the stock price

This research uses Geometric Brownian Motion (GBM) as the basic model for Monte Carlo Simulation to generate future stock prices because it accurately reflects how stock prices behave in the real world. GBM models prices as a continuous process that always stays positive, incorporates both expected returns (drift) and volatility (risk) and ensures that the logarithm of returns is normally distributed. This makes it ideal for simulating realistic price paths over time.

5.3.4 Parameters setting up

Following the establishment of the foundational model for simulating the Snowball structure, this research delineates the specific methodologies employed for accurately pricing the Snowball product. The approach involves utilizing Geometric Brownian Motion (GBM) to simulate daily stock prices, with a focus on the stock's trajectory over 252 trading days, corresponding to one year. Specifically, the GBM method is applied iteratively for 252 periods to generate stock price paths for the S&P 500 from June 2024 to June 2025. Additionally, a ceiling and floor for stock movements are set at 1.1 and 0.9 times the current stock price, respectively, reflecting the limited volatility observed in the actual stock market. Although significant daily fluctuations can occur, such instances are infrequent and are not considered within the scope of the simulated stock prices, as they are deemed sudden and unpredictable. The Monte Carlo Simulation (MCS) methodology necessitates a substantial number of repeated experiments to enhance data accuracy; hence, the number of iterations is established at 5,000. This configuration yields 5,000 distinct stock price change paths for the S&P 500 over the specified year, with the visualization of the GBM paths presented below.

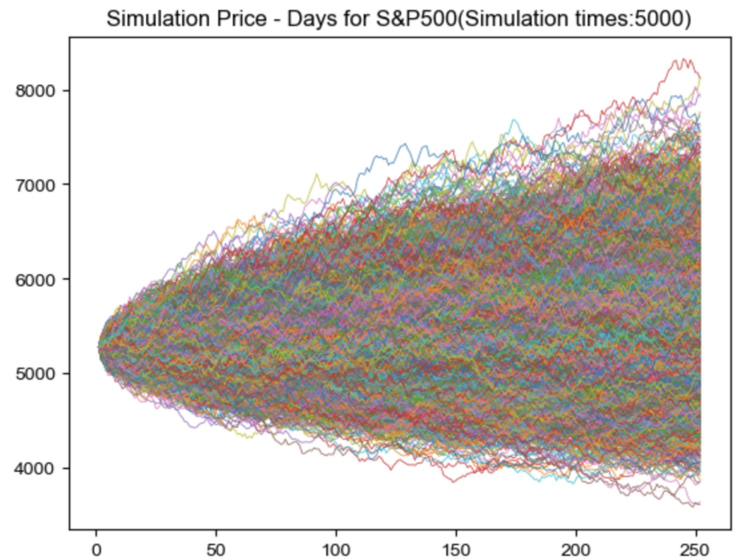


Fig. 8. Monte Carlo 5000-time Simulation

Next, based on the knock-in and knock-out levels we set, this paper determines the product's knock-in and knock-out and calculates the changes in net asset value (NAV) accordingly, by classifying all changes in NAV into the following four scenarios:

- Scenario 1: If there is a knock-out, the investor receives a discounted coupon payment.
- Scenario 2: If no knock-in and knock-out occurs, the full coupon is paid at maturity.
- Scenario 3: If only a knock-in occurs but the price recovers, the NAV remains 1.
- Scenario 4: If a knock-in occurs and the price does not recover, the NAV reflects a loss. This is the loss caused by the decline in asset value without the consideration of cash discount issues.

After considering the asset changes brought about by the knock-in and knock-out for each path, this research adds all the calculated NAV results to the Running Sum. After 5000 calculations, this research divides the Running Sum by 5000 to obtain the average NAV. This completes the pricing of the Snowball Autocallable, with the final price being 5308.226, which is 1.0058 times the initial price.

5.3.5 Validation of the Results

The variance and timeliness of the data are important measures [13] when dealing with the validation of the results. If the variance of the data decreases and approaches zero, it indicates that the data is converging, thus proving the simulation is reliable. Additionally, computational time addresses the cost issue of the model. If the runtime is excessively long, it becomes nearly impossible for pricing models to deal with decades of historical data. Our results show that as the number of simulations increases, the variance of the data first increases then decreases, and converges to 0 finally. Although there are fluctuations, this may be due to insufficient simulation runs and the inherent randomness of the event. The research also found that within 5000 to 6000 times, the model achieves the highest efficiency (explain why this research chooses 5000 times as the experimental times), with both rather low and stable variance and shorter runtime. For times larger than this period the improvement of variance compared to the selected range is minimal, and the runtime starts to increase significantly.

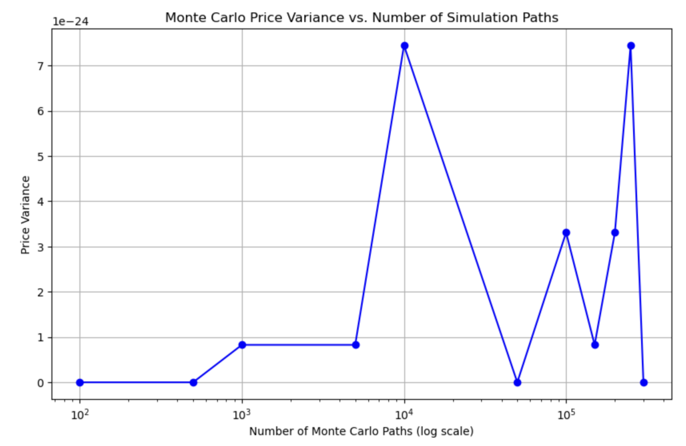


Fig. 9. Monte Carlo Results Convergence Test

Standardly, each stock price distribution, as an individual process, follows the logarithm of the Normal Distribution.

$$\bar{V_N} = \frac{1}{N}\sum_{i=1}^{N} V_i,$$

$$\bar{V_N} \xrightarrow{i.d.} \mathbf{N}(\mu, \frac{\sigma}{\sqrt{N}}).$$

Therefore, the running average of all these paths should also follow the normal distribution, which statistically measures the uncertainty (variance) of this simulation. Theoretically, the uncertainty in the simulation results is given by

$$\sqrt{\mathbb{V}[\bar{V_N}]} = \frac{\sigma}{\sqrt{N}}.$$

However, the real variance and expectation are not known since the whole process is simulated and we use this process to calculate the expectation (the averaged NAV). Thus, the estimation could only be approached as the second variance formula we introduced before [12],

$$\bar{\sigma_N} = \sqrt{\frac{1}{N}(\sum_{i=1}^{N} V_i{}^2) - (\frac{1}{N}\sum_{i=1}^{N} V_i)^2},$$

and this gives the standard definition of the standard error in the Monte-Carlo Simulation

$$\varepsilon_N = \frac{\bar{\sigma_N}}{\sqrt{N}}.$$

In conclusion, the Monte-Carlo Simulation offers a robust and precise simulation method for a structured product like the Snowball Autocallable product, with the flexibility to handle complex scenarios like knock-in and knock-out conditions. By leveraging the Geometric Brownian Motion model, we simulated the one-year price path which reflects the real financial market behaviors. Through the implementations of these models, we derived a comprehensive set of potential outcomes, facilitating a nuanced understanding of how varying conditions affect the net asset value (NAV) of the Snowball product. Our results demonstrate the importance of convergence and variance analysis, affirming that a higher number of simulations enhances the reliability of the pricing model. Notably, we established that 5000 simulation runs strike an optimal balance between computational efficiency and precision, yielding stable variance and acceptable runtimes. However, since Monte-Carlo Simulation is still an estimation method, it still has space for improvement.

6 Result Analysis

After introducing the two methods of pricing the Snowball option, we would like to address the comparison between Monte Carlo simulations and PDE methods for option pricing. The final results generated by the two methods are,

- Price generated by the PDE-based method is 5309.124.
- Price generated by Monte-Carlo Simulation is 5308.226.

By calculation, the similarity reaches 99.98%. Since this paper already addresses the feasibility and the reliability of the two methods significantly in the previous chapters, this paper would like to analyze the reason causing the distinctions between the two-pricing data by Monte Carlo simulations and the PDE-based method. After the analysis, we conclude the rationales from the following two reasons. For Monte Carlo simulations, the accuracy of our results depends heavily on the number of simulations conducted. Ideally, the number of simulations should be calculated by the formula below, the same formula as the error analysis,

$$\varepsilon_N = \frac{\bar{\sigma_N}}{\sqrt{N}}.$$

The errors should be expressed by the absolute value of $|f-\mu|$. Set the confidence interval as 95% (the confidence interval is the interval expected to contain the estimated values) and ω is the standard deviation (Recall that the 95% of the distribution is within 1.96 ω from the μ).

$$\mu - \frac{1.96\omega}{\sqrt{M}} < f < \mu + \frac{1.96\omega}{\sqrt{M}}.$$

By the calculation, the simulation times (with the confidence interval 95%) should be around 10000 times. However, our research decides to balance the efficiency and the precision of the data. Although the runtime in our case does not vary largely, for the companies or investors who deal with the ecumenical data (for example, 10 years), the increment of the calculation time matters significantly. At the same time, the companies or individual investors increasing time costs in pursuit of data accuracy should also be supported.

On the other hand, PDE methods rely on real-time stock prices of the underlying asset. Since future real-time prices cannot be predicted, this limitation introduces biases and discrepancies in the PDE results. In summary, while both approaches are effective, enhancing the simulation times in MC simulation and addressing the challenges of real-time price forecasting in PDE methods could further improve accuracy.

7 Risk Management

7.1 Internal risk analysis

To capture the different impacts of knock-out price, knock-in price, and sigma on the price of an option, we perform a sensitivity analysis.

7.1.1 Knock-in Level

Figure 10 shows the relationship between the knock-in level and the option value. Knock-in level of 0.85 is a reasonable compromise between risk and value. Option values below 0.85 remain high, but above 0.85, the value of the option drops sharply due to the increased risk of knock-in events. The knock-in probability at 0.85 is moderate, which means the option retains reasonable value but minimizes risk, thereby balancing potential returns with knock-in risks.

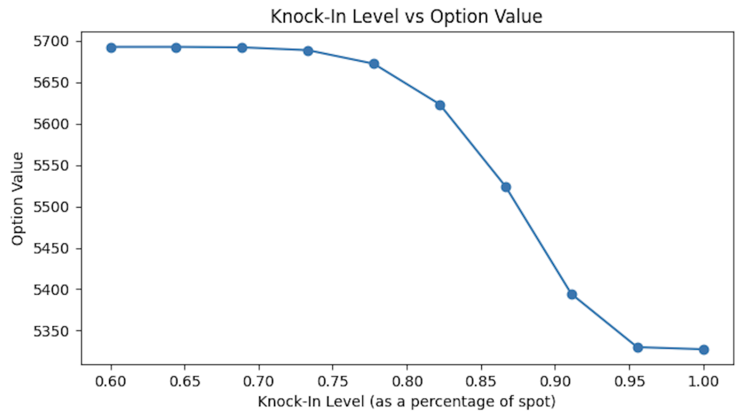


Fig. 10. The impact of knock-in level on option value

7.1.2 Knock-out Level

Figure 11 shows the relationship between the knock-out level and the option value. Using 1.03 as the knockout level balances the potential returns and risks of an early termination. This conservative margin reduces the chances of an option being knocked out prematurely by requiring the asset to exceed the spot price by 3%. It can be seen from the graph that increasing the knock-out level results in a higher option value, but the incremental gain diminishes as the knock-out level goes beyond 1.03. As long as the option remains active at this level, investors maintain a competitive payout regardless of market conditions while maximizing option value without taking on excessive risk.

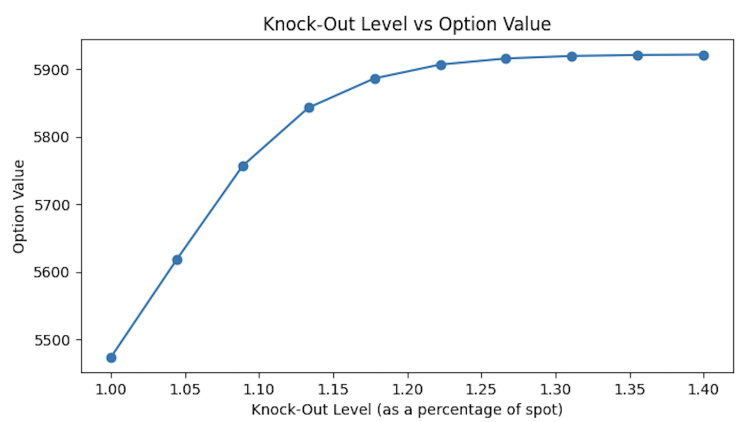


Fig. 11. The impact of knock out level on option value

7.1.3 Volatility

Figure 12 illustrates the negative correlation between volatility and option value, which is characterized by a decline in option value with increasing volatility. Volatility increases the likelihood that an asset will breach knock-in barriers or knock-out barriers as a result of greater uncertainty in its price movements. Option value declines as the probability of termination or knockout rises, leading to a lower expected payoff. When volatility is high in structured products, such as AutoCallables, the option's attractiveness and value are reduced.

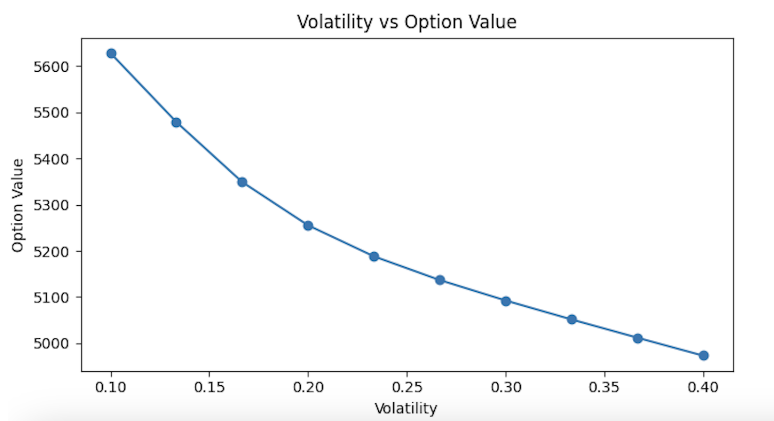


Fig. 12. The impact of volatility on option value

7.2 External risk analysis

7.2.1 Credit risk

Credit risk refers to the risk of a counterparty being unable to fully perform a contract. It exists not only in lending and bond investments, but also in wealth management products such as Snowball Auto-callable. Therefore, as the issuer of Snowball Auto-callable products, it is necessary to establish a credit rating system and conduct risk diversification and other measures to reduce the impact of credit risk on products.

7.2.2 Market trend risk

For investors, also the buyers of Snowball Auto-callable,who short volatility, choosing the time window of volatility within the duration is more conducive to profit; in only one of the five scenarios mentioned above, a loss of principal will occur. If the market falls sharply and breaks below the knock-in price, investors are responsible for all losses incurred by the S&P 500 as a result of the decline. Thus, this product is more suitable for investors in slightly volatile market conditions or a slowly rising market, based on the premise that the issuer and investors have a correct judgment of the expected trend of the underlying assets. When the underlying assets change unexpectedly, the issuer may suffer losses from market risks without correct hedging operations and losses.

7.2.3 Liquidity risk

Liquidity risk refers to the risk that an asset cannot be traded quickly and smoothly or realized at a reasonable price because of the imbalance between buyers and sellers in the market. As this product does not allow early redemption and needs to trigger the knock-out conditions for automatic redemption,the issuer should guarantee the liquidity of the products during risk hedging to prevent the liquidity shortage of the issuer, and investors should also have a certain amount of idle funds to deal with the risk of irregular redemption of Snowball Auto-callable.

7.3 Risk management strategy-Delta hedging strategy

Delta refers to the sensitivity index of all types of financial derivatives to the underlying assets. It is defined as the rate of change of the option price with respect to the price of the underlying asset. It is the slope of the curve that relates the option price to the underlying asset price. [5]

In general,

$$Delta = \frac{\partial V}{\partial S}$$

here V refers to the price of Snowball Auto-callable and S refers to the price of S&P 500.

For this Snowball Auto-callable, delta and Gamma follows the graph below.

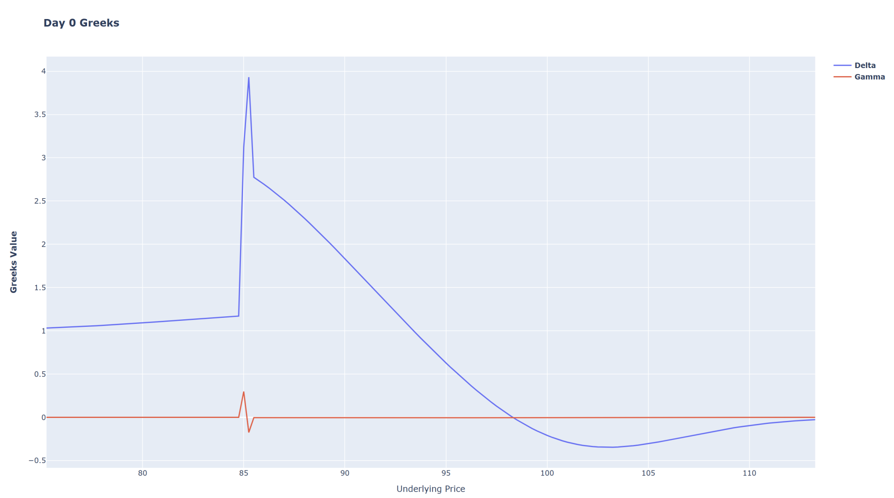


Fig. 13. The trend of greeks as S&P 500 moves

Figure 13 shows that when the price of the underlying asset falls close to the knock-in line, the snowball Delta increases rapidly, whereas below the knock-in line, Delta approaches 1; that is, the snowball option almost fails after knocking in, and its value decreases rapidly.

The core method of delta hedging is to construct a portfolio and calculate the value of delta. When Delta is equal to 0, the value of the portfolio is stable; that is, when the value of the underlying asset fluctuates within a certain range, it is not influenced by the portfolio price.

For issuers, profit includes two main parts: time value and volatility. Time value refers to the degree of change of the option price in unit time; the greater the degree of change, the greater the operating space of the issuer. Volatility pertains to the fluctuation degree of the index, and Vega indicates the changing degree of the option price within the unit volatility. Similarly,the greater the degree of change, the greater the operating space of issuers.

Thus, in this case, the hedging operations conducted by the issuer primarily involve gradual position building and high selling and low buy transactions. This is achieved through continuous adjustments in the buying and selling of stock index futures to mitigate the impact of the delta value on the price of stock index futures. The objective is to maintain the stability of the portfolio value until the delta value approaches zero. This means that when the S&P 500 falls and Delta becomes larger, it is helpful to buy stock index futures to reduce the impact on prices. Risk hedging can be achieved through low buying and high selling operations.

In real transactions, it is important to balance between risk and cost; if the hedging frequency is too high, it will reduce hedging risk but increase hedging cost. If the hedging frequency is too low, it will increase the hedging risk. Therefore, it is necessary to establish risk exposure. When the index changes within the range of risk exposure, it is acceptable to the issuer; therefore, there is no need to conduct the hedging operation. When the fluctuation of the index exceeds risk exposure, the issuer needs to hedge immediately and realize risk hedging by buying and selling stock index futures. The interests of investors and issuers can be guaranteed through a reasonable risk-hedging strategy and risk-management mechanism.

8 Conclusion

In conclusion, the pricing results of our S&P 500-Linked Snowball Autocallable demonstrate a certain level of stability. It is an innovative product in terms of structural design, is feasible for issuance, and has potential investment value in a volatile market. In addition, it is subject to the same limitations as other auto-callable products, such as low liquidity and higher risk when compared to traditional fixed-income securities. The use of two pricing methods ensures reliable pricing and provides an in-depth mathematical derivation, thus filling a market gap. The paper has a limitation in that it does not justify the initial conditions, such as volatility, coupon rate and barrier levels, which are based on general market assumptions. It is not possible to fully verify the pricing's optimality, even after conducting a sensitivity analysis. In the future, research will be conducted on the feasibility of the product on the market, as well as stress tests on the product.

References

[1] Cox, John C., Stephen A. Ross, and Mark Rubinstein. “Option Pricing: A Simplified Approach.” Journal of Financial Economics 7, no. 3 (1979): 229-263.

[2] Schwartz, Eduardo S. “The Valuation of Warrants: Implementing a New Approach.” Journal of Financial Economics 4, no. 1 (1977): 79-93.

[3] Boyle, Phelim P. “Options: A Monte Carlo Approach.” Journal of Financial Economics 4, no. 3 (1977): 323-338.

[4] Wilmott, Paul, Sam Howison, and Jeff Dewynne. 2010. The Mathematics of Financial Derivatives : A Student Introduction. Cambridge: Cambridge University Press, Dr.

[5] Hull, John. 2012. Options, Futures, and Other Derivatives. Boston: Prentice Hall.

[6] Lu, Xuewen. 2024. “Analysis of Stock Price Volatility Based on GARCH Family Models.” Scientific Journal of Economics and Management Research 6, no. 3.

[7] Deng, Geng, Joshua Mallett, and Craig McCann. 2011. “Modeling Autocallable Structured Products.” Journal of Derivatives & Hedge Funds 17 (4): 326–40. https://doi.org/10.1057/jdhf.2011.25.

[8] Evans, Lawrence C. 2010. Partial Differential Equations. Providence, R.I.: American Mathematical Society.

[9] Victor Grigor’e Ganzha, and Evgenii Vasilev Vorozhtsov. 2017. Numerical Solutions for Partial Differential Equations. CRC Press.

[10] Rich, Don R. 2000. The Mathematical Foundations of Barrier Option-Pricing Theory.

[11] Meding, Isak, and Viking Zandhoff Westerlund. ”Pricing European options with the Black-Scholes and Monte Carlo methods: a comparative study.” (2022).

[12] Jäckel, Peter. 2002. Monte Carlo Methods in Finance. Chichester, West Sussex, England: J. Wiley.

[13] Jabbour, George M., and Yi-Kang Liu. 2011. “Option Pricing and Monte Carlo Simulations.” Journal of Business & Economics Research (JBER) 3 (9).

[14] Izvorski, Ivailo. 1998. “A Nonuniform Grid Method for Solving PDE’s.” Journal of Economic Dynamics and Control 22 (8-9): 1445–52. https://doi.org/10.1016/s0165-1889(98)00020-7.