Available online at www.sciencedirect.com
ScienceDirect
Mathematics and Computers in Simulation 182 (2021) 690–704
MATHEMATICS AND COMPUTERS IN SIMULATION
www.elsevier.com/locate/matcom

Original articles

Optimal non-uniform finite difference grids for the Black–Scholes equations

Jisang Lyu[a], Eunchae Park[a], Sangkwon Kim[a], Wonjin Lee[b], Chaeyoung Lee[a], Sungha Yoon[a], Jintae Park[a], Junseok Kim[a,*]

[a] *Department of Mathematics, Korea University, Seoul 02841, Republic of Korea*
[b] *Department of Financial Engineering, Korea University, Seoul 02841, Republic of Korea*

Received 20 July 2020; received in revised form 14 October 2020; accepted 3 December 2020
Available online 7 December 2020

Abstract

In this article, we present optimal non-uniform finite difference grids for the Black–Scholes (BS) equation. The finite difference method is mainly used using a uniform mesh, and it takes considerable time to price several options under the BS equation. The higher the dimension is, the worse the problem becomes. In our proposed method, we obtain an optimal non-uniform grid from a uniform grid by repeatedly removing a grid point having a minimum error based on the numerical solution on the grid including that point. We perform several numerical tests with one-, two- and three-dimensional BS equations. Computational tests are conducted for both cash-or-nothing and equity-linked security (ELS) options. The optimal non-uniform grid is especially useful in the three-dimensional case because the option prices can be efficiently computed with a small number of grid points.

Keywords: Black–Scholes equations; Optimal non-uniform grid; Finite difference method; Equity-linked securities

1. Introduction

The structure of financial derivatives has recently become more complex. Equity-linked security (ELS) is an autocallable financial product, a complex structure commonly traded in South Korea. The most popular ELS product is based on three underlying assets. Currently, three-asset ELS products are the most traded in South Korea's financial market. We surveyed the ELS products traded from March 1, 2019, to February 29, 2020, in three financial companies, and the three-asset products account for more than 75% of the total trades, see Fig. 1.

The recent trend of constructing ELS products with multiple underlying assets complicated the structure of the product. Hence, pricing these complicated autocallable products efficiently and accurately [11] is practical and important. Three-asset step-down ELS, which is the most commonly traded structure, is automatically exercised on each redemption date based on the condition of each product. Step-down indicates that the strike price condition at each redemption date decreases until maturity. Usually, the price of the underlying asset represents the minimum

* Corresponding author.
E-mail address: cfdkim@korea.ac.kr (J. Kim).
URL: http://math.korea.ac.kr/~cfdkim/ (J. Kim).

https://doi.org/10.1016/j.matcom.2020.12.002

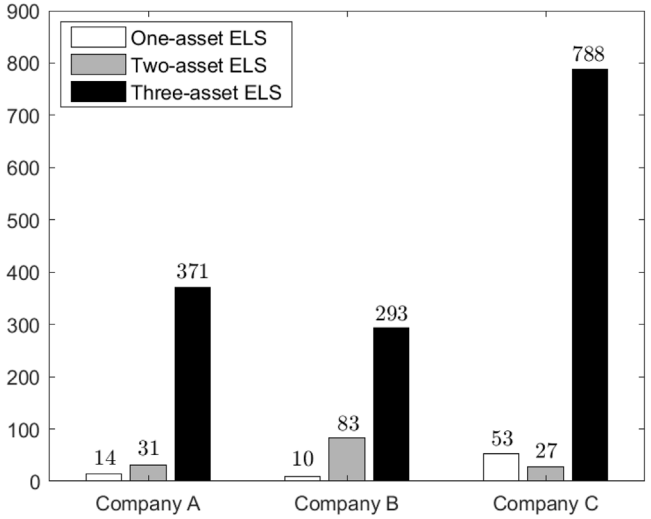

Fig. 1. Survey on the ELS products of three financial companies.

price among the underlying assets. On the redemption date, whether it will be exercised is decided. When the price of the underlying asset is higher than the strike price, the option is automatically exercised with a coupon rate. If the early exercise does not happen, the option continues until the next redemption date. At maturity, if the price of the underlying asset is above the knock-in barrier during the contract, the option ends with a dummy rate. If not, the option ends with the price of the underlying asset at maturity, i.e., with a loss.

The option price can be obtained by solving the Black–Scholes (BS) equation. Let s_i $(i = 1, 2, \ldots, n)$ indicate the price of the underlying ith asset and $u(s_1, s_2, \ldots, s_n, t)$ indicate the value of the option. Consider the n-asset BS equation as follows [3,17]:

$$\frac{\partial u(\mathbf{s}, t)}{\partial t} + \frac{1}{2}\sum_{i,j=1}^{n} \sigma_i \sigma_j \rho_{ij} s_i s_j \frac{\partial^2 u(\mathbf{s}, t)}{\partial s_i \partial s_j} + r\sum_{i=1}^{n} s_i \frac{\partial u(\mathbf{s}, t)}{\partial s_i} = r u(\mathbf{s}, t), \tag{1}$$

for $(\mathbf{s}, t) = (s_1, s_2, \ldots, s_n, t) \in \mathbf{R}_+^n \times [0, T)$. Here, σ_i denotes a constant volatility of the ith asset, $r > 0$ is a constant interest rate and ρ_{ij} denotes the correlation coefficient between two underlying ith and jth assets. The final condition is the payoff function $\Phi(\mathbf{s})$ at maturity T

$$u(\mathbf{s}, T) = \Phi(\mathbf{s}). \tag{2}$$

There are two popular numerical methods for computing option pricing [8,9,14]: Monte Carlo simulation (MCS) and finite difference method (FDM). The advantage of MCS is that it can be applied to higher dimensions with only a small additional computing time. It differs from the FDM in that its price is not the same for every simulation. The advantage of FDM is that it can obtain the same price for every simulation; however, applying it to multi-asset option pricing is difficult. The FDM takes considerable cost to price options with multi-assets because it mainly uses a uniform grid. Therefore, FDM with a uniform grid is inappropriate for pricing the three-asset ELS in terms of efficiency.

Several studies using non-uniform grids have been conducted to reduce computational cost while having the same accuracy [2,4,19]. Shojaei et al. [25] coupled uniform grids with different grid spacing to make a non-uniform grid at a much smaller computational cost. Dilloo and Tangman [7] demonstrated the performance of a high-order accurate FDM through a proposed local mesh refinement strategy that creates the non-uniform grid by adding two nodes to a uniform grid. The additional two nodes in the proposed strategy are located near the singular point usually found in the payoff of options. Mishra and Lu [24] used non-uniform grid in pricing multi-asset options for practical purposes. They used a dense and uniform grid on the interval of interest and used a sparse and non-uniform grid outside the interval of interest. Milovanović and von Sydow [22] used a non-uniform grid to price European- and American-type multi-asset options using a radial basis function generated finite difference (RBF-FD) method.

Furthermore, in [23], the authors placed more nodes in the area for higher accuracy. The sparse grid showed high accuracy even with a relatively small number of nodes. In [15], the authors priced a multi-dimensional European option with a numerical method based on RBF. They used the operator splitting method to price the two-asset option. In addition, the computational time and errors with respect to the number of discretized grid points was checked. They refined the nodes in areas of interest to reduce computational costs while having an accurate solution. In [28], the author solved the 3D Heston–Hull–White partial differential equation on a non-uniform grid for efficiency. Zhang and Li [29] suggested piecewise uniform grids (PU-MidK) with a grid point at every barrier and tested the proposed method to price the barrier call option. The test was based on error analysis and demonstrated that the PU-MidK is efficient compared to the uniform and non-uniform grids suggested in [21]. Most non-uniform grid methods were refined to be finer in the hot zone, representing the interval of interest. Soleymani and Barfeie [26] used a non-uniform grid to solve the stochastic volatility jump model. The tests on the several jump models with the constructed grid showed improved computational efficiency and accuracy. In [5], the authors constructed a non-uniform grid to price European- and American-type options using Merton's jump diffusion model. The suggested non-uniform grid was dense near the strike price and at expiry. They used FDM for the spatial differential operators and the Galerkin finite element method for time stepping. In [20], the author used FDM on refined two-dimensional non-uniform grids composed of repeatedly finer rectangular grids. A previous study [18] used a non-uniform grid to price three-asset ELS. However, the suggested non-uniform grid was obtained by trial and error. To make progress, we are going to obtain a non-uniform grid using the systematic approach.

In this paper, we propose an algorithm to construct a non-uniform grid using systematic error. We can obtain an optimal non-uniform grid by sequentially removing a node having the minimum error based on the numerical solution on the grid including that point.

The remainder of this paper is structured as follows. In Section 2, we describe FDM to solve three-dimensional BS equation and propose an algorithm to construct the non-uniform finite difference grid. In Section 3, the numerical experiments demonstrate the performance of the proposed algorithm. Finally, the conclusions are provided in Section 4.

2. Numerical solutions

By changing the variable with $\tau = T - t$, Eq. (1) can be rewritten as follows:

$$\frac{\partial u(\mathbf{s},\tau)}{\partial \tau} = \frac{1}{2}\sum_{i,j=1}^{n} \sigma_i \sigma_j \rho_{ij} s_i s_j \frac{\partial^2 u(\mathbf{s},\tau)}{\partial s_i \partial s_j} + r\sum_{i=1}^{n} s_i \frac{\partial u(\mathbf{s},\tau)}{\partial s_i} - ru(\mathbf{s},\tau), \tag{3}$$
$$\text{for } (\mathbf{s},\tau) = (s_1, s_2, \ldots, s_n, \tau) \in \mathbf{R}_+^n \times (0, T].$$

The initial condition is the payoff function $\Phi(\mathbf{s})$ at expiry $\tau = 0$:

$$u(\mathbf{s}, 0) = \Phi(\mathbf{s}). \tag{4}$$

2.1. Three-dimensional Black–Scholes equation

Let x, y and z be the price of the three underlying assets, and $\tau = T - t$ be the time to maturity. The option price $u(x, y, z, \tau)$ for $(x, y, z) \in \Omega$ and $\tau \in (0, T]$ follows the BS partial differential equation:

$$\begin{aligned} u_\tau(x,y,z,\tau) &= rxu_x(x,y,z,\tau) + ryu_y(x,y,z,\tau) + rzu_z(x,y,z,\tau) \\ &\quad + \frac{1}{2}\sigma_x^2 x^2 u_{xx}(x,y,z,\tau) + \frac{1}{2}\sigma_y^2 y^2 u_{yy}(x,y,z,\tau) \\ &\quad + \frac{1}{2}\sigma_z^2 z^2 u_{zz}(x,y,z,\tau) + \rho_{xy}\sigma_x\sigma_y xy u_{xy}(x,y,z,\tau) \\ &\quad + \rho_{yz}\sigma_y\sigma_z yz u_{yz}(x,y,z,\tau) + \rho_{zx}\sigma_x\sigma_z xz u_{zx}(x,y,z,\tau) \\ &\quad - ru(x,y,z,\tau), \quad (x,y,z,\tau) \in \Omega \times (0,T], \\ u(x,y,z,0) &= \Phi(x,y,z), \end{aligned} \tag{5}$$

where Φ denotes the payoff function at maturity T and Ω is the computational domain. The subscripts of u denote partial derivatives with respect to the corresponding parameters. ρ_{xy}, ρ_{yz} and ρ_{zx} are the correlation values between the two asset variables. σ_x, σ_y and σ_z are the volatilities of the underlying assets x, y and z, respectively. First, we discretize the computational domain $\Omega = [L_x, R_x] \times [L_y, R_y] \times [L_z, R_z]$ with variable steps $h^x_{i-1} = x_i - x_{i-1}$, $h^y_{j-1} = y_j - y_{j-1}$ and $h^z_{k-1} = z_k - z_{k-1}$. Here, $x_0 = L_x$, $y_0 = L_y$, $z_0 = L_z$, $x_{N_x} = R_x$, $y_{N_y} = R_y$ and $z_{N_z} = R_z$. N_x, N_y, N_z and N_τ are the number of grid intervals in the x-, y-, z- and τ-directions, respectively. $\Delta\tau = T/N_\tau$ denotes the time step size. Let $u^n_{ijk} \equiv u(x_i, y_j, z_k, n\Delta\tau)$ be the numerical approximation of the solution, where $i = 0, \ldots, N_x$, $j = 0, \ldots, N_y$, $k = 0, \ldots, N_z$ and $n = 0, \ldots, N_\tau$. For both the cash-or-nothing option and ELS, we use the zero Dirichlet boundary conditions at $x = L_x$, $y = L_y$ and $z = L_z$. We use zero Neumann boundary conditions [12] when pricing the cash-or-nothing option and hybrid condition [6] when pricing ELS at $x = R_x$, $y = R_y$ and $z = R_z$. Subsequently, we apply the operator splitting method (OSM) [13] to solve Eq. (5):

$$\frac{u^{n+1}_{ijk} - u^n_{ijk}}{\Delta\tau} = \left(\mathcal{L}^x_{BS}u\right)^{n+\frac{1}{3}}_{ijk} + \left(\mathcal{L}^y_{BS}u\right)^{n+\frac{2}{3}}_{ijk} + \left(\mathcal{L}^z_{BS}u\right)^{n+1}_{ijk}, \tag{6}$$

where $\left(\mathcal{L}^x_{BS}u\right)^{n+\frac{1}{3}}_{ijk}$, $\left(\mathcal{L}^y_{BS}u\right)^{n+\frac{2}{3}}_{ijk}$ and $\left(\mathcal{L}^z_{BS}u\right)^{n+1}_{ijk}$ are defined as

$$\begin{aligned}
\left(\mathcal{L}^x_{BS}u\right)^{n+\frac{1}{3}}_{ijk} &= \frac{(\sigma_x x_i)^2}{2} D_{xx}u^{n+\frac{1}{3}}_{ijk} + r x_i D_x u^{n+\frac{1}{3}}_{ijk} + \frac{1}{3}\sigma_x\sigma_y\rho_{xy}x_i y_j D_{xy}u^n_{ijk} \\
&\quad + \frac{1}{3}\sigma_y\sigma_z\rho_{yz}y_j z_k D_{yz}u^n_{ijk} + \frac{1}{3}\sigma_z\sigma_x\rho_{zx}z_k x_i D_{zx}u^n_{ijk} - \frac{1}{3}r u^{n+\frac{1}{3}}_{ijk}, \\
\left(\mathcal{L}^y_{BS}u\right)^{n+\frac{2}{3}}_{ijk} &= \frac{(\sigma_y y_j)^2}{2} D_{yy}u^{n+\frac{2}{3}}_{ijk} + r y_j D_y u^{n+\frac{2}{3}}_{ijk} + \frac{1}{3}\sigma_x\sigma_y\rho_{xy}x_i y_j D_{xy}u^{n+\frac{1}{3}}_{ijk} \\
&\quad + \frac{1}{3}\sigma_y\sigma_z\rho_{yz}y_j z_k D_{yz}u^{n+\frac{1}{3}}_{ijk} + \frac{1}{3}\sigma_z\sigma_x\rho_{zx}z_k x_i D_{zx}u^{n+\frac{1}{3}}_{ijk} - \frac{1}{3}r u^{n+\frac{2}{3}}_{ijk}, \\
\left(\mathcal{L}^z_{BS}u\right)^{n+1}_{ijk} &= \frac{(\sigma_z z_k)^2}{2} D_{zz}u^{n+1}_{ijk} + r z_k D_z u^{n+1}_{ijk} + \frac{1}{3}\sigma_x\sigma_y\rho_{xy}x_i y_j D_{xy}u^{n+\frac{2}{3}}_{ijk} \\
&\quad + \frac{1}{3}\sigma_y\sigma_z\rho_{yz}y_j z_k D_{yz}u^{n+\frac{2}{3}}_{ijk} + \frac{1}{3}\sigma_z\sigma_x\rho_{zx}z_k x_i D_{zx}u^{n+\frac{2}{3}}_{ijk} - \frac{1}{3}r u^{n+1}_{ijk}.
\end{aligned}$$

For the discretization of Eq. (6), we use

$$\begin{aligned}
D_x u_{ijk} &= -\frac{h^x_i}{h^x_{i-1}(h^x_{i-1} + h^x_i)}u_{i-1,jk} + \frac{h^x_i - h^x_{i-1}}{h^x_{i-1}h^x_i}u_{ijk} + \frac{h^x_{i-1}}{h^x_i(h^x_{i-1} + h^x_i)}u_{i+1,jk}, \\
D_{xx} u_{ijk} &= \frac{2}{h^x_{i-1}(h^x_{i-1} + h^x_i)}u_{i-1,jk} - \frac{2}{h^x_{i-1}h^x_i}u_{ijk} + \frac{2}{h^x_i(h^x_{i-1} + h^x_i)}u_{i+1,jk}, \\
D_{xy} u_{ijk} &= \frac{u_{i+1,j+1,k} - u_{i-1,j+1,k} - u_{i+1,j-1,k} + u_{i-1,j-1,k}}{h^x_i h^y_j + h^x_{i-1}h^y_j + h^x_i h^y_{j-1} + h^x_{i-1}h^y_{j-1}}.
\end{aligned}$$

2.2. Proposed algorithm

We propose an optimal non-uniform grid with a minimum number of grid points for the BS equation. The algorithm is as follows:

Step (1) Let $\Omega^0 = \{x_i | x_i = L_x + hi,\ i = 0, 1, \ldots, N_x\}$ *be the starting uniform discrete domain with step size* $h = (R_x - L_x)/N_x$. *Set the desired number of grid points* N_p. *Let* u_{ref} *be a reference solution of Eq. (3) on* Ω^0 *at the current price of an underlying asset of the product.*

We repeat *Step (2)* from $k = 1$ to $k = N_x - N_p + 1$.

Step (2) Let Ω^k_i *be the grid without* i*th element of* Ω^{k-1}, *i.e.,* $\Omega^k_i \subsetneq \Omega^{k-1}$. *Calculate an error* $e^k_i = |u^k_i - u_{ref}|/u_{ref}$, *where* u^k_i *is a solution to Eq. (3) on* Ω^k_i *at the current price of an underlying asset of product. Then, we obtain an index* j *satisfying* $e^k_j = \min_{1 \le i \le N_x + 1 - k} e^k_i$ *and set* $\Omega^k = \Omega^k_j$.

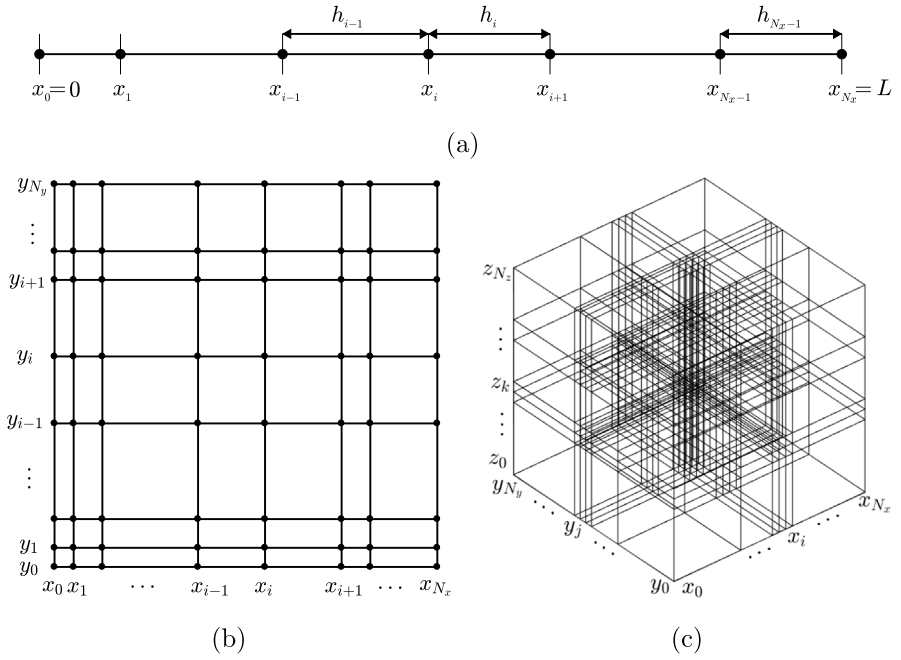

Fig. 2. (a) Schematic representations of an optimal non-uniform grid in one-dimensional space, and its extension to (a) two and (b) three dimensions.

Fig. 2(a) represents a non-uniform grid from this process. We take the optimal non-uniform grid obtained from the one-asset option and extend it to two- and three-asset options. Fig. 2(b) and (c) show how to construct two- and three-dimensional non-uniform grids from the one-dimensional optimal non-uniform grid.

3. Numerical experiments

In this section, depending on the number of underlying assets for each product, we create a non-uniform grid structure with the proposed algorithm in Section 2. As the payoff function is not differentiable at the strike price, more grid points were added near the strike price [7,10]. Here, we use the interest rate $r = 0.03$ and set u_{ref} as the reference price computed on a uniform grid with grid spacing of 1.

3.1. Cash-or-nothing option

We consider the pricing of the cash-or-nothing option of one underlying asset. This option is a derivative with a fixed payment. Fixed payment is paid if the underlying asset closes above the strike price at maturity; otherwise, it is not paid, i.e., the payment is zero. The parameters used are as follows: maturity time $T = 1$, time step $\Delta t = 1/30$, current price $S_0 = 100$, strike price $K = 100$, payment (cash) $C = 100$ and computational domain $\Omega = [L_x, R_x] \times [L_y, R_y] \times [L_z, R_z]$ with $L_x = L_y = L_z = 0.5$ and $R_x = R_y = R_z = 300.5$. We fix some grid points on the boundary of the domain Ω and neighboring points of the strike of the derivative product, see blue dots in Fig. 3.

3.1.1. Cash-or-nothing option on one asset

The volatility used is $\sigma_x = 0.3$. The payoff function of the cash-or-nothing option is

$$\phi(x, \tau = 0) = \begin{cases} 100, & \text{if } x \geq K, \\ 0, & \text{otherwise.} \end{cases} \tag{7}$$

Fig. 3 shows the results of the process of the proposed algorithm. Markers (black dots, red dots and blue dots) indicate Ω^k grid points for some k, points to be excluded, and fixed points for each operation, respectively.

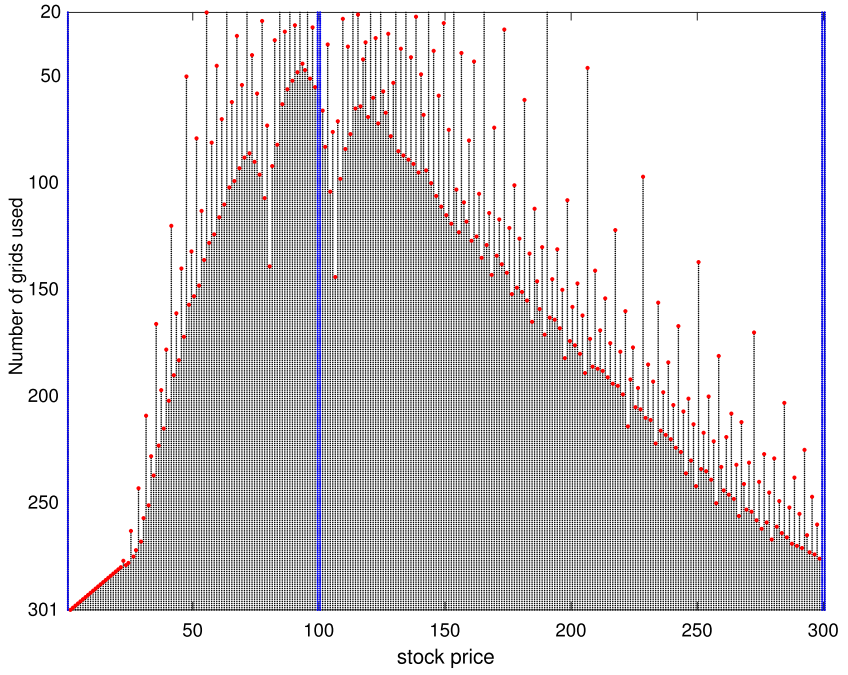

Fig. 3. Distribution of grid points with respect to the number of grid points for cash-or-nothing option of the one asset. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

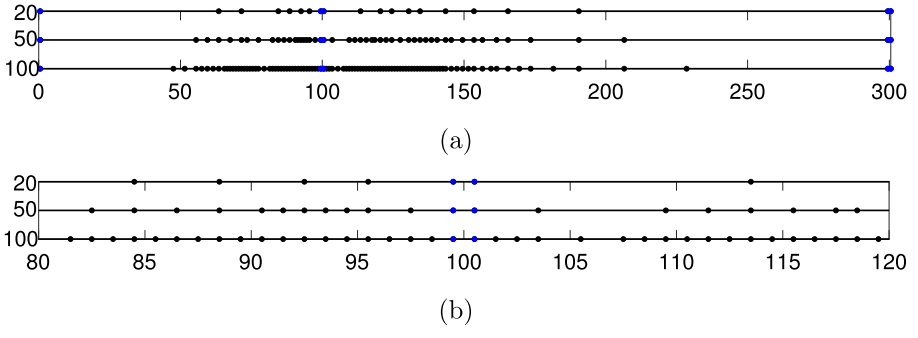

Fig. 4. (a) and (b) denote the distribution of non-uniform grid points for cash-or-nothing option with the number of grid points 20, 50 and 100 on $\Omega = [0.5,\ 300.5]$ and $[80,\ 120]$, respectively.

Fig. 4(a) shows the distribution of the non-uniform grid obtained with $N_p = 20,\ 50$, and 100 on $\Omega = [0.5,\ 300.5]$. Fig. 4(b) shows the distribution of the non-uniform grid enlarged on $[80,\ 120]$ for better visualization.

Fig. 5 shows the prices and elapsed times of the computation with uniform grids with respect to the number of grid points. Furthermore, the price of using the non-uniform grid with $N_p = 20$ is shown as a dotted line. We can confirm that a uniform grid requires a very large amount of time compared to the non-uniform grid to produce similar results.

3.1.2. Cash-or-nothing option on two assets

We consider the pricing of the cash-or-nothing option of two underlying assets. The parameters used are volatility of each underlying asset $\sigma_x = 0.3,\ \sigma_y = 0.3$, and correlation of underlying assets $\rho_{xy} = 0.5$. The payoff function

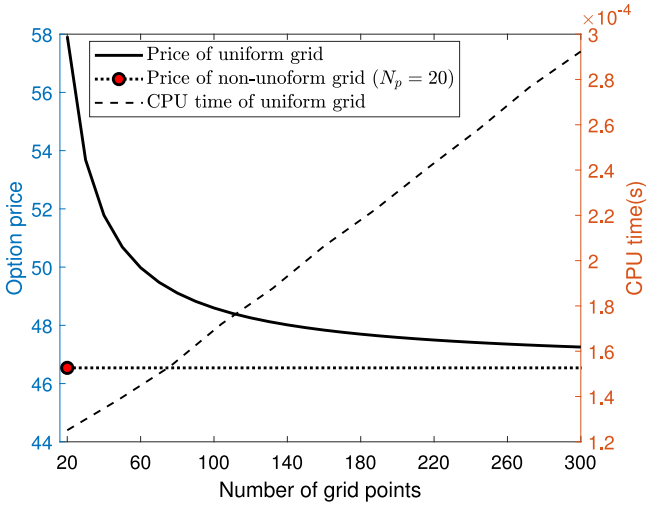

Fig. 5. Option price with the non-uniform grid of $N_p = 20$ and uniform grids from $N_x = 19$ to $N_x = 300$. CPU time is in seconds.

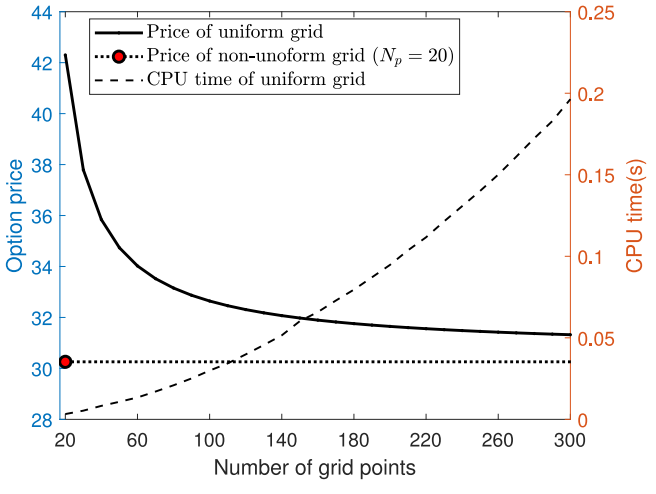

Fig. 6. Option price with a non-uniform grid of $N_p = 20$ and uniform grids from $N_x = N_y = 19$ to $N_x = N_y = 300$. CPU time is denoted in seconds.

of the cash-or-nothing option of the two assets is

$$\phi(x, y, \tau = 0) = \begin{cases} 100, & \text{if } \min(x, y) \geq K, \\ 0, & \text{otherwise.} \end{cases} \tag{8}$$

The two-dimensional non-uniform mesh is constructed using the non-uniform grid found in the one-dimensional space. Fig. 6 represents the prices and the elapsed times of the computation with uniform grids with respect to the number of grid points. Furthermore, the price of using a non-uniform grid with $N_p = 20$ is illustrated as dotted line. The result shows the price of uniform grid converges to the price with the non-uniform grid from our proposed algorithm.

3.1.3. Cash-or-nothing option on three assets

We consider the pricing of cash-or-nothing option of three underlying assets. The parameters used are: maturity time $T = 1$, volatility of each underlying asset $\sigma_x = 0.3$, $\sigma_y = 0.3$, $\sigma_z = 0.3$, correlation of each underlying asset

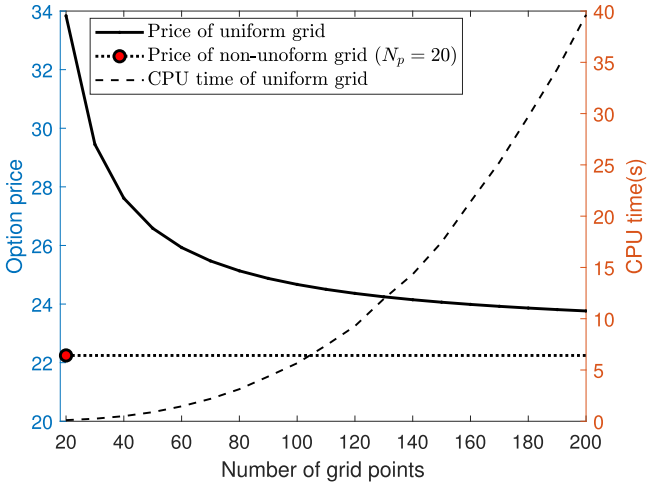

Fig. 7. Option price with a non-uniform grid of $N_p = 20$ and uniform grids from $N_x = N_y = N_z = 19$ to $N_x = N_y = N_z = 200$. CPU time is denoted in seconds.

$\rho_{xy} = 0.5$, $\rho_{xz} = 0.5$, $\rho_{yz} = 0.5$. The payoff function of the cash-or-nothing option of the three assets is

$$\phi(x, y, z, \tau = 0) = \begin{cases} 100, & \text{if } \min(x, y, z) \geq K, \\ 0, & \text{otherwise.} \end{cases} \tag{9}$$

A three-dimensional non-uniform mesh is constructed using the non-uniform grid observed in a one-dimensional space, similar to the two-dimensional space.

Fig. 7 shows the prices and the elapsed times of the computation with uniform grids with respect to the number of grid points. Herein, we perform the test until $N_x = N_y = N_z = 200$ owing to the high computational cost in three-dimensional space. The price of using a non-uniform grid with $N_p = 20$ is shown as a dotted line. We can conclude that the optimal non-uniform grid has similar results compared to a uniform grid in three-dimensions as well.

3.2. ELS option

We fix points on the strike prices and knock-in barrier for the ELS options. We conduct tests on ELS depending on the number of underlying assets. We proceed with a numerical test for checking the efficiency and accuracy of the proposed algorithm. We perform a convergence test using a reference value. The markers in each figure are the same as in the cash-or-nothing option. Unless otherwise specified, the parameters used are maturity time $T = 3$, time step $\Delta t = 1/360$, and the computational domain is $\Omega = [L_x, R_x] \times [L_y, R_y] \times [L_z, R_z]$ with $L_x = L_y = L_z = 0$ and $R_x = R_y = R_z = 300$ [16]. We fix some grid points that are on the boundary of the domain Ω, strike and neighboring points of the strike of the derivative product.

In the numerical test for ELS options, the parameters with current price $S_0 = 100$, strike prices $[K_1, K_2, K_3, K_4, K_5, K_6] = [95, 95, 90, 90, 85, 85]$, coupon rates $[c_1, c_2, c_3, c_4, c_5, c_6] = [0.05, 0.10, 0.15, 0.20, 0.25, 0.30]$, knock-in barrier $D = 50$, face value $F = 100$ and dummy rate $d = 0.3$ are used. Parameters for one- and two-asset ELS are similarly defined.

Fig. 8 shows the payoff function at early redemption dates. When numerically solving the equation, we define two option values $u(S, \tau)$ and $v(S, \tau)$ for hitting knock-in barrier or not, respectively. As shown in Fig. 9, $v(S, \tau)$ is above $u(S, \tau)$. We define Dirichlet boundary condition for $v(S, \tau)$ at the knock-in barrier after solving $u(S, \tau)$. Subsequently, we solve $v(S, \tau)$ in the range from the knock-in barrier to the upper bound. Therefore, we reflect the case when the stock price has been under the knock-in barrier and reduce the computational cost simultaneously.

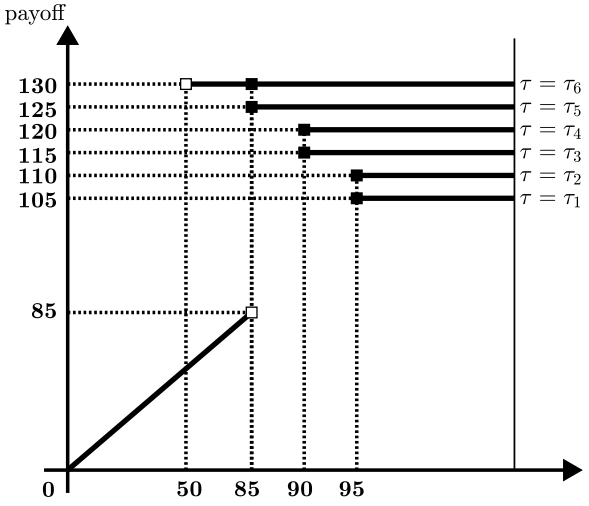

Fig. 8. Payoff function at early redemption dates.

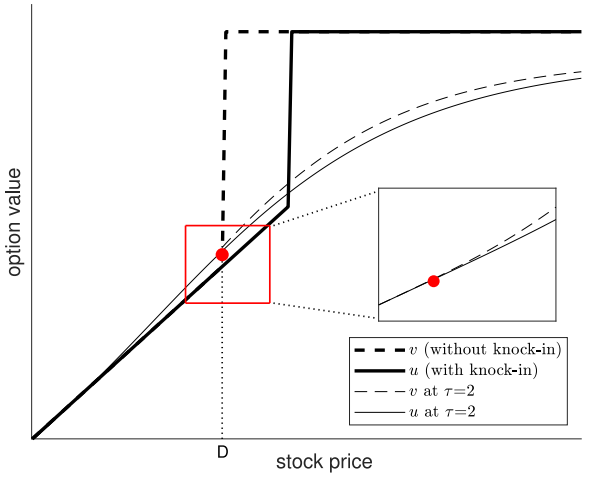

Fig. 9. Option values u and v at $\tau = 2$.

3.2.1. *One-asset ELS option*

We consider the pricing of one-asset ELS option. The parameters used are maturity time $T = 3$ and volatility of underlying asset $\sigma_x = 0.3$. Let $u(x, \tau)$ be the solution with the knock-in barrier event and $v(x, \tau)$ be the solution without a knock-in barrier event. The payoff function of the one-asset ELS option is

$$\begin{aligned} u(x, \tau = 0) &= \begin{cases} Fx/S_0, & \text{if } x < K_6, \\ F(1 + c_6), & \text{otherwise}, \end{cases} \\ v(x, \tau = 0) &= \begin{cases} F(1 + c_6), & \text{if } x \geq K_6, \\ F(1 + d), & \text{if } D < x < K_6, \\ Fx/S_0, & \text{otherwise}. \end{cases} \end{aligned} \tag{10}$$

Fig. 10 shows the non-uniform grids for each operation and presents the results of the process of the proposed algorithm for one-asset ELS option. In Fig. 10, the markers indicate the same things as in the cash-or-nothing option.

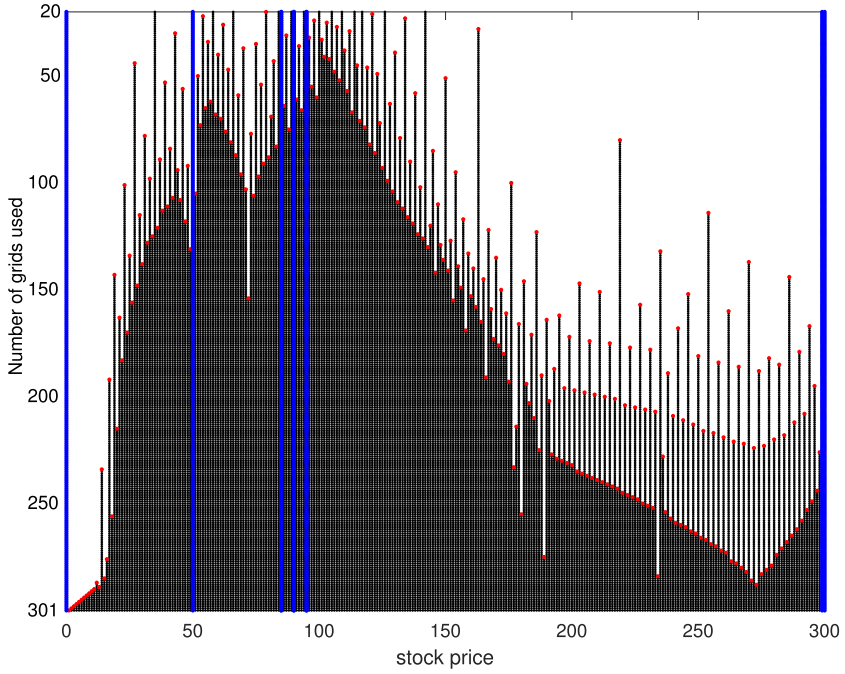

Fig. 10. Generating an optimal non-uniform grid for one-asset ELS.

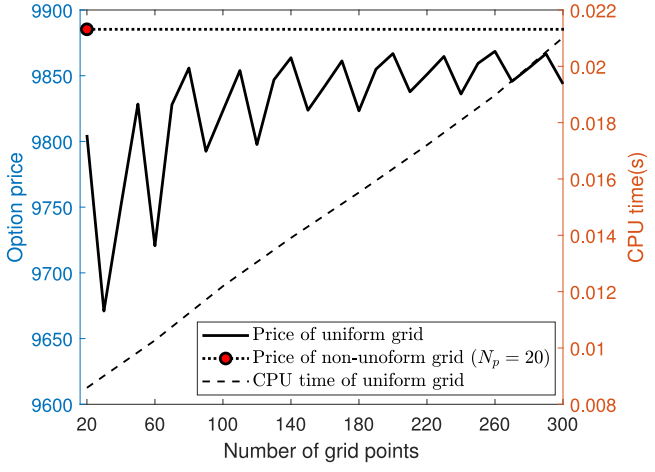

Fig. 11. Option price with a non-uniform grid of $N_p = 20$ and uniform grids from $N_x = 19$ to $N_x = 300$. CPU time is denoted in seconds.

Fig. 11 shows the prices and the elapsed times computed on uniform grids with respect to the number of grid points. Furthermore, the price of using a non-uniform grid with $N_p = 20$ is plotted as a dotted line. We can also conclude that the option price obtained in the optimal non-uniform grid is not only accurate but also takes less computation time.

3.2.2. Two-asset ELS option

We consider pricing the two-asset ELS option. The parameters are as follows: volatility of each underlying asset $\sigma_x = 0.3$, $\sigma_y = 0.3$, and correlation of underlying assets $\rho_{xy} = 0.5$. Let $S = \min(x, y)$ be the minimum of two

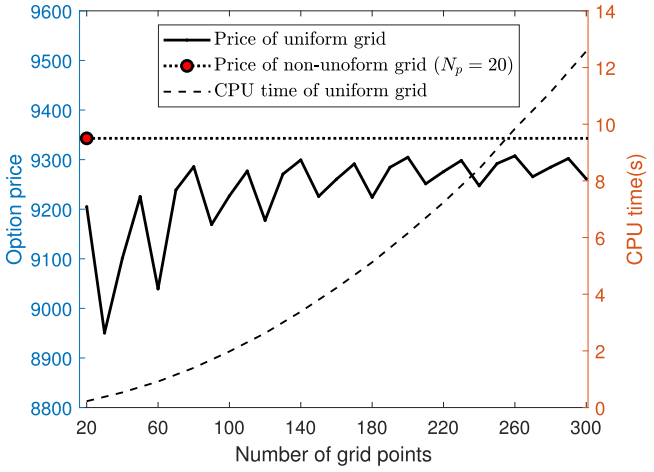

Fig. 12. Option price with a non-uniform grid of $N_p = 20$ and uniform grids from $N_x = N_y = 19$ to $N_x = N_y = 300$. CPU time is denoted in seconds.

underlying assets, x and y. The payoff function of the two-asset ELS option is

$$\begin{aligned} u(S, \tau = 0) &= \begin{cases} FS/S_0, & \text{if } S < K_6, \\ F(1 + c_6), & \text{otherwise}, \end{cases} \\ v(S, \tau = 0) &= \begin{cases} F(1 + c_6), & \text{if } S \geq K_6, \\ F(1 + d), & \text{if } D < S < K_6, \\ FS/S_0, & \text{otherwise}. \end{cases} \end{aligned} \tag{11}$$

Fig. 12 shows the prices and the elapsed times of using uniform grids with respect to the number of grid points, and the price of using a non-uniform grid with $N_p = 20$ is illustrated as a dotted line.

3.2.3. Three-asset ELS option

We consider the pricing of the three-asset ELS option. The parameters used are as follows: volatility of each underlying asset $\sigma_x = 0.3$, $\sigma_y = 0.3$, $\sigma_z = 0.3$, and correlation of each underlying asset $\rho_{xy} = 0.5$, $\rho_{xz} = 0.5$, $\rho_{yz} = 0.5$. Let $S = \min(x, y, z)$ be the minimum of the three underlying assets, x, y and z. The payoff function of three-asset ELS option is

$$\begin{aligned} u(S, \tau = 0) &= \begin{cases} FS/S_0, & \text{if } S < K_6, \\ F(1 + c_6), & \text{otherwise}. \end{cases} \\ v(S, \tau = 0) &= \begin{cases} F(1 + c_6), & \text{if } S \geq K_6, \\ F(1 + d), & \text{if } D < S < K_6, \\ FS/S_0, & \text{otherwise}. \end{cases} \end{aligned} \tag{12}$$

The prices and the elapsed times using the uniform grids are shown in Fig. 13 and the prices converge to the price of using a non-uniform grid with $N_p = 20$.

Given some tolerance of option prices, we determine the number of uniform grids corresponding to the non-uniform grid ($N_p = 20$) from our algorithm in one asset. Table 1 lists the results with the cash-or-nothing option and the ELS option. It demonstrates that using non-uniform grid 20 points yields equivalent results using uniform grids with more points for each product. We also confirm that the computational cost is significantly reduced by using the optimal non-uniform grid.

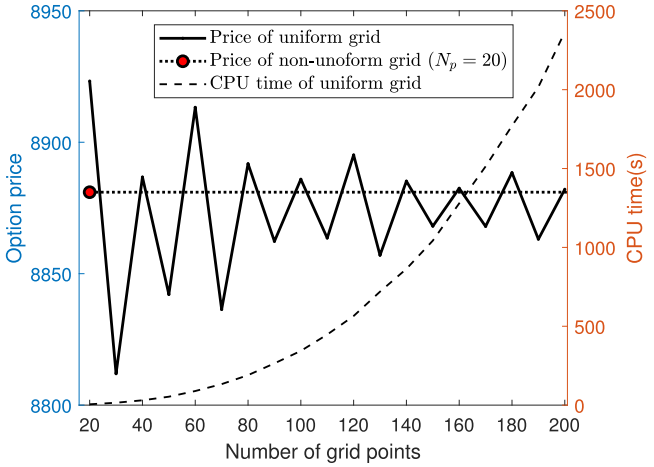

Fig. 13. Option price with a non-uniform grid of $N_p = 20$ and uniform grids from $N_x = N_y = N_z = 19$ to $N_x = N_y = N_z = 200$. CPU time is denoted in seconds.

Table 1
Comparing the performance of the optimal non-uniform grid with 20 points to the uniform grids.

Product	Uniform (N_x)	CPU time ratio	Relative error
Two-asset cash-or-nothing	300	77.24	0.0135
Three-asset cash-or-nothing	200	592.86	0.0279
Two-asset ELS	300	115.05	0.0018
Three-asset ELS	200	391.68	0.0001

3.3. Comparison test

In this section, we compare the results using the proposed algorithm with the results of the cash-or-nothing option of one underlying asset using two different non-uniform grids. We use the reference price u_{ref}, solved by a uniform grid with a space step size of 1 and parameters as used in Section 3.1. The authors in [27] considered the following distribution of the grid points for option pricing:

$$x_i = \begin{cases} xL + d\sinh(\xi_i), & \xi_{min} \le \xi_i < 0, \\ xL + d\xi_i, & 0 \le \xi_i \le \xi_{int}, \\ xR + d\sinh(\xi_i - \xi_{int}), & \xi_{int} < \xi_i \le \xi_{max}, \end{cases} \qquad 1 \le i \le N_x \tag{13}$$

where $\sinh(\cdot)$ denotes the hyperbolic sine function, $\xi_{min} = \sinh^{-1}(L_x - xL/d)$, $\xi_{int} = (xR - Lx)/d$, $\xi_{max} = \xi_{int} + \sinh^{-1}((R_x - xR)/d)$, $d = K/20$, $xL = K\max(0.5, \exp(-0.25T))$, $xR = K$, and so that $[xL, xR] \subset [L_x, R_x]$. In [1], the authors used the following non-uniform grid for option pricing:

$$x_i = \frac{1}{\xi}\sinh(s_i \sinh^{-1}(\xi(L - K)) - (1 - s_i)\sinh^{-1}(\xi K)) + K, \tag{14}$$

where $\xi = 0.5$ and s_i are uniform nodes on $[0, 1]$ for $1 \le i \le N_x$.

Fig. 14 shows the distribution of the non-uniform grid by Eq. (13), Eq. (14) and the proposed algorithm for one-asset cash-or-nothing option with $N_p = 20$.

Table 2 lists the relative errors using the three different non-uniform grids with $N_p = 20, 30, \ldots, 100$. We can recognize that as the number of grid points increases, the relative error converges to zero, as shown in Fig. 15. The proposed algorithm, however, performs well even with fewer grid points compared to the other two methods.

4. Conclusions

The BS equation using FDM is generally solved with a uniform grid of underlying assets. However, using a uniform grid is not an efficient method to evaluate option prices as it incurs a high computational cost when pricing

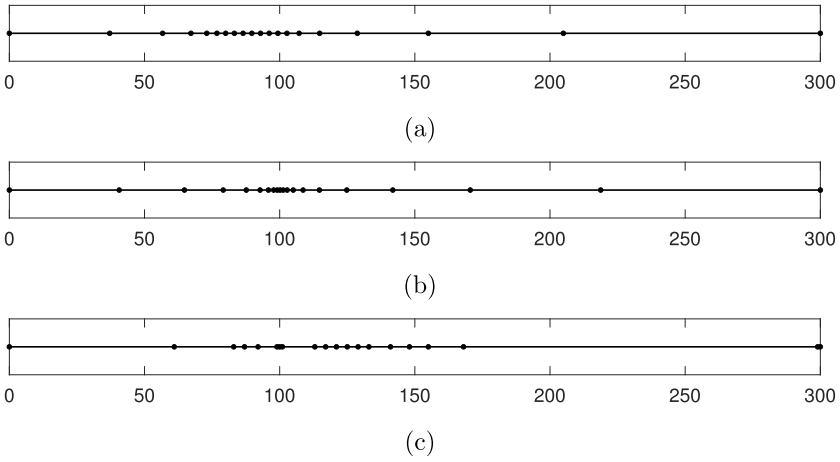

Fig. 14. Distribution of non-uniform grid points with $N_p = 20$, (a) grid generated by Eq. (13), (b) grid generated by Eq. (14), and (c) proposed grid points for one-asset cash-or-nothing option.

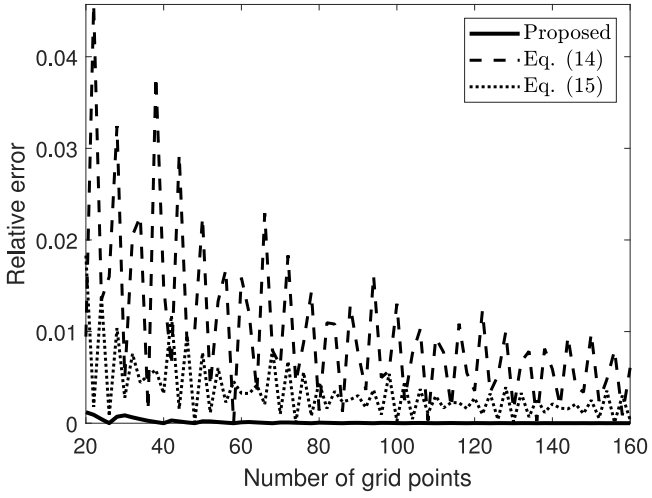

Fig. 15. Relative errors of three different non-uniform grids with the number of grid points $20, \ldots, 160$.

Table 2
Relative errors of three different non-uniform grids with $N_p = 20, 30, \ldots, 100$.

N_p	Proposed	Eq. (13)	Eq. (14)
20	0.00120205	0.00944987	0.01829768
30	0.00000068	0.01497975	0.00325043
40	0.00010051	0.01588143	0.00326732
50	0.00005997	0.00140081	0.00446850
60	0.00001433	0.01303634	0.00022335
70	0.00001480	0.00353232	0.00273891
80	0.00000402	0.00589805	0.00210764
90	0.00000329	0.00604714	0.00036742
100	0.00000289	0.00176875	0.00233099

options with three underlying assets. Generally, financial derivatives are composed of multiple assets. Therefore, calculating the price of these derivatives using FDM is difficult. For this reason, we need to obtain a non-uniform grid with fewer grid points that also produces a result similar to that of a uniform grid with many points. We

proposed an algorithm that finds an optimal non-uniform grid from a uniform grid by removing the node with the smallest relative error. The computational results confirmed the accuracy and efficiency of our proposed method. From this study, we obtained an optimal non-uniform grid that expedites pricing multi-asset options, including three-asset ELS. We also obtain the optimal non-uniform grid for particular financial derivatives. If these products have different parameters, then the optimal non-uniform grid formed from our proposed algorithm is also different. Thus, in future work, we will intend to obtain the optimal non-uniform grid in many different types of parameter cases.

Acknowledgments

This work was supported by the Brain Korea 21 FOUR (BK 21 FOUR) from the Ministry of Education of Korea. The corresponding author (J.S. Kim) was supported by Korea University Research Fund. The authors appreciate the reviewers for their constructive comments, which have improved the quality of this paper.

References

[1] Z. Al-Zhour, M. Barfeie, F. Soleymani, E. Tohidi, A computational method to price with transaction costs under the nonlinear black–scholes model, Chaos Solitons Fractals 127 (2019) 291–301.
[2] J. Bodeau, G. Riboulet, T. Roncalli, Non-uniform grids for PDE in finance, SSRN Electron. J. (2000) http://dx.doi.org/10.2139/ssrn.1031941.
[3] M. Bustamante, M. Contreras, Multi-asset black–scholes model as a variable second class constrained dynamical system, Physica A 457 (2016) 540–572.
[4] W. Chen, S. Wang, A 2nd-order ADI finite difference method for a 2D fractional black–scholes equation governing European two asset option pricing, Math. Comput. Simulation 171 (2020) 279–293.
[5] Y. Chen, W. Wang, A. Xiao, An efficient algorithm for options under merton's jump-diffusion model on nonuniform grids, Comput. Econ. 53 (4) (2019) 1565–1591.
[6] Y. Choi, D. Jeong, J. Kim, Y.R. Kim, S. Lee, S. Seo, M. Yoo, Robust and accurate method for the black–scholes equations with payoff-consistent extrapolation, Commun. Korean Math. Soc. 30 (3) (2015) 297–311.
[7] M.J. Dilloo, D.Y. Tangman, A high-order finite difference method for option valuation, Comput. Math. Appl. 74 (4) (2017) 652–670.
[8] S. Fadugba, C. Nwozo, T. Babalola, The comparative study of finite difference method and Monte Carlo method for pricing European option, Math. Theory Model. 2 (4) (2012) 60–67.
[9] S. Gulen, C. Popescu, M. Sari, A new approach for the black–scholes model with linear and nonlinear volatilities, Mathematics 7 (8) (2019) 760.
[10] J. Hu, S. Gan, High order method for black–scholes PDE, Comput. Math. Appl. 75 (7) (2018) 2259–2270.
[11] M. Huang, G. Huo, A simple and efficient numerical method for pricing discretely monitored early-exercise options, 2019, arXiv preprint, arXiv:1905.13407.
[12] D. Jeong, S. Seo, H. Hwang, D. Lee, Y. Choi, J. Kim, Accuracy, robustness, and efficiency of the linear boundary condition for the black–scholes equations, Discrete Dyn. Nat. Soc. 2015 (2015).
[13] D. Jeong, I.S. Wee, J. Kim, An operator splitting method for pricing the ELS option, J. Korean Soc. Ind. Appl. Math. 14 (3) (2010) 175–187.
[14] D. Jeong, M. Yoo, C. Yoo, J. Kim, A hybrid Monte Carlo and finite difference method for option pricing, Comput. Econ. 53 (1) (2019) 111–124.
[15] M.K. Kadalbajoo, A. Kumar, L.P. Tripathi, Radial-basis-function-based finite difference operator splitting method for pricing American options, Int. J. Comput. Math. 95 (11) (2018) 2343–2359.
[16] R. Kangro, R. Nicolaides, Far field boundary conditions for black–scholes equations, SIAM J. Numer. Anal. 38 (4) (2000) 1357–1368.
[17] L. Khodayari, M. Ranjbar, A computationally efficient numerical approach for multi-asset option pricing, Int. J. Comput. Math. 96 (6) (2019) 1158–1168.
[18] J. Kim, T. Kim, J. Jo, Y. Choi, S. Lee, H. Hwang, M. Yoo, D. Jeong, A practical finite difference method for the three-dimensional black–scholes equation, European J. Oper. Res. 252 (1) (2016) 183–190.
[19] C.C.W. Leentvaar, Pricing Multi-Asset Options with Sparse Grids (Ph.D. thesis), Delft University of Technology, Delft, Netherlands, 2008.
[20] W. Liu, Block-centred finite difference methods on rectangular composite grids with refinement in space for parabolic equation, Int. J. Comput. Math. 97 (3) (2020) 564–584.
[21] A. Mijatović, M. Pistorius, Continuously monitored barrier options under Markov processes, Math. Finance 23 (1) (2013) 1–38.
[22] S. Milovanović, L. von Sydow, Radial basis function generated finite differences for option pricing problems, Comput. Math. Appl. 75 (4) (2018) 1462–1481.
[23] S. Milovanović, L. von Sydow, A high order method for pricing of financial derivatives using radial basis function generated finite differences, Math. Comput. Simulation 174 (2020) 205–217.
[24] C. Mishra, X. Lu, A case study on pricing foreign exchange options using the modified Craig–Sneyd ADI scheme, Int. J. Comput. Math. 97 (6) (2020) 1320–1338.

[25] A. Shojaei, F. Mossaiby, M. Zaccariotto, U. Galvanetto, An adaptive multi-grid peridynamic method for dynamic fracture analysis, Int. J. Mech. Sci. 144 (2018) 600–617.
[26] F. Soleymani, M. Barfeie, Pricing options under stochastic volatility jump model: A stable adaptive scheme, Appl. Numer. Math. 145 (2019) 69–89.
[27] F. Soleymani, B.N. Saray, Pricing the financial Heston–Hull–White model with arbitrary correlation factors via an adaptive FDM, Comput. Math. Appl. 77 (4) (2019) 1107–1123.
[28] M.Z. Ullah, Numerical solution of Heston–Hull–White three-dimensional PDE with a high order FD scheme, Mathematics 7 (8) (2019) 704.
[29] G. Zhang, L. Li, Analysis of Markov chain approximation for option pricing and hedging: Grid design and convergence behavior, Oper. Res. 67 (2) (2019) 407–427.