European Journal of Operational Research 252 (2016) 183–190

Contents lists available at ScienceDirect

European Journal of Operational Research

journal homepage: www.elsevier.com/locate/ejor

Decision Support

A practical finite difference method for the three-dimensional Black–Scholes equation

Junseok Kim [a], Taekkeun Kim [b], Jaehyun Jo [c], Yongho Choi [a], Seunggyu Lee [a], Hyeongseok Hwang [d], Minhyun Yoo [d], Darae Jeong [a,*]

[a] Department of Mathematics, Korea University, Seoul 136-713, Republic of Korea
[b] HMC Investment Securities, Seoul 150-876, Republic of Korea
[c] Business School, Korea University, Seoul 136-701, Republic of Korea
[d] Department of Financial Engineering, Korea University, Seoul 136-701, Republic of Korea

ARTICLE INFO

Article history:
Received 21 July 2014
Accepted 2 December 2015
Available online 12 December 2015

Keywords:
Option pricing
Equity–linked securities
Black–Scholes partial differential equation
Operator splitting method
Non–uniform grid

ABSTRACT

In this paper, we develop a fast and accurate numerical method for pricing of the three-asset equity-linked securities options. The option pricing model is based on the Black–Scholes partial differential equation. The model is discretized by using a non-uniform finite difference method and the resulting discrete equations are solved by using an operator splitting method. For fast and accurate calculation, we put more grid points near the singularity of the nonsmooth payoff function. To demonstrate the accuracy and efficiency of the proposed numerical method, we compare the results of the method with those from Monte Carlo simulation in terms of computational cost and accuracy. The numerical results show that the cost of the proposed method is comparable to that of the Monte Carlo simulation and it provides more stable hedging parameters such as the Greeks.

1. Introduction

Equity-linked securities (ELS) is one of the most popular investment products whose return is linked to the performance of an underlying equity such as a single stock, a group of stocks, or an equity-based index. The ELS option pricing model is based on the Black–Scholes (BS) partial differential equation (PDE) (Black & Scholes, 1973). For some cases, there exist closed-form solutions of pricing derivatives (Capiński, 2015; Pun, Chung, & Wong, 2015). However, in most cases with complex structures (e.g., ELS), it is inefficient or impossible to derive the exact solutions. To solve the problem, an alternative way is a numerical approach (see, for example, Bandi & Bertsimas, 2014; Han & Wu, 2003; Jeong, Kim, & Wee, 2009; Wilmott, Dewynne, & Howison, 1993; Zvan, Vetzal, & Forsyth, 2000). In numerical approaches, there are several ways such as Monte Carlo simulation (MCS), binomial method, finite difference method, etc.

Among these, MCS is the most commonly used method for pricing complex derivatives because of its simplicity to use. However, it should be run with a large number of random samples to get a reliable option value or calculate its sensitivities which are called by the Greeks (Choi & Song, 2008; González-Parra, Arenas, & Chen-Charpentier, 2013; Marroquı & Moreno, 2013). The Greeks, which play a key role in hedging, are defined as changes in option value relative to changes in each independent variable. It is important to calculate the Greeks accurately for hedging an option. For example, Delta is the rate of change in option price relative to the underlying asset. Therefore, in hedging, the Delta provides the number of short units of the underlying asset which is combined with a call option to offset immediate market risk. That is, the accurate value of the Delta can give a dynamical strategy for hedging against risk. For more details about the Greeks, we refer the reader to Haug (2007).

Finite difference methods (FDM) such as alternating direction implicit (ADI) and operator splitting method (OSM) have been widely used in practice. The reason is that the numerical solution from FDM quickly converges to a stable value unlike MCS. Nevertheless, when pricing ELS with more than three underlying assets, FDM is not practical because it needs a lot of computational resources. Therefore, MCS is the only practical choice for three-asset ELS even though there are drawbacks such as slow convergence and non-deterministic values due to its randomness.

The main purpose of this paper is to develop a fast and accurate FDM for pricing three-asset ELS options and computing its Greeks. In this study, the mathematical model is discretized by using a non-uniform FDM and the resulting discrete equations are solved by

* Corresponding author. Tel.: +82 2 32903077; fax: +82 2 9298562.
E-mail addresses: cfdkim@korea.ac.kr (J. Kim), kz@hmcib.com (T. Kim), jojae511@korea.ac.kr (J. Jo), poohyongho@korea.ac.kr (Y. Choi), sky509@korea.ac.kr (S. Lee), hhs288@korea.ac.kr (H. Hwang), ymh1989@korea.ac.kr (M. Yoo), tinayoyo@korea.ac.kr (D. Jeong).
URL: http://math.korea.ac.kr/~cfdkim (J. Kim)

http://dx.doi.org/10.1016/j.ejor.2015.12.012
0377-2217/

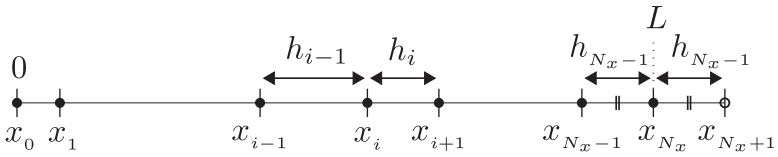

Fig. 1. Schematic representation of the non-equidistant grid on x-axis. The ghost point is defined as $x_{N_x+1} = 2x_{N_x} - x_{N_x-1}$.

means of OSM on adaptive grids with more grid points near non-smooth position of payoff. To show the superiority of the proposed numerical method, we compare the numerical results with those from MCS. We obtain more stable results with our proposed method than MCS in pricing and computing the Greeks of the three-asset ELS option.

The rest of the paper is organized as follows. Section 2 describes the Black–Scholes equation of European options with three underlying assets. Section 3 contains the finite difference discretizations for the BS PDE and a numerical solution algorithm using OSM on adaptive grids. We present the numerical experiments in Section 4. Finally, we conclude our study by summarizing the computational results in Section 5.

2. Black–Scholes model with three underlying assets

The pricing of options on multi-underlying assets is of great interest in the financial industry (Persson & von Persson, 2007; Rambeerich, Tangman, Lollchund, & Bhuruth, 2013). In this paper, we focus on fast and accurate calculations of the three-asset option value and its Greeks.

Let x, y, and z be three underlying assets' prices and t be a time variable. For $(x, y, z) \in \Omega$ and $t \in T$, the option price $u(x, y, z, t)$ follows the BS PDE:

$$\begin{aligned} &u_t(x,y,z,t) + rxu_x(x,y,z,t) + ryu_y(x,y,z,t) + rzu_z(x,y,z,t) \\ &+\frac{1}{2}\sigma_x^2x^2u_{xx}(x,y,z,t) + \frac{1}{2}\sigma_y^2y^2u_{yy}(x,y,z,t) + \frac{1}{2}\sigma_z^2z^2u_{zz}(x,y,z,t) \\ &+\rho_{xy}\sigma_x\sigma_yxyu_{xy}(x,y,z,t) + \rho_{yz}\sigma_y\sigma_zyzu_{yz}(x,y,z,t) \\ &+\rho_{zx}\sigma_x\sigma_zxzu_{zx}(x,y,z,t) - ru(x,y,z,t) = 0 \end{aligned} \tag{1}$$

with the terminal condition $u(x, y, z, T) = \Phi(x, y, z)$, where Φ is the payoff function at the expiration time T (Reisinger & Wittum, 2004). Subscripts x, y, z, and t of u denote the partial derivatives with respect to those variables. σ_x, σ_y, and σ_z are volatilities of underlying assets x, y, and z, respectively. ρ_{xy}, ρ_{yz}, and ρ_{zx} represent the correlation values between two subscript asset variables. r is the constant risk-free interest rate. After we transform the backward-in-time PDE (1) into the forward-in-time PDE by using $\tau = T - t$, we have the initial value problem

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

$$u(x, y, z, 0) = \Phi(x, y, z).$$

3. Numerical method

Let us first discretize the computational domain $\Omega = [0, L] \times [0, M] \times [0, N]$ with positive non-uniform space 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, $x_0 = y_0 = z_0 = 0$, $x_{N_x} = L$, $y_{N_y} = M$, and $z_{N_z} = N$. A time step size is defined as $\Delta\tau = T/N_\tau$. The numbers of grid points in the x-, y-, z-, and τ-directions are denoted by N_x, N_y, N_z, and N_τ, respectively. Fig. 1 represents a schematic illustration of the non-equidistant grid on x-axis. Here, we define the ghost point x_{N_x+1} as $x_{N_x} + h_{N_x-1}$ to apply the linear boundary condition conveniently.

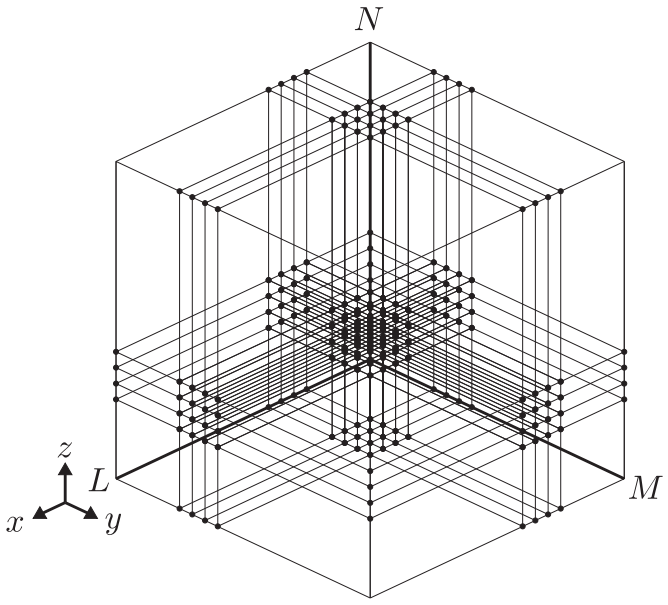

Fig. 2. Schematic representation of non-uniform mesh in three-dimensional space.

Fig. 2 shows a schematic illustration of three-dimensional non-uniform grid structure which will be used in this paper.

Let us denote the numerical approximation of the solution by $u^n_{ijk} \equiv u(x_i, y_j, z_k, n\Delta\tau)$, where $i = 0, \ldots, N_x$, $j = 0, \ldots, N_y$, $k = 0, \ldots, N_z$, and $n = 0, \ldots, N_\tau$. We use the zero Dirichlet boundary conditions at $x = 0$, $y = 0$, and $z = 0$ and the linear boundary conditions at $x = L$, $y = M$, and $z = N$. Now, we apply OSM (Duffy, 2006; Jeong & Kim, 2013) in order to solve Eq. (2). OSM has been used extensively in mathematical finance for solving numerically multi-asset option pricing models. The basic idea of OSM is to divide each time step into fractional time steps with simpler operators.

We consider the following semi-implicit scheme:

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

where the discrete difference operators $\mathcal{L}^x_{BS}$, $\mathcal{L}^y_{BS}$, and $\mathcal{L}^z_{BS}$ are defined by

$$\begin{aligned} (\mathcal{L}^x_{BS}u)^{n+\frac{1}{3}}_{ijk} &= \frac{(\sigma_x x_i)^2}{2}D_{xx}u^{n+\frac{1}{3}}_{ijk} + rx_iD_xu^{n+\frac{1}{3}}_{ijk} + \frac{1}{3}\sigma_x\sigma_y\rho_{xy}x_iy_jD_{xy}u^n_{ijk} \\ &+\frac{1}{3}\sigma_y\sigma_z\rho_{yz}y_jz_kD_{yz}u^n_{ijk} + \frac{1}{3}\sigma_z\sigma_x\rho_{zx}z_kx_iD_{zx}u^n_{ijk} - \frac{1}{3}ru^{n+\frac{1}{3}}_{ijk}, \\ (\mathcal{L}^y_{BS}u)^{n+\frac{2}{3}}_{ijk} &= \frac{(\sigma_y y_j)^2}{2}D_{yy}u^{n+\frac{2}{3}}_{ijk} + ry_jD_yu^{n+\frac{2}{3}}_{ijk} + \frac{1}{3}\sigma_x\sigma_y\rho_{xy}x_iy_jD_{xy}u^{n+\frac{1}{3}}_{ijk} \\ &+\frac{1}{3}\sigma_y\sigma_z\rho_{yz}y_jz_kD_{yz}u^{n+\frac{1}{3}}_{ijk} + \frac{1}{3}\sigma_z\sigma_x\rho_{zx}z_kx_iD_{zx}u^{n+\frac{1}{3}}_{ijk} - \frac{1}{3}ru^{n+\frac{2}{3}}_{ijk}, \end{aligned}$$

$$(\mathcal{L}^z_{BS}u)^{n+1}_{ijk}=\frac{(\sigma_z z_k)^2}{2}D_{zz}u^{n+1}_{ijk}+rz_kD_zu^{n+1}_{ijk}+\frac{1}{3}\sigma_x\sigma_y\rho_{xy}x_iy_jD_{xy}u^{n+\frac{2}{3}}_{ijk}$$
$$+\frac{1}{3}\sigma_y\sigma_z\rho_{yz}y_jz_kD_{yz}u^{n+\frac{2}{3}}_{ijk}+\frac{1}{3}\sigma_z\sigma_x\rho_{zx}z_kx_iD_{zx}u^{n+\frac{2}{3}}_{ijk}-\frac{1}{3}ru^{n+1}_{ijk}.$$

For the discretization of the spatial variables in Eq. (3), we employ the following difference equations:

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

Then, OSM consists of the following three discrete equations

$$\frac{u^{n+\frac{1}{3}}_{ijk}-u^n_{ijk}}{\Delta\tau}=\left(\mathcal{L}^x_{BS}u\right)^{n+\frac{1}{3}}_{ijk},\tag{4}$$

$$\frac{u^{n+\frac{2}{3}}_{ijk}-u^{n+\frac{1}{3}}_{ijk}}{\Delta\tau}=\left(\mathcal{L}^y_{BS}u\right)^{n+\frac{2}{3}}_{ijk},\tag{5}$$

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

Now, we describe the numerical algorithm for Eqs. (4)–(6). Given u^n_{ijk}, Eq. (4) is rewritten as follows:

$$\alpha_iu^{n+\frac{1}{3}}_{i-1,jk}+\beta_iu^{n+\frac{1}{3}}_{ijk}+\gamma_iu^{n+\frac{1}{3}}_{i+1,jk}=f_{ijk}\text{ for }i=1,\ldots,N_x,\tag{7}$$

where

$$\alpha_i=-\frac{(\sigma_xx_i)^2}{h^x_{i-1}(h^x_{i-1}+h^x_i)}+rx_i\frac{h^x_i}{h^x_{i-1}(h^x_{i-1}+h^x_i)},\tag{8}$$

$$\beta_i=\frac{(\sigma_xx_i)^2}{h^x_{i-1}h^x_i}-rx_i\frac{h^x_i-h^x_{i-1}}{h^x_{i-1}h^x_i}+\frac{1}{\Delta\tau},\tag{9}$$

$$\gamma_i=-\frac{(\sigma_xx_i)^2}{h^x_i(h^x_{i-1}+h^x_i)}-rx_i\frac{h^x_{i-1}}{h^x_i(h^x_{i-1}+h^x_i)},\tag{10}$$

$$f_{ijk}=\frac{1}{3}\sigma_x\sigma_y\rho_{xy}x_iy_jD_{xy}u^n_{ijk}+\frac{1}{3}\sigma_y\sigma_z\rho_{yz}y_jz_kD_{yz}u^n_{ijk}$$
$$+\frac{1}{3}\sigma_x\sigma_z\rho_{zx}x_iz_kD_{zx}u^n_{ijk}-\frac{1}{\Delta\tau}u^n_{ijk}.\tag{11}$$

For fixed index j and k, the solution vector $u^{n+\frac{1}{3}}_{1:N_x,jk}=[u^{n+\frac{1}{3}}_{1jk}\ u^{n+\frac{1}{3}}_{2jk}\ \cdots\ u^{n+\frac{1}{3}}_{N_xjk}]^T$ can be found by solving the tridiagonal system

$$A_xu^{n+\frac{1}{3}}_{1:N_x,jk}=f_{1:N_x,jk},$$

where A_x is a tridiagonal matrix constructed from Eq. (7) with the zero Dirichlet ($u^{n+\frac{1}{3}}_{0jk}=0$ at $x=0$) and linear boundary ($u^{n+\frac{1}{3}}_{N_x+1,jk}=2u^{n+\frac{1}{3}}_{N_xjk}-u^{n+\frac{1}{3}}_{N_x-1,jk}$ at $x=L$) conditions, i.e.,

$$A_x=\begin{pmatrix}\beta_1&\gamma_1&0&\cdots&0&0\\ \alpha_2&\beta_2&\gamma_2&\cdots&0&0\\ 0&\alpha_3&\beta_3&\cdots&0&0\\ \vdots&\vdots&\vdots&\ddots&\vdots&\vdots\\ 0&0&0&\cdots&\beta_{N_x-1}&\gamma_{N_x-1}\\ 0&0&0&\cdots&\alpha_{N_x}-\gamma_{N_x}&\beta_{N_x}+2\gamma_{N_x}\end{pmatrix}.$$

Algorithm 1 Numerical algorithm for Eq. (4).

Require: Previous data u^n.
procedure FIND THE SOLUTION $u^{n+\frac{1}{3}}$
  for $k=1;\ k\le N_z;\ k$++ **do**
    for $j=1;\ j\le N_y;\ j$++ **do**
      for $i=1;\ i\le N_x;\ i$++ **do**
        Set α_i, β_i, γ_i, and f_{ijk} by using Eqs. (8)–(11)
      end for
      Solve $A_xu^{n+\frac{1}{3}}_{1:N_x,jk}=f_{1:N_x,jk}$ by using the Thomas algorithm
    end for
  end for
end procedure

Then, Eq. (4) is implemented in a loop over the y- and z- directions with the following Algorithm 1.

Similarly, Eqs. (5) and (6) are solved. Note that one execution from Eqs. (4)–(6) advances the numerical solution with a $\Delta\tau$-step in time. For more details about the solution algorithm, see references (Jeong & Kim, 2013; Jeong, Wee, & Kim, 2010).

4. Numerical experiments

In order to demonstrate the efficiency and accuracy of the proposed method, we perform the numerical experiments such as convergence test, domain size effect, non-uniform grid size effect, comparison with MCS, computation of the Greeks, and pricing three-asset step-down ELS. In all numerical tests, unless otherwise stated, we use the same parameters $r=0.03$, $\sigma_x=\sigma_y=\sigma_z=0.3$, and $\rho_{xy}=\rho_{yz}=\rho_{zx}=0.5$. OSM and MCS algorithms are implemented using MATLAB (Mathworks Inc., Natick, MA, USA) and closed-form solutions are obtained using Mathematcia (Wolfram, 1999). Especially, in MCS, we use the antithetic variates technique (Hammersley & Morton, 1956) among the variance reduction methods. The programs are executed on Intel(R) Core(TM) Duo CPU @3.00 GHz desktop PC.

4.1. Convergence test

We perform a convergence test in order to verify the accuracy of OSM. We consider the cash-or-nothing option with three underlying assets. The payoff is given as

$$u(x,y,z,0)=\begin{cases}100 & \text{if } x\ge K_1, y\ge K_2, z\ge K_3,\\ 0 & \text{otherwise.}\end{cases}$$

Here, we take $K_1=K_2=K_3=100$. For this test, we use $\Omega=[0,200]\times[0,200]\times[0,200]$ and $T=1/12$. We discretize each direction of the domain as $[0, 100-(m+0.5)h,\ldots,100-1.5h, 100-0.5h, 100+0.5h, 100+1.5h,\ldots,100+(m+0.5)h, 200]$, where $m=\text{round}[100/h-0.5]-1$. The order of accuracy is defined as the ratio of successive errors: $\log_2(e_h/e_{h/2})$, where e_h is the difference of the numerical solution using the space step size h and the exact solution u_e. Exact solution for the cash-or-nothing option is obtained from the formula which is described in Appendix.

First, we compute the numerical solutions $u(100,100,100,T)$ with $\Delta\tau=1/1440$ and a set of increasingly finer grids, i.e., $h=8,4,2$, to test the convergence rate for the spatial discretization. To get the option value by OSM at a point $(x,y,z)=(100,100,100)$, we use trilinear interpolation from the eight neighborhood points. The exact solution is calculated as $u_e=24.41647$ at $(x,y,z)=(100,100,100)$. Table 1 shows that this method has the second order accuracy in space.

To show the convergence rate for the temporal discretization, we fix the space step size as $h=1$ and choose a set of decreasing time steps, $\Delta\tau=1/90, 1/180, 1/360$. All other parameters are the same as

Table 1
Option values, errors, and convergence orders of OSM at $x = y = z = 100$ and $T = 1/12$ on various mesh grids with $\Delta\tau = 1/1440$.

h	8		4		2
Value	20.63802		23.50780		24.24836
Error	3.77844		0.90867		0.16810
Order		2.05596		2.43444	

Table 2
Option values, errors, and convergence orders of OSM at $x = y = z = 100$ and $T = 1/12$ with various time steps and $h = 1$.

$\Delta\tau$	1/90		1/180		1/360
Value	26.12745		25.17284		24.73388
Error	1.71098		0.75637		0.31741
Order		1.17766		1.25274	

Table 3
Values of $u(100, 100, 100, T)$ on various computational domain $\Omega = [0, L]^3$.

L	125	150	175	200	225	250
Values	27.55637	23.43614	23.42480	23.42479	23.42479	23.42479

Table 4
Values of $u(100, 100, 100, T)$ and computational time on various non-uniform grids.

$h = 1$	Value (CPU time)	$h = 2$	Value (CPU time)
0.5: 1: 199.5	24.73388 (286.95)	1: 2: 199	24.51486 (42.95)
69.5: 1: 130.5	24.73388 (11.86)	69: 2: 131	24.51486 (2.33)
79.5: 1: 120.5	24.73497 (5.01)	79: 2: 121	24.51577 (1.05)
89.5: 1: 110.5	24.82122 (1.08)	89: 2: 111	24.56993 (0.34)
91.5: 1: 108.5	24.57615 (0.74)	91: 2: 109	24.35032 (0.26)
95.5: 1: 104.5	21.46677 (0.27)	95: 2: 105	21.48070 (0.15)
99.5: 1: 100.5	9.76569 (0.17)	99: 2: 101	10.55660 (0.09)

before. As expected from the discretization, first-order accuracy with respect to time is observed in Table 2.

4.2. Domain size effect

We investigate the effect of domain size $\Omega = [0, L]^3$ for the numerical solution $u(100, 100, 100, T)$. We take $L = 125, 150, 175, 200, 225$, and 250 with $h = 1$, $T = 1/2$, and $\Delta\tau = 1/360$. For each x-, y-, and z-directions, we set the mesh as $[0, 0.5, 1.5, \ldots, L - 1.5, L - 0.5, L]$. From Table 3, we can observe that if L is larger than 200, the values do not change significantly. Therefore, $L = 200$ is large enough to be used for calculating option values when the strike price is 100.

4.3. Non-uniform grid effect

In this test, we find an optimal non-uniform grid structure which gives the smallest number of grid points with equivalent accuracy of uniform mesh. Table 4 shows the value of $u(100, 100, 100, T)$ calculated by using non-uniform grids on $\Omega = [0, 200]^3$ with $\Delta\tau = 1/360$ and $T = 1/12$. In Table 4, a: h: b denotes that mesh is $[0,\ a,\ a + h,\ a + 2h, \ldots,\ b - h,\ b,\ (b + 200)/2,\ 200]$ for all three x-, y-, and z-directions. As shown in Table 4, the values at the top five lines are almost the same. This result suggests that we can use smaller number of grid points with equivalent accuracy. Next to the values, we also put CPU times. By using non-uniform grid, we can substantially reduce the calculation time.

4.4. Comparison of MCS and OSM

We compare the results from MCS and OSM with equivalent computational costs. MCS is tested with a temporal step size $\Delta\tau = 1/360$ and 10^4, 10^5, and 10^6 samples until $T = 1/12$. In OSM simulations, non-uniform meshes [0 69.5:1:130.5 165.25 200], [0 74.5:1:125.5 162.75 200], and [0 79.5:1:120.5 160.25 200] for each direction are used with $\Delta\tau = 1/1440$ until $T = 1/12$. In Fig. 3, the open circles are the results of $u(100, 100, 100, T)$ from MCS with varying number of samples. For each number of samples, we plot the results obtained from 10 trials. In the legend in Fig. 3, 69.5: 130.5 denotes the numerical solution (solid symbol) from OSM on a non-uniform grid [0 69.5: 1: 130.5 165.25 200]. The other notations represent non-uniform grids in the same manner. The solid line is the analytic solution. This simulation result indicates that OSM converges to the analytic solution faster than MCS does with the same computational cost.

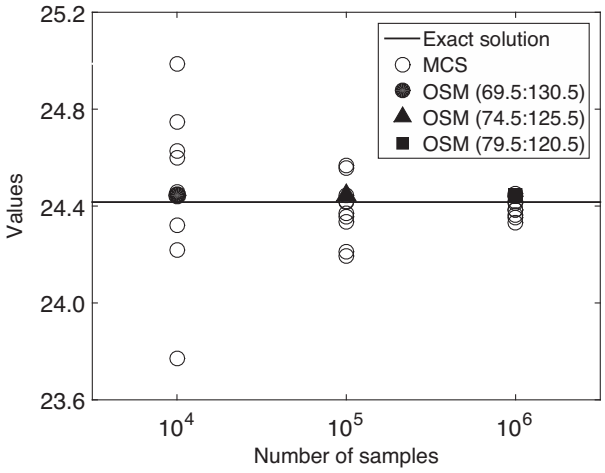

Fig. 3. Results from MCS (open circles), OSM (solid symbols), and analytic solution (solid line) at $\mathbf{x} = (100, 100, 100)$.

4.5. Greeks

In this section, the temporal step sizes $\Delta\tau = 1/720$, $\Delta\tau = 1/360$ are used for OSM and MCS, respectively. We get the values of the Greeks at a point $\mathbf{x} = (100, 100, 100)$ and compare the two numerical results calculated by MCS and OSM.

4.5.1. Delta

Delta (Δ) is the rate of change of the option price with respect to the price of the underlying asset. We calculate the Delta Δ_x at $\mathbf{x} = (100, 100, 100)$, which is defined as

$$\Delta_x(\mathbf{x}, T) := u_x(\mathbf{x}, T) \approx \frac{u(\mathbf{x} + 0.5h\mathbf{e}, T) - u(\mathbf{x} - 0.5h\mathbf{e}, T)}{h}, \quad (12)$$

where $\mathbf{e}$ denotes (1, 0, 0). Therefore, we need two simulations to obtain two values $u(\mathbf{x} + 0.5h\mathbf{e}, T)$ and $u(\mathbf{x} - 0.5h\mathbf{e}, T)$ when we use MCS. However, in OSM, we can get the Delta from only one simulation. Fig. 4 shows the results by a closed-form solution, MCS, and OSM. Here, the closed-form solution of the Delta is derived by the formula in Appendix and its value is $\Delta_x(\mathbf{x}, T) = 1.38192$. In MCS, the values of the Delta converge as the number of samples increases. With OSM, we obtain the Delta on three different non-uniform grids. In the legend in Fig. 4, 69.5: 130.5 represents a non-uniform grid [0 69.5: 1: 130.5 165.25 200]. The other legends represent non-uniform grids in the same manner. We need 4.15 seconds for one MCS trial with 10^5 samples. This means that 10 MCS trials with 10^6 samples takes 415 seconds. However, the numerical solution by OSM on [0 69.5: 1: 130.5 165.25 200] is calculated in 24.99 seconds. As shown in Fig. 4, we can confirm that OSM is more stable and fast method than MCS.

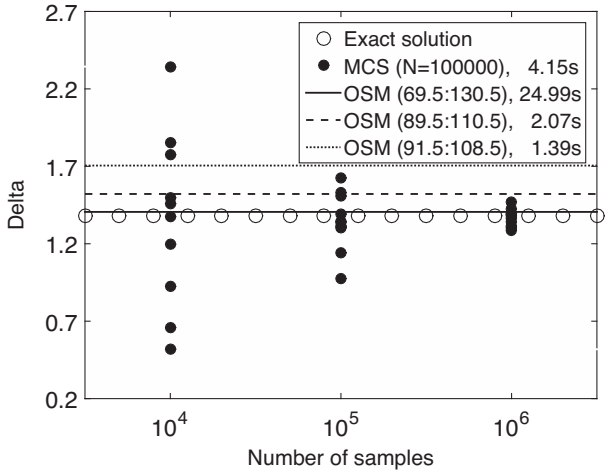

Fig. 4. Delta obtained from closed-form solution (open circles), MCS (solid circles), and OSM (three different lines) at $\mathbf{x} = (100, 100, 100)$.

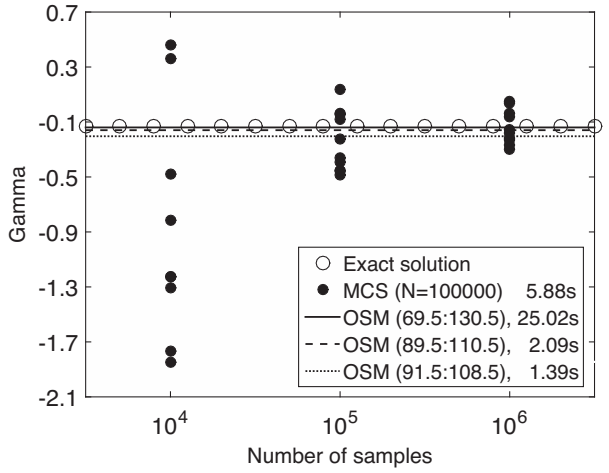

Fig. 5. Gamma obtained from closed-form solution (open circles), MCS (solid circles), and OSM (three different lines) at $\mathbf{x} = (100, 100, 100)$.

4.5.2. Gamma

In this section, we consider the Gamma (Γ). Gamma represents the change in the Delta of an option relative to the change in the underlying assets. Therefore, in hedging, traders need to measure Gamma as large moves in an underlying asset could change the risks they hold in option positions dramatically.

For comparison, we calculate the Gamma Γ_x at a point $\mathbf{x} = (100, 100, 100)$. Since Γ_x is the second derivative of u with respect to x, we can express the Gamma Γ_x as follows:

$$\Gamma_x(\mathbf{x}, T) := u_{xx}(\mathbf{x}, T) \approx \frac{u(\mathbf{x} - h\mathbf{e}, T) - 2u(\mathbf{x}, T) + u(\mathbf{x} + h\mathbf{e}, T)}{h^2}. \tag{13}$$

In order to evaluate the Gamma Γ_x with MCS by the formula (13), we need three different simulations. However, we can get the Gamma from OSM through only one simulation. Therefore, we can save the computational time by using OSM on a non-uniform grid. Fig. 5 represents the Gamma Γ_x from a closed-form solution, MCS, and OSM. The Gamma from a closed-form solution is $\Gamma_x(\mathbf{x}, T) = -0.133136$. In MCS, the Gamma with 10^5 samples is obtained in about 5.88 seconds. To get a reliable result with MCS, we have to perform the simulation with more than 10^6 samples. It means that the computational time takes more than 588 seconds. However, with OSM, we can obtain a comparable value of the Gamma to closed-form solution within only 25.02 seconds. This shows that OSM is more efficient than MCS for calculating the Gamma.

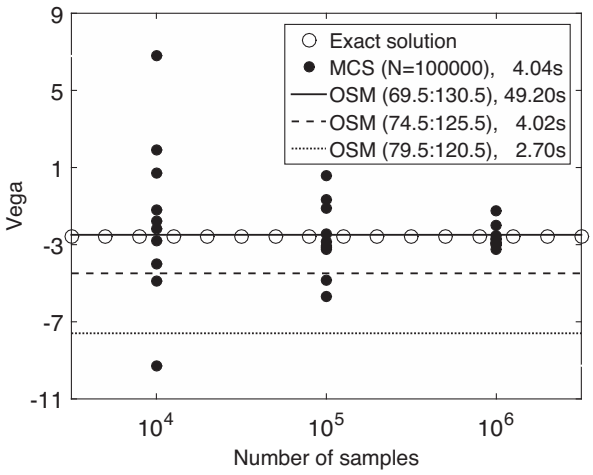

Fig. 6. Vega obtained from closed-form solution (open circles), MCS (solid circles), and OSM (three different lines) at $\mathbf{x} = (100, 100, 100)$ and $\sigma_x = 0.35$.

4.5.3. Vega

Vega is the option's sensitivity to changes in the volatility. This value is a number that tells in direction and extent the option price will move if there is a positive change in the volatility.

In this section, we compare the Vega from MCS and OSM. Since the Vega is the derivative of the option price u with respect to the volatility σ_x, we can calculate the Vega at a point $\mathbf{x} = (100, 100, 100)$ and $\sigma_x = 0.35$ with $\Delta\sigma_x = 0.1$, i.e.,

$$\begin{aligned}\text{Vega}(\mathbf{x}, T)|_{\sigma_x=0.35} &:= u_{\sigma_x}(\mathbf{x}, T)|_{\sigma_x=0.35} \\ &\approx \frac{u(\mathbf{x}, T)|_{\sigma_x=0.4} - u(\mathbf{x}, T)|_{\sigma_x=0.3}}{0.1}.\end{aligned} \tag{14}$$

Fig. 6 shows the numerical results of the Vega from a closed-form solution, MCS, and OSM. Here, the Vega from closed-form formula is -2.59518. As shown in Fig. 6, the numerical result of OSM is more closer to exact solution than MCS. Also, the values of Vega by OSM is stable and deterministic while those of MCS slowly converge.

4.5.4. Rho

As the sensitivity of the option value to interest rate, Rho is defined the first derivative of the option value with respect to the interest rate r. In this section, we evaluate the Rho at a point $\mathbf{x} = (100, 100, 100)$ and $r = 0.03$ as follows.

$$\begin{aligned}\text{Rho}(\mathbf{x}, T)|_{r=0.03} &:= u_r(\mathbf{x}, T)|_{r=0.03} \\ &\approx \frac{u(\mathbf{x}, T)|_{r=0.03+0.00015} - u(\mathbf{x}, T)|_{r=0.03-0.00015}}{0.0003}.\end{aligned} \tag{15}$$

Here, the Rho means the rate of change in the option value per 1% change in the risk-free interest rate $r = 0.03$. Fig. 7 represents the Rho from a closed-form solution, MCS, and OSM.

In Fig. 7, the numerical results by OSM are closer to the closed-form solution. However, the results by MCS have a big deviation from the exact solution. Therefore, OSM is more efficient to evaluate the Rho than MCS.

4.5.5. Theta

Theta (θ) is the sensitivity of an option's value to change in time. This indicates an absolute change in option value for one-unit reduction in time. We can calculate this value though the following formula:

$$\theta(\mathbf{x}, T) := u_t(\mathbf{x}, T) \approx \frac{u(\mathbf{x}, T) - u(\mathbf{x}, T - \Delta t)}{\Delta t}. \tag{16}$$

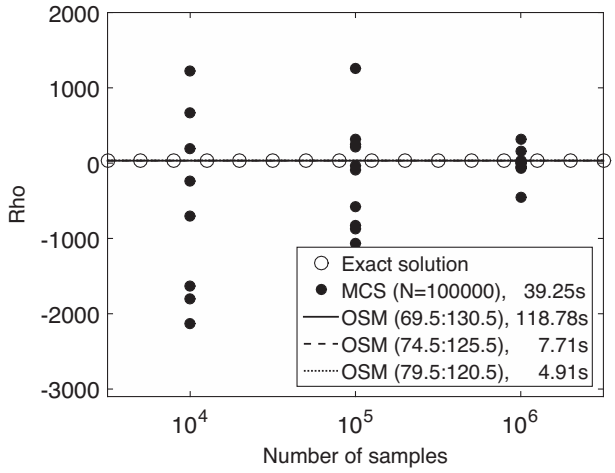

Fig. 7. Rho obtained from closed-form solution (open circles), MCS (solid circles), and OSM (three different lines) at $\mathbf{x} = (100, 100, 100)$ and $r = 0.03$.

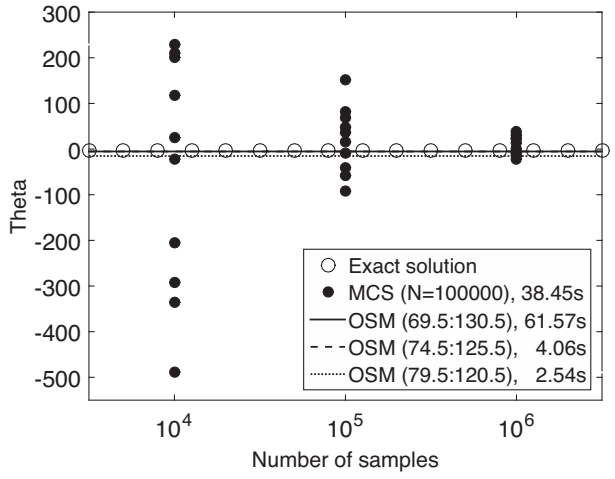

Fig. 8. Theta obtained from closed-form solution (open circles), MCS (solid circles), and OSM (three different lines) at $\mathbf{x} = (100, 100, 100)$ and $T = 1/12$.

By Eq. (16), we obtain the value of θ from closed-form solution, MCS, and OSM. These results are shown in Fig. 8. One of the notable points is that the value of θ from OSM is more quickly evaluated than MCS. The θ can be calculated with two option values $u(\mathbf{x}, T - \Delta t)$ and $u(\mathbf{x}, T)$. Therefore, MCS needs two calculations unlike OSM which needs only one calculation.

From these simulation results for computing the values of the Greeks, we can confirm that the numerical results from OSM converge to the analytic solutions faster than MCS with the same computational cost.

4.6. Three-asset step-down ELS

As a practical example, let us consider a three-asset step-down ELS which is the most popular kind of ELS structure in Korean financial market. The three-asset step-down ELS is a derivative securities whose payoff is determined by three underlying equity prices. This option is automatically exercised at each validation date before or at maturity if the price of underlying asset is above a pre-defined exercise level. The validation date is called the early redemption date. In step-down ELS, the pre-specified exercise levels are lowered from previous validation date to next date (Lee, 2013). If the price of underlying asset is below those levels on each validation date, the payoff will be deferred maturity (Kim, Bae, & Koo, 2014).

Fig. 9(a) and (b) show the pay-off function at maturity and early redemption date for the step-down ELS, respectively. Here, the early redemption value denotes the pre-specified interim evolution date which has early redemption chances (Kim et al., 2014).

Let $x(t)$, $y(t)$, $z(t)$ be linearly scaled underlying assets' prices at time t with $(x(0), y(0), z(0)) = (100, 100, 100)$. Let T be maturity, F be face value, and t_i for $i = 1, \ldots, N$ be early obligatory redemption observation dates. K_i and c_i are corresponding exercise prices and rates of return, respectively. Let D denote the knock-in barrier level and d denote a dummy. Let $S_t = \min\{x(t), y(t), z(t)\}$ be the minimum of three underlying assets at t. By definition of τ, we have $S_\tau = \min\{x(T-\tau), y(T-\tau), z(T-\tau)\}$. Let $u(x, y, z, \tau)$ and $v(x, y, z, \tau)$ be the solutions with and without knock-in event, respectively. A knock-in event happens when $\min\{S_t | 0 \le t \le T\} < D$. The initial conditions are given as

$$u(x, y, z, \tau = 0) = \begin{cases} FS_T/100 & \text{if } S_T < K_6, \\ (1 + c_6)F & \text{otherwise.} \end{cases} \tag{17}$$

$$v(x, y, z, \tau = 0) = \begin{cases} (1 + c_6)F & \text{if } S_T \ge K_6, \\ (1 + d)F & \text{if } D < S_T < K_6, \\ FS_T/100 & \text{otherwise.} \end{cases} \tag{18}$$

and Fig. 9(b) shows the diagonal plots of the early obligatory redemption conditions along the line $x = y = z$.

The solution algorithm is as follows: First, we update u and v by solving Eqs. (4)–(6) with the initial conditions (17) and (18). After that we replace the values of v by u in the region which is bounded by the knock-in barrier D, i.e., $v^1_{ijk} = u^1_{ijk}$ for $(x_i, y_j, z_k) \in \Omega_{ki} = \{(x, y, z) | x < D, y < D, z < D\}$. We apply this replacement after every time step until $\tau = T$, i.e., $v^n_{ijk} = u^n_{ijk}$ for $(x_i, y_j, z_k) \in \Omega_{ki}$ and $n = 1, \ldots, N_t$. Let $\Omega_m = \{(x, y, z) | x \ge K_m, y \ge K_m, z \ge K_m\}$. At the first observation date

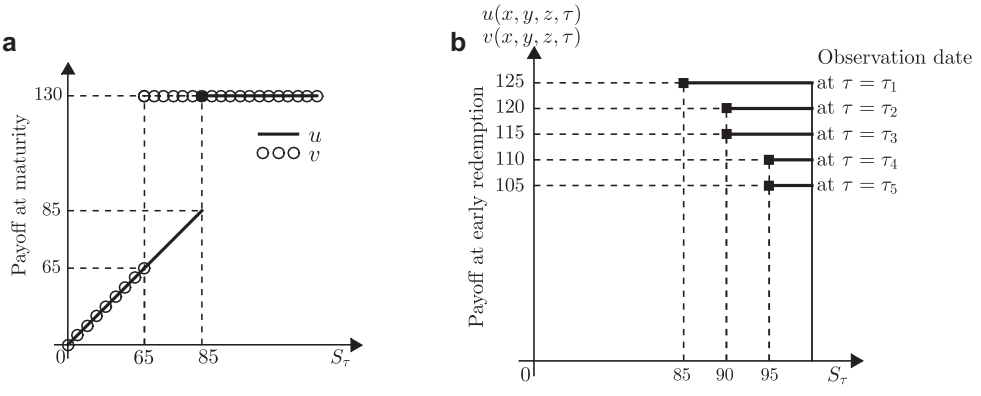

Fig. 9. Pay-off functions at (a) maturity and (b) early redemption for three-asset step-down ELS.

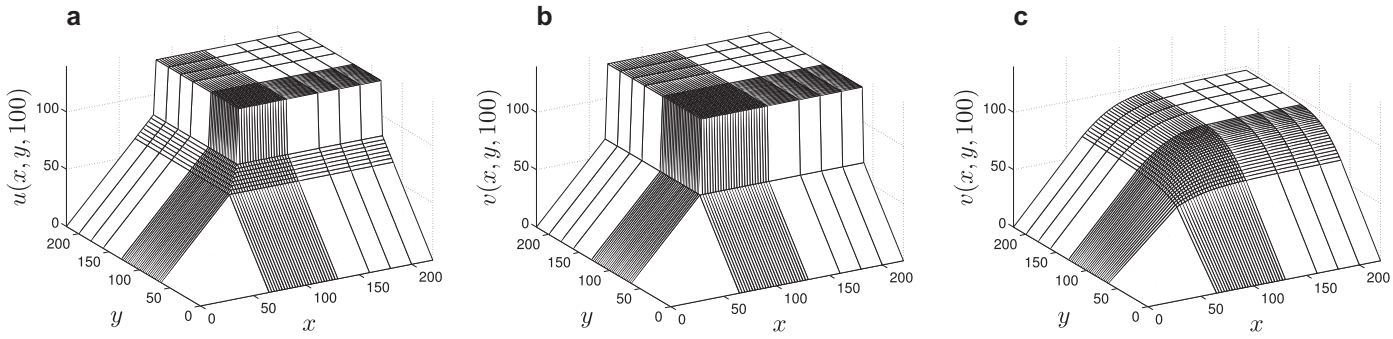

Fig. 10. (a) and (b) are payoff functions of u and v at $z = 100$, respectively. (c) is the final solution of v at $z = 100$ and $\tau = 3$.

Table 5
Parameters of three-asset step-down ELS.

Observation date (years)	Exercise price	Return rate
$\tau_1 = T/6$	$K_1 = 0.85K_0$	$c_1 = 0.25$
$\tau_2 = 2T/6$	$K_2 = 0.90K_0$	$c_2 = 0.20$
$\tau_3 = 3T/6$	$K_3 = 0.90K_0$	$c_3 = 0.15$
$\tau_4 = 4T/6$	$K_4 = 0.95K_0$	$c_4 = 0.10$
$\tau_5 = 5T/6$	$K_5 = 0.95K_0$	$c_5 = 0.05$

$\tau_1 = T/6$, we reset values of u and v as $u_{ijk}^{n_1} = v_{ijk}^{n_1} = (1 + c_1)F$ for $(x_i, y_j, z_k) \in \Omega_1$. Likewise, at the intermediate observation dates τ_n, we reset values of u and v as $u_{ijk}^{n_m} = v_{ijk}^{n_m} = (1 + c_m)F$ for $(x_i, y_j, z_k) \in \Omega_m$ for $m = 1, 2, 3, 4$, and 5. The parameters are listed in Table 5.

In this section, we chose the following parameters: the reference price $K_0 = 100$, the interest rate $r = 0.03$, the volatilities of the underlying assets $\sigma_x = \sigma_y = \sigma_z = 0.3$, the total time $T = 3$ years, the face price $F = 100$, the knock-in barrier level $D = 0.65K_0$, the dummy rate $d = 0.3$, the correlation $\rho_{xy} = \rho_{yz} = \rho_{zx} = 0.5$, and the computational domain $\Omega = [0, 200] \times [0, 200] \times [0, 200]$. We compare the results from MCS and OSM for the step-down ELS option (see Fig. 9). As a reference solution value, we run MCS with 10^7 samples and $\Delta\tau = 1/360$. The option value obtained from this MCS is 84.4431 and it takes 5035.73 seconds. For OSM simulation, $\Delta\tau = 1/30$ and a non-uniform mesh [0 60:2.5:130 160 180 200 220] for each direction are used. Fig. 10(a and b) show payoff functions of u and v at $z = 100$, respectively. Fig. 10(c) is the final solution of v at $z = 100$ and $\tau = 3$. We get its numerical value as 84.6347 and it takes only 12.0938 seconds. The relative percent error is $(84.4431 - 84.6347)/84.4431 \times 100\,\% = -0.2269\,\%$. This percent error demonstrates the efficiency and accuracy of the non-uniform OSM.

5. Conclusions

To price the three-asset equity-linked securities (ELS) options, a fast and accurate numerical method is developed. In this paper, we use the Black–Scholes partial differential equation for the option pricing. The model is discretized by using a non-uniform finite difference method and the resulting discrete equations are solved by means of the operator splitting method. In particular, in order to do fast and accurate calculation, we use non-uniform grid structures which put more grid points near the singularity of the nonsmooth payoff function. The numerical results show that the cost of the proposed method is comparable to that of Monte Carlo simulation and it provides more stable hedging parameters such as the Greeks.

Acknowledgments

The first author (J.S. Kim) was supported by a subproject of project Research for Applications of Mathematical Principles (No C21501) and supported by the National Institute of Mathematics Sciences (NIMS). The corresponding author (D. Jeong) was supported by a Korea University Grant. The authors are grateful to the anonymous referees whose valuable suggestions and comments significantly improved the quality of this paper.

Appendix.

The closed-form solution $u(x_1, x_2, x_3, t)$ of Eq. (1) is given by

$$u(x_1, x_2, x_3, t) = \frac{Ee^{-r(T-t)}}{\sigma_1\sigma_2\sigma_3\sqrt{(2\pi\,(T-t))^3\det\Sigma}} \times \int_{K_3}^{\infty}\int_{K_2}^{\infty}\int_{K_1}^{\infty} \frac{\exp\left(-0.5\alpha\Sigma^{-1}\alpha^T\right)}{\xi_1\xi_2\xi_3} d\xi_1 d\xi_2 d\xi_3,$$

where the payoff function is

$$u(x_1, x_2, x_3, 0) = \begin{cases} E & \text{if } x_1 \geq K_1, x_2 \geq K_2, x_3 \geq K_3, \\ 0 & \text{otherwise,} \end{cases}$$

and the correlation matrix Σ is

$$\Sigma = \begin{pmatrix} 1 & \rho_{12} & \rho_{13} \\ \rho_{21} & 1 & \rho_{23} \\ \rho_{31} & \rho_{32} & 1 \end{pmatrix},$$

which is symmetric for the correlation coefficients ρ_{ij} and $\alpha = (\alpha_1, \alpha_2, \alpha_3)$ where α_i is defined by $\alpha_i = \left[\ln(x_i/\xi_i) + \left(r - 0.5\sigma_i^2\right)(T-t)\right]/(\sigma_i\sqrt{T-t})$. We evaluate the closed-form option price and the Greeks by using *Mathematica* (Wolfram, 1999).

References

Bandi, C., & Bertsimas, D. (2014). Robust option pricing. *European Journal of Operational Research, 239*(3), 842–853.
Black, F., & Scholes, M. (1973). The pricing of options and corporate liabilities. *The Journal of Political Economy, 81*(3), 637–654.
Capiński, M. J. (2015). Hedging conditional value at risk with options. *European Journal of Operational Research, 242*(2), 688–691.
Choi, Y., & Song, J. (2008). An improved approach for valuing American options and their greeks by least-squares Monte Carlo simulation. *Asia–Pacific Journal of Finance Studies, 37*(2), 217–244.
Duffy, D. J. (2006). *Finite difference methods in financial engineering: a partial differential equation approach*. New York: John Wiley and Sons.
González-Parra, G., Arenas, A. J., & Chen-Charpentier, B. M. (2013). Positive numerical solution for a nonarbitrage liquidity model using nonstandard finite difference schemes. *Numerical Methods for Partial Differential Equations, 30*(1), 210–221.
Hammersley, J. M., & Morton, K. W. (1956). A new monte carlo technique: antithetic variates. In *Mathematical proceedings of the cambridge philosophical society, vol. 52, no. 03* (pp. 449–475). Cambridge University Press. July
Han, H., & Wu, X. (2003). A fast numerical method for the Black–Scholes equation of American options. *SIAM Journal on Numerical Analysis, 41*(6), 2081–2095.
Haug, E. G. (2007). *The complete guide to option pricing formulas*. McGraw-Hill Companies.
Jeong, D., & Kim, J. (2013). A comparison study of ADI and operator splitting methods on option pricing models. *Journal of Computational and Applied Mathematics, 247*, 162–171.

Jeong, D., Kim, J., & Wee, I. S. (2009). An accurate and efficient numerical method for Black–Scholes equations. *Communications of the Korean Mathematical Society, 24*(4), 617–628.

Jeong, D., Wee, I. S., & Kim, J. (2010). An operator splitting method for pricing the ELS option. *Journal of the Korean Society for Industrial and Applied Mathematics, 14*(3), 175–187.

Kim, Y., Bae, H. O., & Koo, H. K. (2014). Option pricing and Greeks via a moving least square meshfree method. *Quantitative Finance, 14*(10), 1753–1764.

Lee, Y. (2013). *Heston–Hull–White Model for Long Term Exotic Option*. KAIST Master's thesis.

Marroquı, N., & Moreno, M. (2013). Optimizing bounds on security prices in incomplete markets. does stochastic volatility specification matter? *European Journal of Operational Research, 225*(3), 429–442.

Persson, J., & von Persson, L. (2007). Pricing European multi-asset options using a space-time adaptive FD-method. *Computing and Visualization in Science, 10*(4), 173–183.

Pun, C. S., Chung, S. F., & Wong, H. Y. (2015). Variance swap with mean reversion, multifactor stochastic colatility and jumps. *European Journal of Operational Research, 245*(2), 571–580.

Rambeerich, N., Tangman, D. Y., Lollchund, M. R., & Bhuruth, M. (2013). High-order computational methods for option valuation under multifactor models. *European Journal of Operational Research, 224*(1), 219–226.

Reisinger, C., & Wittum, G. (2004). On multigrid for anisotropic equations and variational inequalities. *Computing and Visualization in Science, 7*(3–4), 189–197.

Wilmott, P., Dewynne, J., & Howison, S. (1993). *Option pricing : mathematical models and computation*. Oxford: Oxford Financial Press.

Wolfram, S. (1999). *The MATHEMATICA® Book Version 4*. Cambridge: Cambridge University Press.

Zvan, R., Vetzal, K. R., & Forsyth, P. A. (2000). PDE methods for pricing barrier options. *Journal of Economic Dynamics and Control, 24*(11), 1563–1590.