

Article

# Super-Fast Computation for the Three-Asset Equity-Linked Securities Using the Finite Difference Method

Chaeyoung Lee <sup>1</sup>, Jisang Lyu <sup>1</sup>, Eunghae Park <sup>1</sup>, Wonjin Lee <sup>2</sup>, Sangkwon Kim <sup>1</sup>, Darae Jeong <sup>3</sup>  
and Junseok Kim <sup>1,\*</sup>

<sup>1</sup> Department of Mathematics, Korea University, Seoul 02841, Korea; chae1228@korea.ac.kr (C.L.); lyujis96@korea.ac.kr (J.L.); 012dmsco@korea.ac.kr (E.P.); ksk8863@korea.ac.kr (S.K.)

<sup>2</sup> Department of Financial Engineering, Korea University, Seoul 02841, Korea; wonjin381@korea.ac.kr

<sup>3</sup> Department of Mathematics, Kangwon National University, Gangwon-do 24341, Korea; tinayoyo@kangwon.ac.kr

\* Correspondence: cfdkim@korea.ac.kr

Received: 2 February 2020; Accepted: 24 February 2020; Published: 26 February 2020



**Abstract:** In this article, we propose a super-fast computational algorithm for three-asset equity-linked securities (ELS) using the finite difference method (FDM). ELS is a very popular investment product in South Korea. There are one-, two-, and three-asset ELS. The three-asset ELS is the most popular financial product among them. FDM has been used for pricing the one- and two-asset ELS because it is accurate. However, the three-asset ELS is still priced using the Monte Carlo simulation (MCS) due to the curse of dimensionality for FDM. To overcome the limitation of dimension for FDM, we propose a systematic non-uniform grid with an explicit Euler scheme and an optimal implementation of the algorithm. The computational time is less than 6 s. We perform standard ELS option pricing and compare the results from the fast FDM with the ones from MCS. The computational results confirm the superiority and practicality of the proposed algorithm.

**Keywords:** super-fast computation; Equity-linked securities; Black–Scholes equations; finite difference method

---

## 1. Introduction

The equity-linked security (ELS) is the financial derivative whose payoff relies on the performance of the underlying assets. It is the most common financial instrument traded in South Korea. Thus, valuation of ELS instruments is an important issue. In the financial market, most ELS instruments are products with three underlying assets. The structure of the products has become complex as the number of underlying assets of the ELS increase. Three-asset step-down ELS is an option that is automatically exercised depending on the condition of each product, that is, a kind of autocallable structured product [1]. Whether the option is exercised or not is determined at an early redemption date that is before or at maturity.

Generally, maturity is three years, the early redemption date is every six-months until maturity, and the price used as a criterion is the lowest price among three underlying assets. If the price is above the designated amount, which is called the strike price at the redemption dates, the option is automatically exercised with a coupon rate. If the early redemption did not occur and the prices of the underlying assets

are all above the knock-in barrier until maturity, the option payoff ends with a dummy rate. Otherwise, the option payoff ends at a loss of the amount of the option price at that time. The step-down indicates a decrease in strike price compared to the previous redemption date. Pricing an autocallable product like ELS is an important issue, and not only in Korea, as many options in the world have early-exercise features.

In general, the closed-form solution for pricing options is not available due to the complexity and diversity of their structures. Therefore, one needs to approximate the solution using numerical methods, such as Monte Carlo simulation (MCS) [2,3], the lattice method [4], the finite difference method (FDM) [5–7], finite element method [8,9], or finite volume method [10] for the Black–Scholes (BS) equation. MCS has the advantage of a lower calculation cost and less dependence on dimensions. Many studies have been developed using MCS as it has been considered more suitable than FDM in high dimensional settings [11]. To overcome the high computational cost of using FDM, Leentvaar and Oosterlee [12] used a sparse grid. They observed a satisfactory convergence on relatively coarse grids.

In [13], Ikonen and Toivanen used the operator splitting method (OSM) for pricing American options with stochastic volatility. They compared the computational cost of the projected successive over relaxation method with OSM and observed that OSM is much faster. However, using OSM still has a problem with regard to high computational cost when it is applied to multi-dimensional models. Heinecke et al. [14] priced European and American basket options with FDM. They used a non-uniform sparse grid to reduce the computational time on high dimensions. They also used parallelization to reduce memory costs. Gulen et al. [7] computed European put option values with a higher order scheme to produce a more reliable result. They also compared the CPU time of linear and nonlinear options depending on the number of discretization points of a domain. The results clearly show a higher computation cost as the number of discretization points increases. Guillaume provided an analytical formula for a flexible autocallable structure with discrete observation dates [15] and solved the multi-dimensional BS equation analytically [16], which alleviated the curse of dimensionality.

In [17], Yoo et al. compared the implicit method and explicit finite difference method for pricing ELS. However, they used the modified BS equation using log-transformation and applied a non-systematically made non-uniform grid. In [18], the authors introduced a super-time-stepping technique for solving FDM explicitly when pricing European and American put options. The result demonstrated the efficiency of the proposed technique. High accuracy with a low computational cost was obtained compared to conventional implicit techniques. Boyle and Tian [19] used explicit FDM when pricing different types of barrier options. An adjusted grid was used to better accommodate the boundary conditions of the problem. To save the computational time when pricing ELS, the authors in [20] obtained a non-uniform grid by trial and error. They used an implicit scheme which requires solving a tridiagonal system of discrete equations.

In this paper, we propose a systematic non-uniform grid generation depending on the given different parameters of each contract. The simple and explicit Euler scheme is used with the generated non-uniform grid, which demonstrates an optimal implementation of the algorithm for a super-fast three-asset ELS option pricing.

The outline of this paper is as follows. In Section 2, we introduce the BS equation, present an example of the step-down ELS structure with three underlying assets, and describe the numerical solution algorithms of the governing equation. In Section 3, we present the computational experiments to confirm that the proposed algorithm is fast and accurate. Finally, our conclusions are drawn in Section 4.

## 2. Numerical Method

### 2.1. Three-Dimensional Black–Scholes Equation

Let  $u(x, y, z, \tau)$  be the option price, where  $x, y,$  and  $z$  are the prices of the three underlying assets and  $\tau = T - t$  is the time to maturity. For  $(x, y, z) \in \Omega$  and  $\tau \in (0, T]$ , the BS equation is given as

$$\begin{aligned}
 u_\tau(x, y, z, \tau) = & rxu_x(x, y, z, \tau) + ryu_y(x, y, z, \tau) + rzu_z(x, y, z, \tau) + \frac{1}{2}\sigma_x^2x^2u_{xx}(x, y, z, \tau) \\
 & + \frac{1}{2}\sigma_y^2y^2u_{yy}(x, y, z, \tau) + \frac{1}{2}\sigma_z^2z^2u_{zz}(x, y, z, \tau) + \rho_{xy}\sigma_x\sigma_yxyu_{xy}(x, y, z, \tau) \\
 & + \rho_{yz}\sigma_y\sigma_zyzu_{yz}(x, y, z, \tau) + \rho_{zx}\sigma_x\sigma_zxz u_{zx}(x, y, z, \tau) - ru(x, y, z, \tau), \tag{1}
 \end{aligned}$$

where  $\Omega$  is the computational domain,  $r$  is the risk-free interest rate,  $\sigma_x, \sigma_y,$  and  $\sigma_z$  are the volatilities of the underlying assets  $x, y,$  and  $z,$  respectively, and  $\rho_{xy}, \rho_{yz},$  and  $\rho_{zx}$  are the correlation values between each two underlying assets. More details about the equation can be found in [20].

### 2.2. Step-Down Type ELS

Figure 1 illustrates the payoff structure for a one-asset step-down ELS, which has five early repayments and maturity repayment. We denote the underlying asset by  $S,$  strike prices by  $K_i,$  coupon rates by  $c_i,$  knock-in-barrier by  $D,$  face value by  $F,$  and dummy rate by  $d.$  Here,  $i$  is the index of early repayment and maturity repayment (i.e.,  $i = 1, \dots, 6$ ).

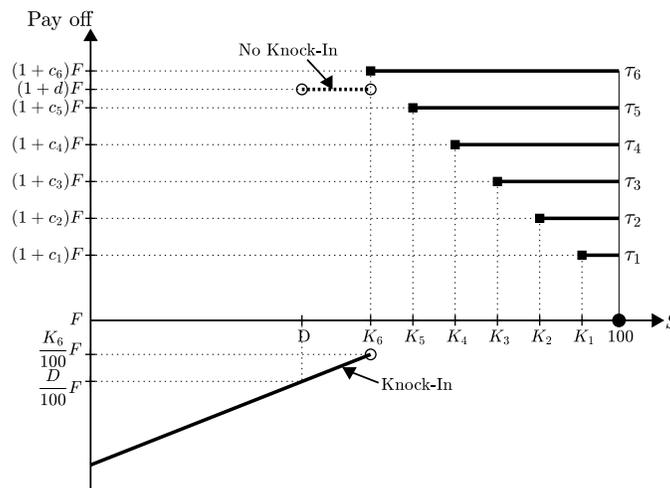


Figure 1. Payoff structure of a one-asset step-down ELS.

The step-down type ELS has nine possible cases as shown in Figure 2. Cases ①–⑥ represent early repayment and maturity repayment. If the repayment terms are satisfied at the time of each repayment, the corresponding coupon is paid and the contract is closed. Case ⑦ represents the case when the dummy is paid if the stock path does not hit the knock-in barrier throughout the contract period and does not satisfy early redemptions. In the cases of ⑧ and ⑨, the principal is lost because the stock path hits the knock-in barrier.

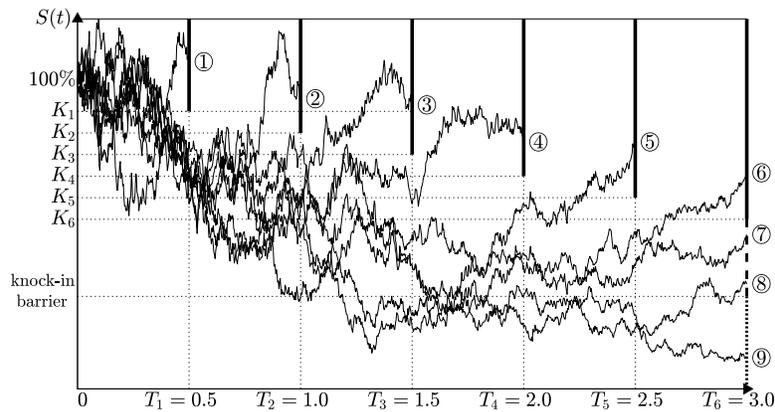


Figure 2. Nine cases of random stock path for the step-down type ELS.

In this paper, we consider the pricing of three-asset ELS options. Let  $S_t = \min(x_t, y_t, z_t)$  be the minimum of the three underlying assets. Let  $u(S, \tau)$  be the solution when the asset price has been below the knock-in barrier and  $v(S, \tau)$  be the solution when not. The initial payoff functions of ELS option are

$$u(S, 0) = \begin{cases} (S_T/S_0)F, & \text{if } S_T < K_6, \\ (1 + c_6)F, & \text{otherwise,} \end{cases} \tag{2}$$

$$v(S, 0) = \begin{cases} (1 + c_6)F, & \text{if } S_T \geq K_6, \\ (1 + d)F, & \text{if } D < S_T < K_6, \\ (S_T/S_0)F, & \text{otherwise.} \end{cases} \tag{3}$$

We overlay  $v(S, \tau)$  with  $u(S, \tau)$  under the knock-in barrier before solving  $v(S, \tau)$ . Figure 3 shows the schematic diagram of the differences between  $u$  and  $v$  at  $\tau = 2.5$ . More details regarding ELS structure can be found in [20].

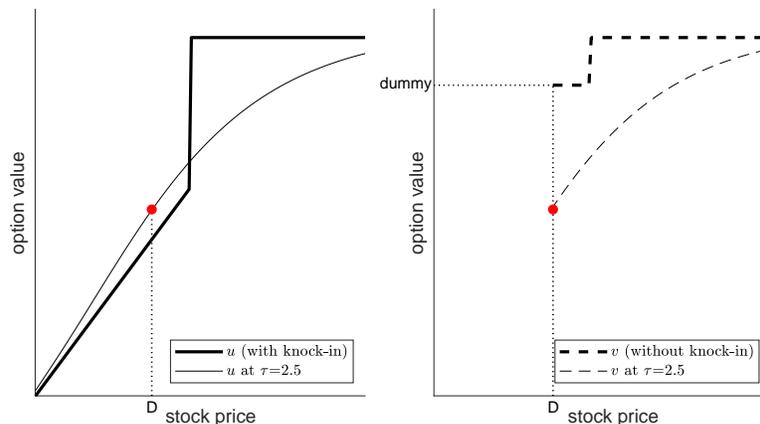


Figure 3. Schematic illustration of  $u$  and  $v$  at  $\tau = 2.5$ .

### 2.3. Solution Algorithm

First, we discretize  $\Omega = [0, L] \times [0, M] \times [0, N]$  with steps  $h_{i-1}^x = x_i - x_{i-1}, h_{j-1}^y = y_j - y_{j-1}$ , and  $h_{k-1}^z = z_k - z_{k-1}$ . Here,  $x_0 = y_0 = z_0 = 0, x_{N_x} = L, y_{N_y} = M$ , and  $z_{N_z} = N$ .  $N_x, N_y, N_z$ , and  $N_\tau$  are the numbers of grid points in the  $x$ -,  $y$ -,  $z$ -, and  $\tau$ -directions, respectively.  $\Delta\tau = T/N_\tau$  is a time step size. Let  $u_{ijk}^n \equiv u(x_i, y_j, z_k, n\Delta\tau)$  be the computational approximation of the solution, where  $i = 0, \dots, N_x$ ,

$j = 0, \dots, N_y, k = 0, \dots, N_z$ , and  $n = 0, \dots, N_\tau$ . We use the first and second order partial derivatives at the discrete point  $(x_i, y_j, z_k)$  as follows:

$$\begin{aligned}
 (D_x u)_{ijk} &= -\frac{h_i^x}{h_{i-1}^x(h_{i-1}^x + h_i^x)} u_{i-1,jk} + \frac{h_i^x - h_{i-1}^x}{h_{i-1}^x h_i^x} u_{ijk} + \frac{h_{i-1}^x}{h_i^x(h_{i-1}^x + h_i^x)} u_{i+1,jk}, \\
 (D_y u)_{ijk} &= -\frac{h_j^y}{h_{j-1}^y(h_{j-1}^y + h_j^y)} u_{i,j-1,k} + \frac{h_j^y - h_{j-1}^y}{h_{j-1}^y h_j^y} u_{ijk} + \frac{h_{j-1}^y}{h_j^y(h_{j-1}^y + h_j^y)} u_{i,j+1,k}, \\
 (D_z u)_{ijk} &= -\frac{h_k^z}{h_{k-1}^z(h_{k-1}^z + h_k^z)} u_{ij,k-1} + \frac{h_k^z - h_{k-1}^z}{h_{k-1}^z h_k^z} u_{ijk} + \frac{h_{k-1}^z}{h_k^z(h_{k-1}^z + h_k^z)} u_{ij,k+1}, \\
 (D_{xx} u)_{ijk} &= \frac{2}{h_{i-1}^x(h_{i-1}^x + h_i^x)} u_{i-1,jk} - \frac{2}{h_{i-1}^x h_i^x} u_{ijk} + \frac{2}{h_i^x(h_{i-1}^x + h_i^x)} u_{i+1,jk}, \\
 (D_{yy} u)_{ijk} &= \frac{2}{h_{j-1}^y(h_{j-1}^y + h_j^y)} u_{i,j-1,k} - \frac{2}{h_{j-1}^y h_j^y} u_{ijk} + \frac{2}{h_j^y(h_{j-1}^y + h_j^y)} u_{i,j+1,k}, \\
 (D_{zz} u)_{ijk} &= \frac{2}{h_{k-1}^z(h_{k-1}^z + h_k^z)} u_{ij,k-1} - \frac{2}{h_{k-1}^z h_k^z} u_{ijk} + \frac{2}{h_k^z(h_{k-1}^z + h_k^z)} u_{ij,k+1}, \\
 (D_{yx} u)_{ijk} &= \frac{u_{i+1,j+1,k} - u_{i-1,j+1,k} - u_{i+1,j-1,k} + u_{i-1,j-1,k}}{h_i^x h_j^y + h_{i-1}^x h_j^y + h_i^x h_{j-1}^y + h_{i-1}^x h_{j-1}^y}, \\
 (D_{zy} u)_{ijk} &= \frac{u_{i,j+1,k+1} - u_{i,j-1,k+1} - u_{i,j+1,k-1} + u_{i,j-1,k-1}}{h_j^y h_k^z + h_{j-1}^y h_k^z + h_j^y h_{k-1}^z + h_{j-1}^y h_{k-1}^z}, \\
 (D_{xz} u)_{ijk} &= \frac{u_{i+1,j,k+1} - u_{i+1,j,k-1} - u_{i-1,j,k+1} + u_{i-1,j,k-1}}{h_k^z h_i^x + h_{k-1}^z h_i^x + h_k^z h_{i-1}^x + h_{k-1}^z h_{i-1}^x}.
 \end{aligned}$$

We apply the homogenous Dirichlet boundary condition at  $x = 0, y = 0$ , and  $z = 0$ ; and zero Neumann boundary condition at  $x = L, y = M$ , and  $z = N$  [21]. That is,

$$\begin{aligned}
 u_{0jk}^n &= 0, \text{ for } 1 \leq j \leq N_y, 1 \leq k \leq N_z, \\
 u_{i0k}^n &= 0, \text{ for } 1 \leq i \leq N_x, 1 \leq k \leq N_z, \\
 u_{ij0}^n &= 0, \text{ for } 1 \leq i \leq N_x, 1 \leq j \leq N_y, \\
 u_{N_x j k}^n &= u_{N_x-1, j k}^n, \text{ for } 1 \leq j \leq N_y - 1, 1 \leq k \leq N_z - 1, \\
 u_{i N_y k}^n &= u_{i, N_y-1, k}^n, \text{ for } 1 \leq i \leq N_x, 1 \leq k \leq N_z - 1, \\
 u_{ij N_z}^n &= u_{ij, N_z-1}^n, \text{ for } 1 \leq i \leq N_x, 1 \leq j \leq N_y.
 \end{aligned}$$

We first solve Equation (1) explicitly with considering the knock-in event. For  $i = 1, \dots, N_x - 1, j = 1, \dots, N_y - 1$ , and  $k = 1, \dots, N_z - 1$ ,

$$\begin{aligned}
 u_{ijk}^{n+1} &= u_{ijk}^n + \Delta\tau \left( A_{ijk,1} u_{i,j-1,k-1}^n + A_{ijk,2} u_{i-1,j,k-1}^n + A_{ijk,3} u_{ij,k-1}^n + A_{ijk,4} u_{i+1,j,k-1}^n \right. \\
 &\quad + A_{ijk,5} u_{i,j+1,k-1}^n + A_{ijk,6} u_{i-1,j-1,k}^n + A_{ijk,7} u_{i,j-1,k}^n + A_{ijk,8} u_{i+1,j-1,k}^n + A_{ijk,9} u_{i-1,jk}^n \\
 &\quad + A_{ijk,10} u_{ijk}^n + A_{ijk,11} u_{i+1,jk}^n + A_{ijk,12} u_{i-1,j+1,k}^n + A_{ijk,13} u_{i,j+1,k}^n + A_{ijk,14} u_{i+1,j+1,k}^n \\
 &\quad \left. + A_{ijk,15} u_{i,j-1,k+1}^n + A_{ijk,16} u_{i-1,j,k+1}^n + A_{ijk,17} u_{ij,k+1}^n + A_{ijk,18} u_{i+1,j,k+1}^n + A_{ijk,19} u_{i,j+1,k+1}^n \right).
 \end{aligned} \tag{4}$$

Here, we define  $A_{ijk,m}$  for  $m = 1, \dots, 19$  as follows:

$$\begin{aligned}
 A_{ijk,1} &= \frac{\rho_{yz}\sigma_y\sigma_z y_j z_k}{h_j^y h_k^z + h_{j-1}^y h_k^z + h_j^y h_{k-1}^z + h_{j-1}^y h_{k-1}^z}, \quad A_{ijk,2} = \frac{\rho_{zx}\sigma_x\sigma_z z_k x_i}{h_k^z h_i^x + h_{k-1}^z h_i^x + h_k^z h_{i-1}^x + h_{k-1}^z h_{i-1}^x}, \\
 A_{ijk,3} &= \frac{(\sigma_z z_k)^2}{h_{k-1}^z (h_{k-1}^z + h_k^z)} - \frac{r z_k h_k^z}{h_{k-1}^z (h_{k-1}^z + h_k^z)}, \quad A_{ijk,4} = -A_{ijk,2}, \quad A_{ijk,5} = -A_{ijk,1}, \\
 A_{ijk,6} &= \frac{\rho_{xy}\sigma_x\sigma_y x_i y_j}{h_i^x h_j^y + h_{i-1}^x h_j^y + h_i^x h_{j-1}^y + h_{i-1}^x h_{j-1}^y}, \quad A_{ijk,7} = \frac{(\sigma_y y_j)^2}{h_{j-1}^y (h_{j-1}^y + h_j^y)} - \frac{r y_j h_j^y}{h_{j-1}^y (h_{j-1}^y + h_j^y)}, \\
 A_{ijk,8} &= -A_{ijk,6}, \quad A_{ijk,9} = \frac{(\sigma_x x_i)^2}{h_{i-1}^x (h_{i-1}^x + h_i^x)} - \frac{r x_i h_i^x}{h_{i-1}^x (h_{i-1}^x + h_i^x)}, \\
 A_{ijk,10} &= r x_i \frac{h_i^x - h_{i-1}^x}{h_{i-1}^x h_i^x} + r y_j \frac{h_j^y - h_{j-1}^y}{h_{j-1}^y h_j^y} + r z_k \frac{h_k^z - h_{k-1}^z}{h_{k-1}^z h_k^z} - \frac{(\sigma_x x_i)^2}{h_{i-1}^x h_i^x} - \frac{(\sigma_y y_j)^2}{h_{j-1}^y h_j^y} - \frac{(\sigma_z z_k)^2}{h_{k-1}^z h_k^z} - r, \\
 A_{ijk,11} &= \frac{(\sigma_x x_i)^2}{h_i^x (h_{i-1}^x + h_i^x)} + \frac{r x_i h_{i-1}^x}{h_i^x (h_{i-1}^x + h_i^x)}, \quad A_{ijk,12} = -A_{ijk,6}, \\
 A_{ijk,13} &= \frac{(\sigma_y y_j)^2}{h_i^y (h_{j-1}^y + h_j^y)} + \frac{r y_j h_{j-1}^y}{h_j^y (h_{j-1}^y + h_j^y)}, \quad A_{ijk,14} = A_{ijk,6}, \quad A_{ijk,15} = -A_{ijk,1}, \quad A_{ijk,16} = -A_{ijk,2}, \\
 A_{ijk,17} &= \frac{(\sigma_z z_k)^2}{h_k^z (h_{k-1}^z + h_k^z)} + \frac{r z_k h_{k-1}^z}{h_k^z (h_{k-1}^z + h_k^z)}, \quad A_{ijk,18} = A_{ijk,2}, \quad A_{ijk,19} = A_{ijk,1}.
 \end{aligned}$$

Figure 4 illustrates the 19-point discrete stencil for updating  $u_{ijk}^{n+1}$ .

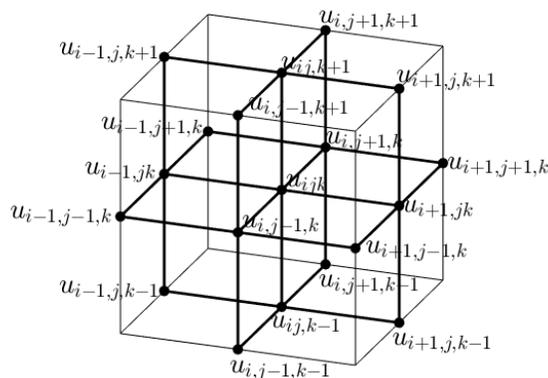


Figure 4. The 19-point discrete stencil for  $u_{ijk}^{n+1}$ .

Next, we set  $v^{n+1}$  at the knock-in-barrier with  $u^{n+1}$  and use it as the Dirichlet boundary condition.  $s$  is where the stock price is equal to the knock-in barrier.

$$\begin{aligned}
 v_{sjk}^n &= u_{sjk}^n, \quad \text{for } s \leq j \leq N_y, \quad s \leq k \leq N_z, \\
 v_{isk}^n &= u_{isk}^n, \quad \text{for } s \leq i \leq N_x, \quad s \leq k \leq N_z, \\
 v_{ijs}^n &= u_{ijs}^n, \quad \text{for } s \leq i \leq N_x, \quad s \leq j \leq N_y.
 \end{aligned}$$

The homogeneous Neumann boundary condition is applied at  $x = L, y = M, z = N$ :

$$\begin{aligned} v_{N_x, jk}^n &= v_{N_x-1, jk}^n, \text{ for } 1 \leq j \leq N_y - 1, 1 \leq k \leq N_z - 1, \\ v_{i, N_y, k}^n &= v_{i, N_y-1, k}^n, \text{ for } 1 \leq i \leq N_x, 1 \leq k \leq N_z - 1, \\ v_{ij, N_z}^n &= v_{ij, N_z-1}^n, \text{ for } 1 \leq i \leq N_x, 1 \leq j \leq N_y. \end{aligned}$$

For  $i = s + 1, \dots, N_x - 1, j = s + 1, \dots, N_y - 1, k = s + 1, \dots, N_z - 1$ , we solve the following

$$\begin{aligned} v_{ijk}^{n+1} &= v_{ijk}^n + \Delta\tau \left( A_{ijk,1} v_{i,j-1,k-1}^n + A_{ijk,2} v_{i-1,j,k-1}^n + A_{ijk,3} v_{ij,k-1}^n + A_{ijk,4} v_{i+1,j,k-1}^n \right. \\ &+ A_{ijk,5} v_{i,j+1,k-1}^n + A_{ijk,6} v_{i-1,j-1,k}^n + A_{ijk,7} v_{i,j-1,k}^n + A_{ijk,8} v_{i+1,j-1,k}^n + A_{ijk,9} v_{i-1,jk}^n \\ &+ A_{ijk,10} v_{ijk}^n + A_{ijk,11} v_{i+1,jk}^n + A_{ijk,12} v_{i-1,j+1,k}^n + A_{ijk,13} v_{ij+1,k}^n + A_{ijk,14} v_{i+1,j+1,k}^n \\ &\left. + A_{ijk,15} v_{i,j-1,k+1}^n + A_{ijk,16} v_{i-1,j,k+1}^n + A_{ijk,17} v_{ij,k+1}^n + A_{ijk,18} v_{i+1,j,k+1}^n + A_{ijk,19} v_{i,j+1,k+1}^n \right). \end{aligned} \tag{5}$$

### 2.4. Weak Condition of Time Step

In this section, we derive the weak conditions of time step under the fully explicit scheme for Equation (4). Since all coefficients of  $u_{ijk}^n$  in Equation (4) are positive, namely,

$$\begin{aligned} A_{ijk,10} + \frac{1}{\Delta\tau} & \tag{6} \\ &= x_i \frac{h_i^x - h_{i-1}^x}{h_{i-1}^x h_i^x} + r y_j \frac{h_j^y - h_{j-1}^y}{h_{j-1}^y h_j^y} + r z_k \frac{h_k^z - h_{k-1}^z}{h_{k-1}^z h_k^z} - \frac{(\sigma_x x_i)^2}{h_{i-1}^x h_i^x} - \frac{(\sigma_y y_j)^2}{h_{j-1}^y h_j^y} - \frac{(\sigma_z z_k)^2}{h_{k-1}^z h_k^z} - r + \frac{1}{\Delta\tau} > 0, \end{aligned}$$

we will use increasing space step sizes, i.e.,  $h_{i-1}^x \leq h_i^x, h_{j-1}^y \leq h_j^y, h_{k-1}^z \leq h_k^z$  for some large  $i, j, k$ . Then, we write Equation (6) as follows:

$$\frac{1}{\Delta\tau} > \frac{(\sigma_x x_i)^2}{h_{i-1}^x h_i^x} + \frac{(\sigma_y y_j)^2}{h_{j-1}^y h_j^y} + \frac{(\sigma_z z_k)^2}{h_{k-1}^z h_k^z} + r.$$

Let  $h_{min} = \min\{h_i^x, h_j^y, h_k^z\}$  for some  $i, j, k$ , then

$$\Delta\tau < \frac{h_{min}^2}{(\sigma_x x_i)^2 + (\sigma_y y_j)^2 + (\sigma_z z_k)^2 + r h_{min}^2}.$$

Due to  $x_{N_x} = L, y_{N_y} = M$ , and  $z_{N_z} = N$ , we have

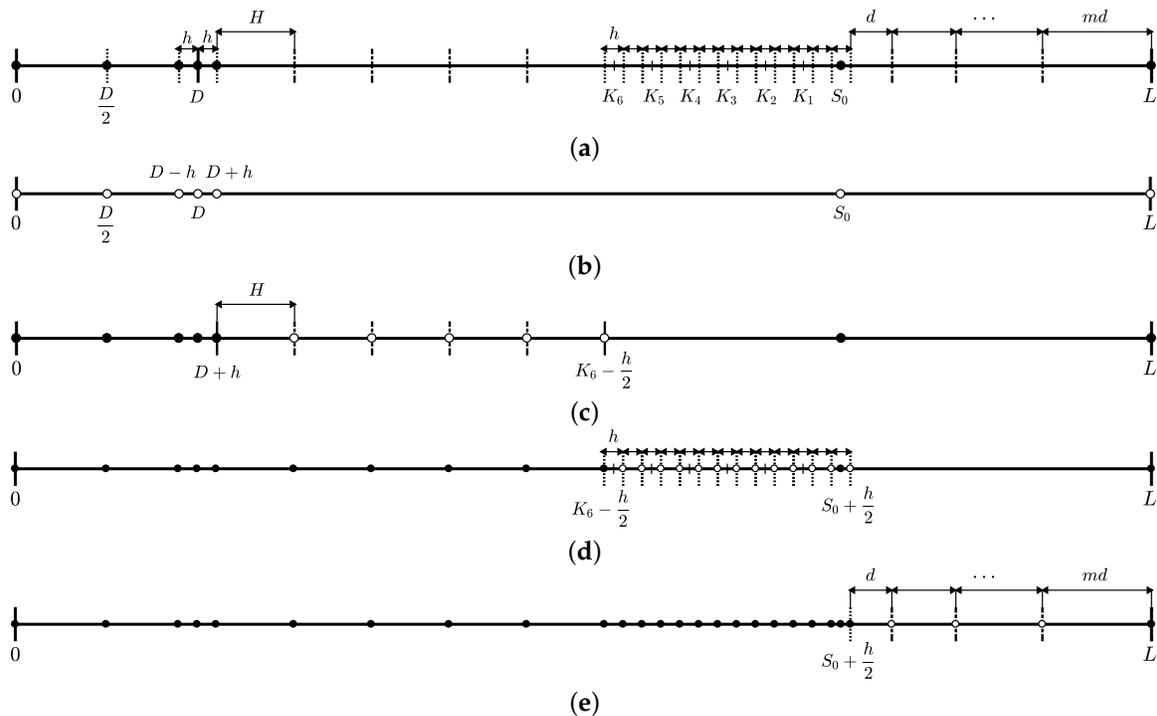
$$\Delta\tau < \frac{h_{min}^2}{(\sigma_x L)^2 + (\sigma_y M)^2 + (\sigma_z N)^2 + r h_{min}^2}. \tag{7}$$

### 3. Numerical Experiments

In this section, we demonstrate the performance of the proposed algorithm using numerical experiments with three types of ELS and comparison with MCS. For all the tests, we use the same parameters: risk-free interest rate  $r = 0.03$ , volatilities  $\sigma_x = \sigma_y = \sigma_z = 0.3$ , correlation  $\rho_{xy} = \rho_{yz} = \rho_{zx} = 0.5$ , the computational domain  $\Omega = [0, L] \times [0, M] \times [0, N]$  with  $L = M = N = 150$ , coupon rates  $[c_1, c_2, c_3, c_4, c_5, c_6] = [0.05, 0.1, 0.15, 0.2, 0.25, 0.3]$ , knock-in-barrier  $D = 50$ , face value  $F = 100$ , and dummy rate  $d = 0.27$ . We test three types of ELS, which have the same parameters as previously mentioned

except for strike percentages. The first type of ELS has strike percentages  $[K_1, K_2, K_3, K_4, K_5, K_6] = [90, 90, 90, 90, 90, 90]$ , the second type has  $[K_1, K_2, K_3, K_4, K_5, K_6] = [95, 95, 90, 90, 85, 85]$ , and the third type has  $[K_1, K_2, K_3, K_4, K_5, K_6] = [95, 90, 85, 80, 75, 70]$ . In other words, the first type of ELS has the same strike percentages at all early repayment dates, the second type has the three different strike percentages, and the last type has a different strike percentage at all repayment dates. Computations were performed on a 2.7 GHz Intel (Intel, Santa Clara, CA, USA) PC with 16 GB of RAM loaded with C language for FDM and with MATLAB 2019b (MathWorks, Natick, MA, USA) for MCS.

We propose a systematic grid generation for fast FDM for pricing a three-asset ELS as shown in Figure 5a.



**Figure 5.** (a) Schematic illustration of the suggested grid  $\Omega_x = \Omega_1 \cup \Omega_2 \cup \Omega_3 \cup \Omega_4$ . (b–e) are schematic illustrations of  $\Omega_1, \Omega_2, \Omega_3$ , and  $\Omega_4$ .

The generation of the non-uniform grid is as follows: Let  $h$  be a given basic grid size.

$$\Omega_1 = \{0, 0.5D, D - h, D, D + h, S_0\},$$

which is the fixed grid points. Let  $H$  be the grid size between  $D + h$  and  $K_6 - 0.5h$ . If we want  $H$  to be approximately  $\tilde{H}$ , then  $H = (K_6 - 1.5h - D) / [(K_6 - 1.5h - D) / \tilde{H} + 1]$ , where  $[\cdot]$  is the floor function. Then, we have the grid point set between  $D + h$  and  $K_6 - 0.5h$ :

$$\Omega_2 = \{D + h + H, D + h + 2H, \dots, K_6 - 0.5h\}.$$

Additionally, we have

$$\Omega_3 = \{K_6 + 0.5h, K_6 + 1.5h, \dots, S_0 - 0.5h, S_0 + 0.5h\},$$

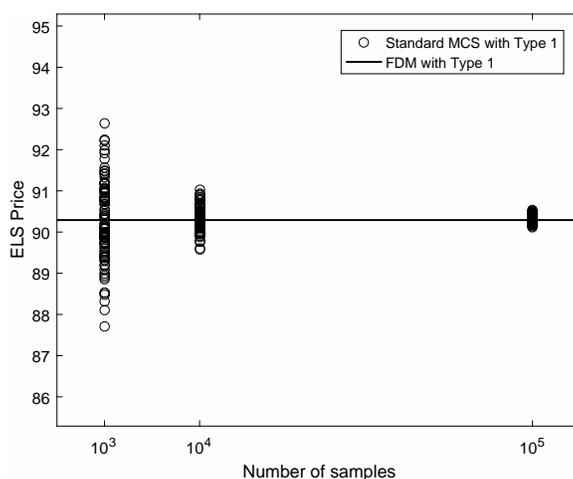
$$\Omega_4 = \{S_0 + 0.5h + d, S_0 + 0.5h + 3d, \dots, L\},$$

where  $d = 2(L - S_0 - 0.5h) / [m(m + 1)]$  for some integer  $m$ , which implies the gap between points next to each other forms an arithmetic sequence. See Figure 5b–e for the schematic illustration of  $\Omega_1, \Omega_2, \Omega_3,$  and  $\Omega_4,$  respectively. Finally, we have a grid set for the  $x$ -axis:

$$\Omega_x = \Omega_1 \cup \Omega_2 \cup \Omega_3 \cup \Omega_4.$$

Let  $\Omega_y = \Omega_x$  and  $\Omega_z = \Omega_x,$  therefore, the discrete domain is  $\Omega_x \times \Omega_y \times \Omega_z.$

Now, we present comparison tests for three types of ELS using standard MCS and fast FDM. For more information regarding the numerical method of MCS for three-asset ELS pricing, refer to [22]. The first type of ELS has strike percentages  $[K_1, K_2, K_3, K_4, K_5, K_6] = [90, 90, 90, 90, 90, 90].$  Figure 6 shows Type 1 ELS MCS price versus the number of samples together with the result of the fast FDM. Here, we plot 100 MCS results for each case. The solid line is the result of the fast FDM.



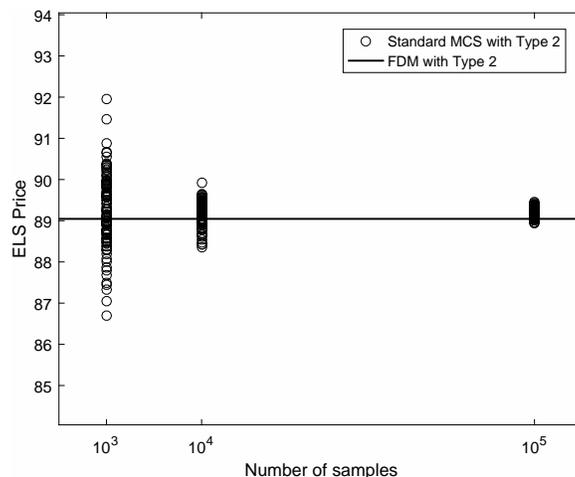
**Figure 6.** Type 1 equity-linked securities (ELS) price versus the number of samples. Here, we plot 100 simulation results for each case.

Table 1 lists the price and computation time of Type 1 ELS with two different methods. For the standard MCS, the price and computation time are obtained with  $10^6$  samples. For the fast FDM, the results are obtained with  $h = 2.5, \tilde{H} = 2.5h, m = 3,$  and  $\Delta\tau = 1/1440.$

**Table 1.** The price and computation time of Type 1 ELS prices with the standard Monte Carlo simulation (MCS) and the fast finite difference method (FDM).

Case	Price	CPU Time (s)
Standard MCS	90.3002	369.4568
Fast FDM	90.2910	2.3900

The second type of ELS has strike percentages  $[K_1, K_2, K_3, K_4, K_5, K_6] = [95, 95, 90, 90, 85, 85].$  Figure 7 shows the Type 2 ELS MCS price versus the number of samples together with the result of the fast FDM, which is represented by the solid line.



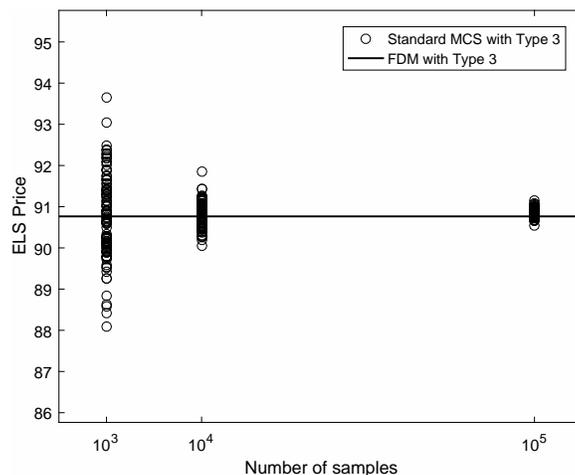
**Figure 7.** The type 2 ELS price versus the number of samples. Here, we plot 100 simulation results for each case.

Table 2 lists the price and computation time of Type 2 ELS with two different methods. For the standard MCS, the price and computation time are obtained with  $10^6$  samples. For the fast FDM, the results are obtained with  $h = 2.5$ ,  $\tilde{H} = 2.5h$ ,  $m = 3$ , and  $\Delta\tau = 1/1440$ .

**Table 2.** Price and computation time of Type 2 ELS prices with the standard MCS and the fast FDM.

Case	Price	CPU Time (s)
Standard MCS	89.1673	386.3436
Fast FDM	89.0452	3.3280

The third type of ELS has strike percentages  $[K_1, K_2, K_3, K_4, K_5, K_6] = [95, 90, 85, 80, 75, 70]$ . Figure 8 shows the Type 3 ELS MCS price versus the number of samples together with the result of the fast FDM, which is represented by the solid line.



**Figure 8.** Type 3 ELS price versus the number of samples. Here, we plot 100 simulation results for each case.

Table 3 lists the price and computation time of Type 3 ELS with two different methods. For the standard MCS, the price and computation time are obtained with  $10^6$  samples. For the fast FDM, the results are obtained with  $h = 2.5$ ,  $\tilde{H} = 2.5h$ ,  $m = 3$ , and  $\Delta\tau = 1/1440$ . From these three types of ELS results, we can confirm that the proposed explicit FDM is fast and accurate.

**Table 3.** Price and computation time of Type 3 ELS prices with the standard MCS and the fast FDM.

Case	Price	CPU Time (s)
Standard MCS	90.8376	369.9680
Fast FDM	90.7650	5.1090

Finally, we compare our method with the previous work [20], which used an operator splitting method (OSM) to verify that the proposed algorithm is fast and accurate. In the OSM, we solve the governing equations with multiple steps:

$$\frac{u_{ijk}^{n+1} - u_{ijk}^n}{\Delta\tau} = (\mathcal{L}_{BS}^x u)_{ijk}^{n+\frac{1}{3}} + (\mathcal{L}_{BS}^y u)_{ijk}^{n+\frac{2}{3}} + (\mathcal{L}_{BS}^z u)_{ijk}^{n+1}, \tag{8}$$

where  $\mathcal{L}_{BS}^x$ ,  $\mathcal{L}_{BS}^y$ , and  $\mathcal{L}_{BS}^z$  are given as

$$\begin{aligned} (\mathcal{L}_{BS}^x u)_{ijk}^{n+\frac{1}{3}} &= \frac{(\sigma_x x_i)^2}{2} D_{xx} u_{ijk}^{n+\frac{1}{3}} + r x_i D_x u_{ijk}^{n+\frac{1}{3}} + \frac{1}{3} \sigma_x \sigma_y \rho_{xy} x_i y_j D_{xy} u_{ijk}^n \\ &\quad + \frac{1}{3} \sigma_y \sigma_z \rho_{yz} y_j z_k D_{yz} u_{ijk}^n + \frac{1}{3} \sigma_z \sigma_x \rho_{zx} z_k x_i D_{zx} u_{ijk}^n - \frac{1}{3} r u_{ijk}^{n+\frac{1}{3}}, \end{aligned} \tag{9}$$

$$\begin{aligned} (\mathcal{L}_{BS}^y u)_{ijk}^{n+\frac{2}{3}} &= \frac{(\sigma_y y_j)^2}{2} D_{yy} u_{ijk}^{n+\frac{2}{3}} + r y_j D_y u_{ijk}^{n+\frac{2}{3}} + \frac{1}{3} \sigma_x \sigma_y \rho_{xy} x_i y_j D_{xy} u_{ijk}^{n+\frac{1}{3}} \\ &\quad + \frac{1}{3} \sigma_y \sigma_z \rho_{yz} y_j z_k D_{yz} u_{ijk}^{n+\frac{1}{3}} + \frac{1}{3} \sigma_z \sigma_x \rho_{zx} z_k x_i D_{zx} u_{ijk}^{n+\frac{1}{3}} - \frac{1}{3} r u_{ijk}^{n+\frac{2}{3}}, \end{aligned} \tag{10}$$

$$\begin{aligned} (\mathcal{L}_{BS}^z u)_{ijk}^{n+1} &= \frac{(\sigma_z z_k)^2}{2} D_{zz} u_{ijk}^{n+1} + r z_k D_z u_{ijk}^{n+1} + \frac{1}{3} \sigma_x \sigma_y \rho_{xy} x_i y_j D_{xy} u_{ijk}^{n+\frac{2}{3}} \\ &\quad + \frac{1}{3} \sigma_y \sigma_z \rho_{yz} y_j z_k D_{yz} u_{ijk}^{n+\frac{2}{3}} + \frac{1}{3} \sigma_z \sigma_x \rho_{zx} z_k x_i D_{zx} u_{ijk}^{n+\frac{2}{3}} - \frac{1}{3} r u_{ijk}^{n+1}. \end{aligned} \tag{11}$$

We solve the below equations, one by one, using the Thomas algorithm:

$$\frac{u_{ijk}^{n+\frac{1}{3}} - u_{ijk}^n}{\Delta\tau} = (\mathcal{L}_{BS}^x u)_{ijk}^{n+\frac{1}{3}}, \quad \frac{u_{ijk}^{n+\frac{2}{3}} - u_{ijk}^{n+\frac{1}{3}}}{\Delta\tau} = (\mathcal{L}_{BS}^y u)_{ijk}^{n+\frac{2}{3}}, \quad \frac{u_{ijk}^{n+1} - u_{ijk}^{n+\frac{2}{3}}}{\Delta\tau} = (\mathcal{L}_{BS}^z u)_{ijk}^{n+1}. \tag{12}$$

For more details about the numerical procedure, please refer to [20]. For comparison, we take the Type 3 ELS with all the same parameter values. The absolute errors of the computed option values with the reference MCS value are 0.0726 and 0.3409 for the proposed method and the OSM, respectively. We achieve a more accurate result. Furthermore, the CPU time for the OSM is 74.2174, which is about 14 times longer than the proposed method.

#### 4. Conclusions

In this paper, we presented a super-fast computational algorithm for three-asset ELS using FDM, which is the most popular investment product in South Korea. To overcome the limitations of

dimension for FDM, we developed a systematic non-uniform grid with an explicit Euler scheme and an optimal implementation of the algorithm. The computational time is less than 6 s, which is very fast. We performed standard ELS option pricing and compared the results from the fast FDM with the ones from MCS. The computational results confirmed the superiority and practicality of the proposed algorithm. We used the BS model, which assumed interest rates and volatility as constant for simplicity. However, this assumption can cause serious errors when calculating prices as they are not constants in real financial markets. Moreover, there is a limitation of the BS model because the Brownian motion oriented asset pricing is not adequate in the case where the stochastic processes' finite dimensional distributions' tails are heavier, compared to the tails of the elliptic distributions. For future work, we will consider the Heston model [23] and the Heston–Hull–White model [24], which assume the interest rates and volatility as variable and the case that the stochastic processes' finite dimensional distributions' tails are heavier.

**Author Contributions:** Conceptualization, J.K.; methodology, D.J. and J.K.; software, C.L. and J.K.; validation, C.L., J.L., E.P., W.L., and S.K.; formal analysis, S.K. and D.J.; investigation, C.L., J.L., E.P., and W.L.; resources, D.J.; writing—original draft preparation, C.L., J.L., E.P., W.L., J.K.; writing—review and editing, C.L., J.L., and S.K.; visualization, J.L., E.P., and W.L.; supervision, J.K.; funding acquisition, J.K. All authors have read and agreed to the published version of the manuscript.

**Funding:** The corresponding author (Junseok Kim) was supported by the Brain Korea 21 Plus (BK 21) from the Ministry of Education of Korea.

**Acknowledgments:** The authors are grateful to the reviewers for constructive and helpful comments on the revision of this article.

**Conflicts of Interest:** The authors declare no conflicts of interest.

## References

1. Deng, G.; Mallett, J.; McCann, C. Modeling autocallable structured products. In *Derivatives and Hedge Funds*; Palgrave Macmillan: London, UK, 2016; pp. 323–344.
2. Haugh, M.B.; Kogan, L. Pricing American options: A duality approach. *Oper. Res.* **2004**, *52*, 258–270. [[CrossRef](#)]
3. Pagés, G. The Monte Carlo Method and Applications to Option Pricing. In *Numerical Probability*; Springer: Cham, Germany, 2018; pp. 27–47.
4. Moon, K.S.; Kim, H. A multi-dimensional local average lattice method for multi-asset models. *Quant. Finance* **2013**, *13*, 873–884. [[CrossRef](#)]
5. Cen, Z.; Le, A. A robust and accurate finite difference method for a generalized Black–Scholes equation. *J. Comput. Appl. Math.* **2011**, *235*, 3728–3733. [[CrossRef](#)]
6. Jeong, D.; Yoo, M.; Kim, J. Finite difference method for the Black–Scholes equation without boundary conditions. *Comput. Econ.* **2018**, *51*, 961–972. [[CrossRef](#)]
7. Gulen, S.; Popescu, C.; Sari, M. A New Approach for the Black–Scholes Model with Linear and Nonlinear Volatilities. *Mathematics*, **2019**, *7*, 760. [[CrossRef](#)]
8. Zhang, R.; Song, H.; Luan, N. Weak Galerkin finite element method for valuation of American options. *Front. Math. China* **2014**, *9*, 455–476. [[CrossRef](#)]
9. Mehrdoust, F. A new hybrid Monte Carlo simulation for Asian options pricing. *J. Stat. Comput. Simul.* **2015**, *85*, 507–516. [[CrossRef](#)]
10. Wang, S. A novel fitted finite volume method for the Black–Scholes equation governing option pricing. *IMA J. Numer. Anal.* **2014**, *24*, 699–720. [[CrossRef](#)]
11. Ma, J. A stochastic correlation model with mean reversion for pricing multi-asset options. *Asia-Pac Financ Markets* **2009**, *16*, 97–109. [[CrossRef](#)]
12. Leentvaar, C.C.W.; Oosterlee, C.W. Pricing multi-asset options with sparse grids and fourth order finite differences. In *Pricing Multi-Asset Options with Sparse Grids and Fourth Order Finite Differences*; de Castro, A.B., Gómez, D., Quintela, P., Salgado, P., Eds.; Springer: Berlin/Heidelberg, Germany, 2007; pp. 809–814.

13. Ikonen, S.; Toivanen, J. Operator splitting methods for American option pricing. *Appl. Math. Lett.* **2004**, *17*, 809–814. [[CrossRef](#)]
14. Heinecke, A.; Schraufstetter, S.; Bungartz, H.J. A highly parallel Black–Scholes solver based on adaptive sparse grids. *Int. J. Comput. Math.* **2012**, *89*, 1212–1238. [[CrossRef](#)]
15. Guillaume, T. Autocallable structured products. *J. Deriv.* **2015**, *22*, 73–94. [[CrossRef](#)]
16. Guillaume, T. On the multidimensional Black–Scholes partial differential equation. *Ann. Oper. Res.* **2019**, *281*, 229–251. [[CrossRef](#)]
17. Yoo, M.; Jeong, D.; Seo, S.; Kim, J. A comparison study of explicit and implicit numerical methods for the equity-linked securities. *Honam Math. J.* **2015**, *37*, 441–455. [[CrossRef](#)]
18. O’Sullivan, S.; O’Sullivan, C. On the acceleration of explicit finite difference methods for option pricing. *Quant. Finance* **2011**, *11*, 1177–1191. [[CrossRef](#)]
19. Boyle, P.P.; Tian, Y. An explicit finite difference approach to the pricing of barrier options. *Appl. Math. Finance* **1998**, *5*, 17–43. [[CrossRef](#)]
20. Kim, J.; Kim, T.; Jo, J.; Choi, Y.; Lee, S.; Hwang, H.; Yoo, M.; Jeong, D. A practical finite difference method for the three-dimensional Black–Scholes equation. *Eur. J. Oper. Res.* **2016**, *252*, 183–190. [[CrossRef](#)]
21. Choi, Y.; Jeong, D.; Kim, J.; Kim, Y.R.; Lee, S.; Seo, S.; Yoo, M. Robust and accurate method for the Black–Scholes equations with payoff-consistent extrapolation. *Commun. Korean Math. Soc.* **2015**, *30*, 297–311. [[CrossRef](#)]
22. Jo, J.; Kim, Y. Comparison of numerical schemes on multi-dimensional Black–Scholes equations. *Bull. Korean Math. Soc.* **2013**, *50*, 2035–2051. [[CrossRef](#)]
23. Heston, S.L. A closed-form solution for options with stochastic volatility with applications to bond and currency options. *Rev. Financ. Stud.* **1993**, *6*, 327–343. [[CrossRef](#)]
24. Ullah, M.Z. Numerical Solution of Heston–Hull–White Three-Dimensional PDE with a High Order FD Scheme. *Mathematics* **2019**, *7*, 704. [[CrossRef](#)]



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).