

Research Article

Accuracy, Robustness, and Efficiency of the Linear Boundary Condition for the Black-Scholes Equations

Darae Jeong,¹ Seungsook Seo,² Hyeongseok Hwang,³ Dongsun Lee,¹
Yongho Choi,¹ and Junseok Kim¹

¹Department of Mathematics, Korea University, Seoul 136-713, Republic of Korea

²Garam Analytics, Seodaemun, Seoul 120-749, Republic of Korea

³Department of Financial Engineering, Korea University, Seoul 136-701, Republic of Korea

Correspondence should be addressed to Junseok Kim; cfdkim@korea.ac.kr

Received 20 August 2014; Accepted 22 December 2014

Academic Editor: Xiao-Qiao He

Copyright © Darae Jeong et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

We briefly review and investigate the performance of various boundary conditions such as Dirichlet, Neumann, linear, and partial differential equation boundary conditions for the numerical solutions of the Black-Scholes partial differential equation. We use a finite difference method to numerically solve the equation. To show the efficiency of the given boundary condition, several numerical examples are presented. In numerical test, we investigate the effect of the domain sizes and compare the effect of various boundary conditions with pointwise error and root mean square error. Numerical results show that linear boundary condition is accurate and efficient among the other boundary conditions.

1. Introduction

We briefly review and perform a comparison study for Dirichlet, Neumann, linear, and partial differential equation (PDE) boundary conditions (BCs) for the Black-Scholes (BS) partial differential equations. Let S and t denote the price of the underlying asset and time, respectively. Then, the value of the option $u(S, t)$ is governed by the BS equation [1]:

$$\frac{\partial u(S, t)}{\partial t} + \frac{1}{2} (\sigma S)^2 \frac{\partial^2 u(S, t)}{\partial S^2} + rS \frac{\partial u(S, t)}{\partial S} = ru(S, t), \quad (1)$$

for $(S, t) \in \mathbf{R}^+ \times [0, T)$,

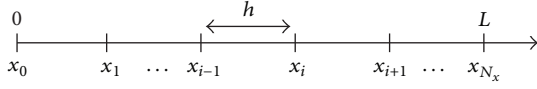
where σ is a constant volatility of the asset and $r > 0$ is a constant riskless interest rate. The final condition is the payoff function $\Lambda(S)$ at expiry T :

$$u(S, T) = \Lambda(S). \quad (2)$$

It is difficult to find the analytic solutions of (1) and (2) for exotic options. Therefore we need to use a numerical

approximation. To obtain an approximation of the option value, one can compute a solution of the BS equations (1) and (2) using a finite difference method (FDM) [2–8], finite element method [9–11], finite volume method [12–14], a fast Fourier transform [15–17], and also their optimal BC [18]. Here, we consider FDM approach to solve the BS PDE. See [1, 19–24] for more details for FDM applied to option pricing. To solve the BS PDE using the FDM, we need to truncate the infinite domain to a finite domain and have to use an artificial BC. To reduce large errors in the numerical solution due to this approximation of the BCs, the truncated domain must be large enough. Thus the purpose of this paper is to investigate the effects of several BCs on the numerical solutions for the BS PDE (1) and (2).

The outline of the paper is as follows. In Section 2, we formulate the generalized version of the BS PDE. And we explain five cases which are Dirichlet I, Dirichlet II, Neumann, linear, and PDE BCs. Section 3 presents the results of the numerical experiments and the conclusions are drawn in Section 4.

FIGURE 1: Uniform grid with a spatial step size h .

2. Numerical Solution

Let $x = S$ be the value of the underlying asset price and let $\tau = T - t$ be the time to expiry; then (1) becomes a more natural initial value problem

$$\frac{\partial u(x, \tau)}{\partial \tau} = \frac{1}{2}(\sigma x)^2 \frac{\partial^2 u(x, \tau)}{\partial x^2} + rx \frac{\partial u(x, \tau)}{\partial x} - ru(x, \tau),$$

for $(x, \tau) \in \Omega \times (0, T]$

(3)

with an initial condition $u(x, 0) = \Lambda(x)$ for $x \in \Omega = (0, L)$. Here we truncate the infinite domain into a finite domain since the infinite domain cannot be represented in the computer [23].

2.1. Discretization with Finite Difference Method. We apply the FDM for solving (3) numerically. Let us first discretize the given computational domain $\Omega = [0, L]$ as a uniform grid with a spatial step size $h = L/N_x$ and a temporal step size $\Delta\tau = T/N_t$, where N_x is the number of subintervals (see Figure 1) and N_t is the number of time steps.

Let us denote the numerical approximation of the solution as

$$u_i^n \equiv u(x_i, \tau_n) = u(ih, n\Delta\tau), \quad (4)$$

where $i = 0, 1, \dots, N_x$ and $n = 0, 1, \dots, N_t$. By applying the fully implicit-in-time and space-centered difference scheme to (3), we have

$$\frac{u_i^{n+1} - u_i^n}{\Delta\tau} = \frac{\sigma^2 x_i^2}{2} \frac{u_{i-1}^{n+1} - 2u_i^{n+1} + u_{i+1}^{n+1}}{h^2} + rx_i \frac{u_{i+1}^{n+1} - u_{i-1}^{n+1}}{2h} - ru_i^{n+1}. \quad (5)$$

We can rewrite (5) by

$$\alpha_i u_{i-1}^{n+1} + \beta_i u_i^{n+1} + \gamma_i u_{i+1}^{n+1} = b_i, \quad (6)$$

where $\alpha_i = rx_i/2h - \sigma^2 x_i^2/2h^2$, $\beta_i = 1/\Delta\tau + \sigma^2 x_i^2/h^2 + r$, $\gamma_i = -rx_i/2h - \sigma^2 x_i^2/2h^2$, and $b_i = u_i^n/\Delta\tau$.

In order to solve the linear system (6), we need to know u_0^n and $u_{N_x}^n$ for all $n = 0, \dots, N_t$. At $x = 0$, we simply set $u_0^n = 0$. Next, we present five different boundary conditions for specifying the values of $u_{N_x}^n$.

2.2. Boundary Conditions. In solving the BS equation numerically, there are many BCs. In this section, we introduce five BCs, which are used to solve the BS equation. We focus on two

options: European call option and cash-or-nothing option. The payoff functions are given by

$$u(x, 0) = \max(x - K, 0), \quad (7)$$

$$u(x, 0) = \begin{cases} C & \text{if } x > K \\ 0 & \text{otherwise,} \end{cases} \quad (8)$$

for the European call option and the cash-or-nothing option, respectively. Here K is the strike price and C denotes the return value at expiration if the option is in-the-money. We have closed-form solutions for these options. For the European call option, the closed-form solution of the BS equation is

$$u(x, \tau) = xN(d_1) - Ke^{-r\tau}N(d_2),$$

$\forall x \in [0, L], \quad \forall \tau \in [0, T],$

(9)

$$d_1 = \frac{\ln(x/K) + (r + (1/2)\sigma^2)\tau}{\sigma\sqrt{\tau}}, \quad d_2 = d_1 - \sigma\sqrt{\tau}, \quad (10)$$

where $N(d) = (1/\sqrt{2\pi}) \int_{-\infty}^d \exp(-x^2/2)dx$ is the cumulative distribution function for the standard normal distribution [25]. For the cash-or-nothing option, the closed form solution is

$$u(x, \tau) = Ce^{-r\tau}N(d_2). \quad (11)$$

2.2.1. Dirichlet I Boundary Condition. From (9), we can observe that $N(d_1)$ and $N(d_2)$ are close to one when x is large enough. Therefore, Dirichlet BC for the European call option is defined by

$$u(L, \tau) = L - Ke^{-r\tau} \quad (12)$$

for a sufficiently large L . By (12), we use $u_{N_x}^{n+1} = x_{N_x} - Ke^{-r(n+1)\Delta\tau}$. Using this, we can rewrite (6) as

$$\begin{pmatrix} \beta_1 & \gamma_1 & 0 & \cdots & 0 \\ \alpha_2 & \beta_2 & \gamma_2 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & \alpha_{N_x-2} & \beta_{N_x-2} & \gamma_{N_x-2} \\ 0 & \cdots & 0 & \alpha_{N_x-1} & \beta_{N_x-1} \end{pmatrix} \begin{pmatrix} u_1^{n+1} \\ u_2^{n+1} \\ \vdots \\ u_{N_x-2}^{n+1} \\ u_{N_x-1}^{n+1} \end{pmatrix} \quad (13)$$

$$= \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_{N_x-2} \\ b_{N_x-1} - \gamma_{N_x-1} (L - Ke^{-r(n+1)\Delta\tau}) \end{pmatrix}.$$

Likewise, we have $u_{N_x}^{n+1} = Ce^{-r(n+1)\Delta\tau}$ for the cash-or-nothing option.

2.2.2. Dirichlet II Boundary Condition. The other Dirichlet boundary condition is simply setting the boundary values to

be fixed all the time with the payoff value. For the max call option, we set $u_{N_x}^{n+1} = L - K$:

$$\begin{pmatrix} \beta_1 & \gamma_1 & 0 & \cdots & 0 \\ \alpha_2 & \beta_2 & \gamma_2 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & \alpha_{N_x-2} & \beta_{N_x-2} & \gamma_{N_x-2} \\ 0 & \cdots & 0 & \alpha_{N_x-1} & \beta_{N_x-1} \end{pmatrix} \begin{pmatrix} u_1^{n+1} \\ u_2^{n+1} \\ \vdots \\ u_{N_x-2}^{n+1} \\ u_{N_x-1}^{n+1} \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_{N_x-2} \\ b_{N_x-1} - \gamma_{N_x-1} (L - K) \end{pmatrix}. \quad (14)$$

For the cash-or-nothing option, we set $u_{N_x}^{n+1} = C$. This boundary condition was used in [26].

2.2.3. Neumann Boundary Condition. To specify values for the derivative of the solution at the boundary of the spatial domain, we have the following equation from (9):

$$\frac{\partial u(x, \tau)}{\partial x} = N(d_1). \quad (15)$$

Therefore, for a sufficiently large L , we assume

$$\frac{\partial u(L, \tau)}{\partial x} = 1, \quad (16)$$

which we call by Neumann BC for European call option. Equation (16) can be discretized as $(u_{N_x}^{n+1} - u_{N_x-1}^{n+1})/h = 1$; that is, $u_{N_x}^{n+1} = u_{N_x-1}^{n+1} + h$. Therefore, we have

$$\begin{pmatrix} \beta_1 & \gamma_1 & 0 & \cdots & 0 \\ \alpha_2 & \beta_2 & \gamma_2 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & \alpha_{N_x-2} & \beta_{N_x-2} & \gamma_{N_x-2} \\ 0 & \cdots & 0 & \alpha_{N_x-1} & \beta_{N_x-1} + \gamma_{N_x-1} \end{pmatrix} \begin{pmatrix} u_1^{n+1} \\ u_2^{n+1} \\ \vdots \\ u_{N_x-2}^{n+1} \\ u_{N_x-1}^{n+1} \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_{N_x-2} \\ b_{N_x-1} - h\gamma_{N_x-1} \end{pmatrix}. \quad (17)$$

Similarly, applying BC in terms of the first derivative of x for the cash-or-nothing option, we can get

$$\frac{\partial u(x, \tau)}{\partial x} = \frac{Ce^{-(r\tau+d_2^2/2)}}{x\sigma\sqrt{2\pi\tau}} \quad (18)$$

from (11). Therefore, we can assume

$$\frac{\partial u(L, \tau)}{\partial x} = 0 \quad (19)$$

for a sufficiently large L . This represents Neumann BC for cash-or-nothing option. We now replace (19) with the one-sided derivative $(u_{N_x}^{n+1} - u_{N_x-1}^{n+1})/h = 0$; that is, $u_{N_x}^{n+1} = u_{N_x-1}^{n+1}$.

2.2.4. Linear Boundary Condition. Linear boundary condition assumes that the second derivative of the option value with respect to the underlying asset price x vanishes to zero for the large value of the asset price. To demonstrate this, we consider (9) at the right end of the domain. The second derivative value is given by

$$\frac{\partial^2 u(x, \tau)}{\partial x^2} = \frac{e^{-d_1^2/2}}{\sigma\sqrt{2\pi\tau}}. \quad (20)$$

If the asset price approaches to the large value L , we assume

$$\frac{\partial^2 u(L, \tau)}{\partial x^2} = 0, \quad (21)$$

which we call by linear BC for European call option. Equation (21) can be discretized as $(u_{N_x-1}^{n+1} - 2u_{N_x}^{n+1} + u_{N_x+1}^{n+1})/h^2 = 0$. By this relation, we obtain the boundary value as $u_{N_x+1}^{n+1} = 2u_{N_x}^{n+1} - u_{N_x-1}^{n+1}$. By substituting this in (3), we get

$$\begin{pmatrix} \beta_1 & \gamma_1 & 0 & \cdots & 0 \\ \alpha_2 & \beta_2 & \gamma_2 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & \alpha_{N_x-1} & \beta_{N_x-1} & \gamma_{N_x-1} \\ 0 & \cdots & 0 & \alpha_{N_x} - \gamma_{N_x} & \beta_{N_x} + 2\gamma_{N_x} \end{pmatrix} \begin{pmatrix} u_1^{n+1} \\ u_2^{n+1} \\ \vdots \\ u_{N_x-1}^{n+1} \\ u_{N_x}^{n+1} \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_{N_x-1} \\ b_{N_x} \end{pmatrix}. \quad (22)$$

Similarly, for the cash-or-nothing option, we have

$$\frac{\partial^2 u(x, \tau)}{\partial x^2} = \frac{Cd_1 e^{-(r\tau+d_2^2/2)}}{\sigma^2\tau\sqrt{2\pi}x^2} \quad (23)$$

from (11). Therefore, for a sufficiently large L , we have

$$\frac{\partial^2 u(L, \tau)}{\partial x^2} = 0. \quad (24)$$

In this case, we can obtain the same BC as European call option.

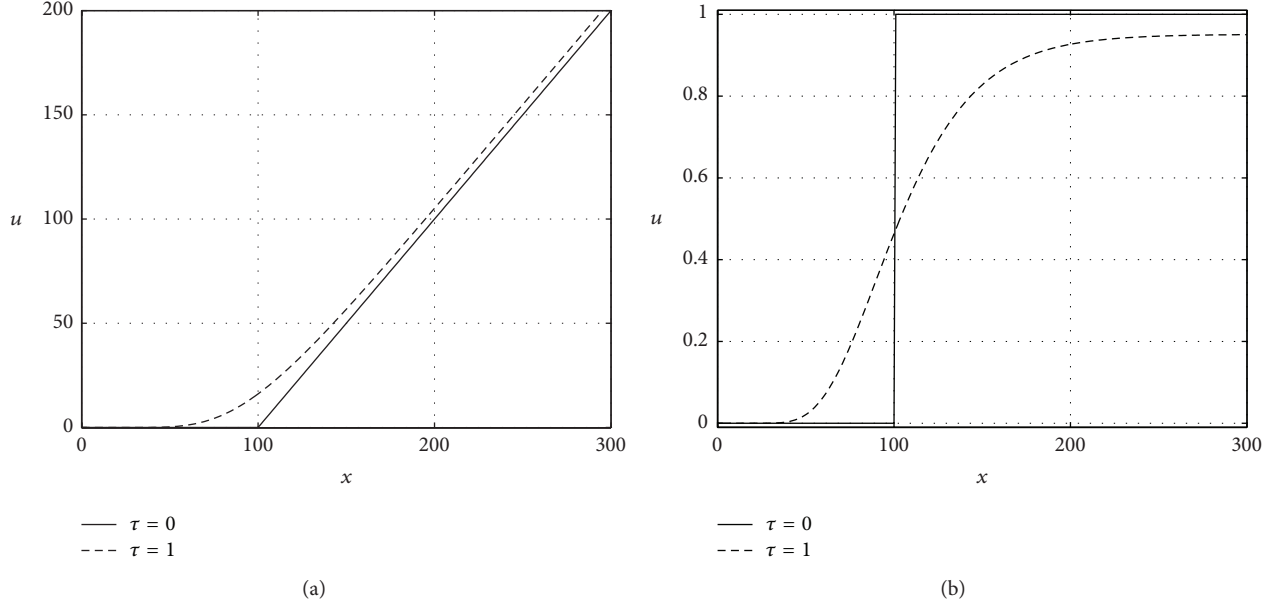


FIGURE 2: Values $u(x, \tau)$ of (a) European call option and (b) cash-or-nothing option at $\tau = 0$ and $\tau = 1$, respectively.

2.2.5. PDE Boundary Condition. Next we consider the PDE BC [23, 27]. Using the BS equation itself, we derive the BC. We use one-sided discretization:

$$\begin{aligned} \frac{u_{N_x}^{n+1} - u_{N_x}^n}{\Delta\tau} &= \frac{\sigma^2 x_{N_x}^2}{2} \frac{u_{N_x}^{n+1} - 2u_{N_x-1}^{n+1} + u_{N_x-2}^{n+1}}{h^2} \\ &+ rx_{N_x} \frac{u_{N_x}^{n+1} - u_{N_x-1}^{n+1}}{h} - ru_{N_x}^{n+1}, \end{aligned} \quad (25)$$

where the drift and volatility terms are discretized by using one-sided derivatives. The linear system of N_x equations can be written in the following matrix form:

$$\begin{pmatrix} \beta_1 & \gamma_1 & 0 & \cdots & 0 \\ \alpha_2 & \beta_2 & \gamma_2 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & \alpha_{N_x-1} & \beta_{N_x-1} & \gamma_{N_x-1} \\ 0 & \cdots & \alpha_{N_x} & \beta_{N_x} & \gamma_{N_x} \end{pmatrix} \begin{pmatrix} u_1^{n+1} \\ u_2^{n+1} \\ \vdots \\ u_{N_x-1}^{n+1} \\ u_{N_x}^{n+1} \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_{N_x-1} \\ b_{N_x} \end{pmatrix}, \quad (26)$$

where $\alpha_{N_x} = -\sigma^2 x_{N_x}^2 / 2h^2$, $\beta_{N_x} = \sigma^2 x_{N_x}^2 / h^2 + rx_{N_x} / h$, $\gamma_{N_x} = 1/\Delta\tau - \sigma^2 x_{N_x}^2 / 2h^2 - rx_{N_x} / h + r$, and $b_{N_x} = u_{N_x}^n / \Delta\tau$.

3. Numerical Experiments

To compare the above five BCs, we perform the numerical tests with European vanilla call option and cash-or-nothing option. Figures 2(a) and 2(b) show European call option and cash-or-nothing option prices at $\tau = 0$ and $\tau = 1$ on $\Omega = [0, 300]$, respectively. Here, $\sigma = 0.35$ and $r = 0.05$ are used. In the Appendix, we provide MATLAB codes for the closed-form solutions of these options.

In the following sections, unless otherwise specified, we use strike price $K = 100$, cash $C = 1$, the risk-free interest rate $r = 0.05$, and volatility $\sigma = 0.35$.

3.1. Convergence Test. First, we investigate the convergence of the numerical solutions with different BCs. To show this, we calculate the root mean square error with respect to spatial and temporal step sizes. Root mean square error (RMSE) is defined by

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{x_j \in [0.8K, 1.2K]} |u_j^n - v_j^n|^2}, \quad (27)$$

where N is the number of grid points on $[0.8K, 1.2K]$. Here, u is exact solution and v is numerical solution.

Table 1 represents the RMSE of European vanilla call option with five different BCs for varying h and $\Delta\tau$ at $T = 1$. As shown in Table 1, we can observe that the RMSE converges to zero with decreasing space and time step sizes. From now on, we will use the $\Delta\tau = 0.025$ and $h = 0.5$ for the following numerical tests.

TABLE 1: RMSE for European call option with varying h and $\Delta\tau$ with five different BCs.

h	$\Delta\tau$			
	0.2	0.1	0.05	0.025
Dirichlet I				
2.0	$1.4525e-3$	$1.4168e-3$	$1.3986e-3$	$1.3894e-3$
1.0	$2.5706e-4$	$2.5075e-4$	$2.4754e-4$	$2.4592e-4$
0.5	$4.5474e-5$	$4.4359e-5$	$4.3790e-5$	$4.3504e-5$
Dirichlet II				
2.0	$1.4560e-3$	$1.4197e-3$	$1.4010e-3$	$1.3917e-3$
1.0	$2.5768e-4$	$2.5126e-4$	$2.4797e-4$	$2.4631e-4$
0.5	$4.5583e-5$	$4.4445e-5$	$4.3867e-5$	$4.3574e-5$
Neumann				
2.0	$1.4533e-3$	$1.4173e-3$	$1.3989e-3$	$1.3896e-3$
1.0	$2.5719e-4$	$2.5085e-4$	$2.4759e-4$	$2.4595e-4$
0.5	$4.5490e-5$	$4.4375e-5$	$4.3799e-5$	$4.3509e-5$
Linear				
2.0	$1.4535e-3$	$1.4174e-3$	$1.3989e-3$	$1.3896e-3$
1.0	$2.5724e-4$	$2.5086e-4$	$2.4756e-4$	$2.4595e-4$
0.5	$4.5505e-5$	$4.4377e-5$	$4.3800e-5$	$4.3509e-5$
PDE				
2.0	$1.4532e-3$	$1.4173e-3$	$1.3989e-3$	$1.3896e-3$
1.0	$2.5715e-4$	$2.5084e-4$	$2.4759e-4$	$2.4595e-4$
0.5	$4.5476e-5$	$4.4373e-5$	$4.3799e-5$	$4.3509e-5$

3.2. *Pointwise Error on Different Domain Size L .* To study the effects of the various BCs, we perform the numerical tests on different domain size L . For comparison, we compute the pointwise error as the absolute difference between the numerical solution u_{num} and the closed-form solution u_{ex} ; that is, $\text{error} = |u_{\text{ex}} - u_{\text{num}}|$.

First, we take the payoff of a European vanilla call option (7). In Figure 3, left and right columns represent the option values and the pointwise errors versus x , respectively. In Figure 3, we compare the numerical results with three different domain sizes $L = 150, 200,$ and 300 . As domain size is large, there is no difference between the numerical results with five different boundary conditions and analytic solution at $x = L$. Also, pointwise error in $[0.8K, 1.2K]$ decreases when the domain size increases.

As second example, we test cash-or-nothing option. Similar to the previous case, we have same results as shown in Figure 4. When L is large, the pointwise error decreases. However, when the domain size is not sufficiently large, we have gap between the analytic and numerical solutions.

Therefore, it is important to choose the domain size L . Also, to reduce the numerical error, we need to choose the proper boundary condition.

3.3. *Root Mean Square Error.* The following numerical tests illustrate RMSE of various BCs with different r and σ . First we consider European call option with $h = 0.5, \Delta t = 0.025,$

$T = 1,$ and $L = 300$. Figures 5(a) and 5(b) represent the RMSE against the interest rate r and the volatility σ , respectively. Here, we use $\sigma = 0.35$ and $r = 0.05$ in Figures 5(a) and 5(b), respectively. As we can observe from Figure 5(a), the RMSE increases as r is large. Also, the RMSE versus σ has similar behavior in Figure 5(a). Through this test, we can see that Dirichlet I and Neumann BCs are more efficient than other BCs in case of r and σ .

Now, we consider cash-or-nothing option with $h = 0.5, \Delta t = 0.025, T = 1,$ and $L = 300$. Figure 6 shows the RMSE of various BCs versus r and σ . Here, we use $\sigma = 0.35$ and $r = 0.05$ in Figures 6(a) and 6(b), respectively. In this test, we can observe that Dirichlet II and Neumann BCs have lower RMSE than the other BCs for various r and σ .

Figure 7 illustrates RMSE of various BCs versus time T with European call option and cash-or-nothing option, respectively. As shown in Figure 7(a), we can see that the choice of BCs is important when τ is large. In addition, we can confirm that Neumann BC is efficient to reduce numerical error. Similarly, we observe that Neumann BC is efficient when L is small or T is large in Figure 7(b).

4. Conclusion

In this paper, we reviewed and studied the performance of the five different boundary conditions such as Dirichlet, Neumann, linear, and partial differential equation boundary

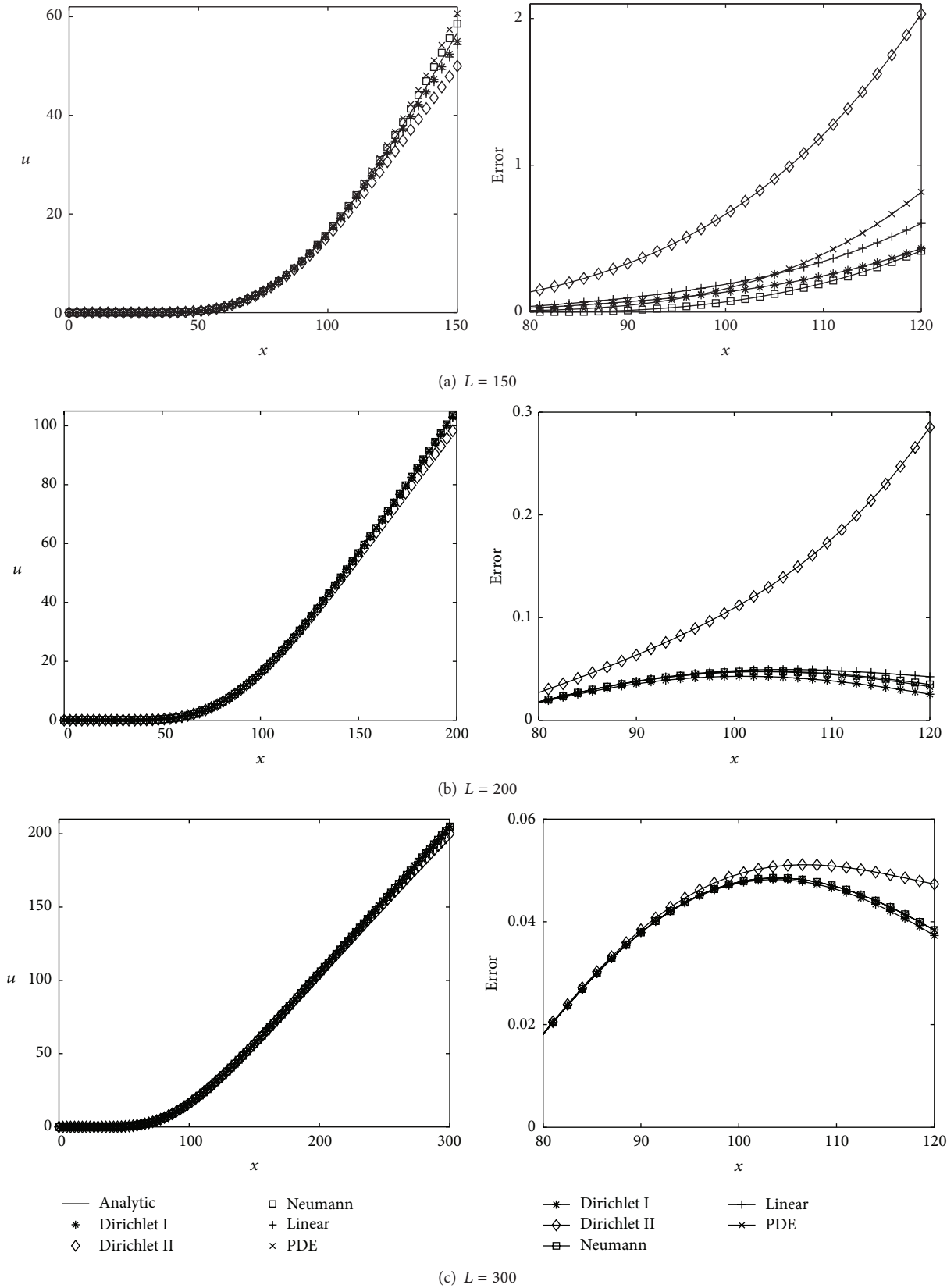
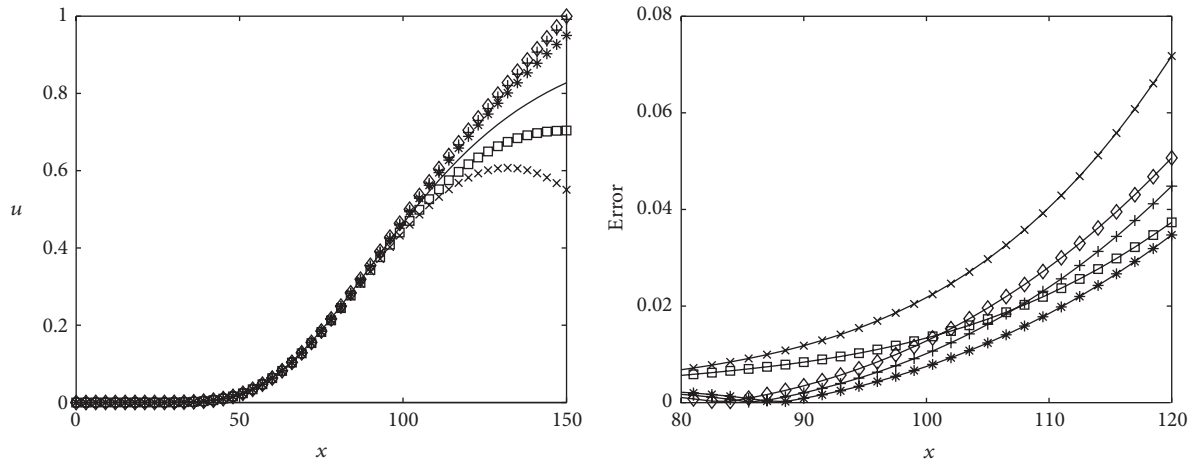
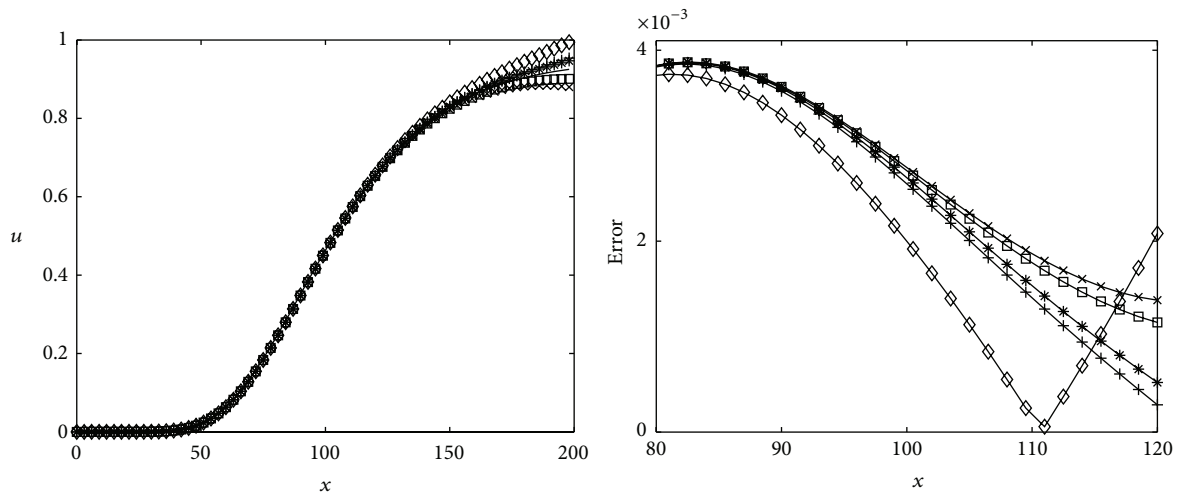


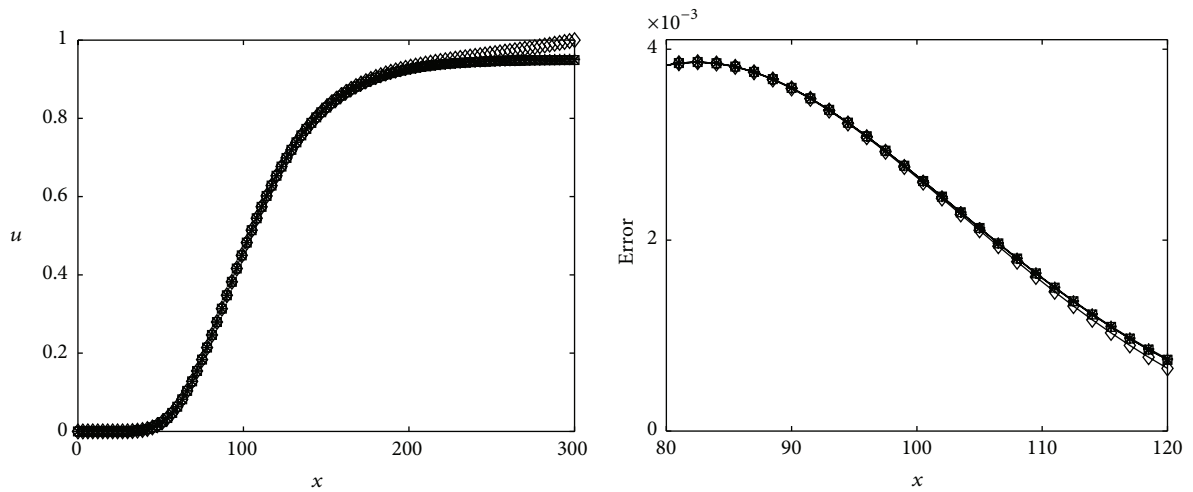
FIGURE 3: Comparison of European call option values (left column) and pointwise error (right column) with five BCs on different domain: (a) $L = 150$, (b) $L = 200$, and (c) $L = 300$, respectively.



(a) $L = 150$



(b) $L = 200$



— Analytic	□ Neumann	* Dirichlet I	+ Linear
* Dirichlet I	+ Linear	◇ Dirichlet II	× PDE
◇ Dirichlet II	× PDE		

(c) $L = 300$

FIGURE 4: Comparison of cash-or-nothing option values (left column) and pointwise error (right column) with five BCs on different domain: (a) $L = 150$, (b) $L = 200$, and (c) $L = 300$, respectively.

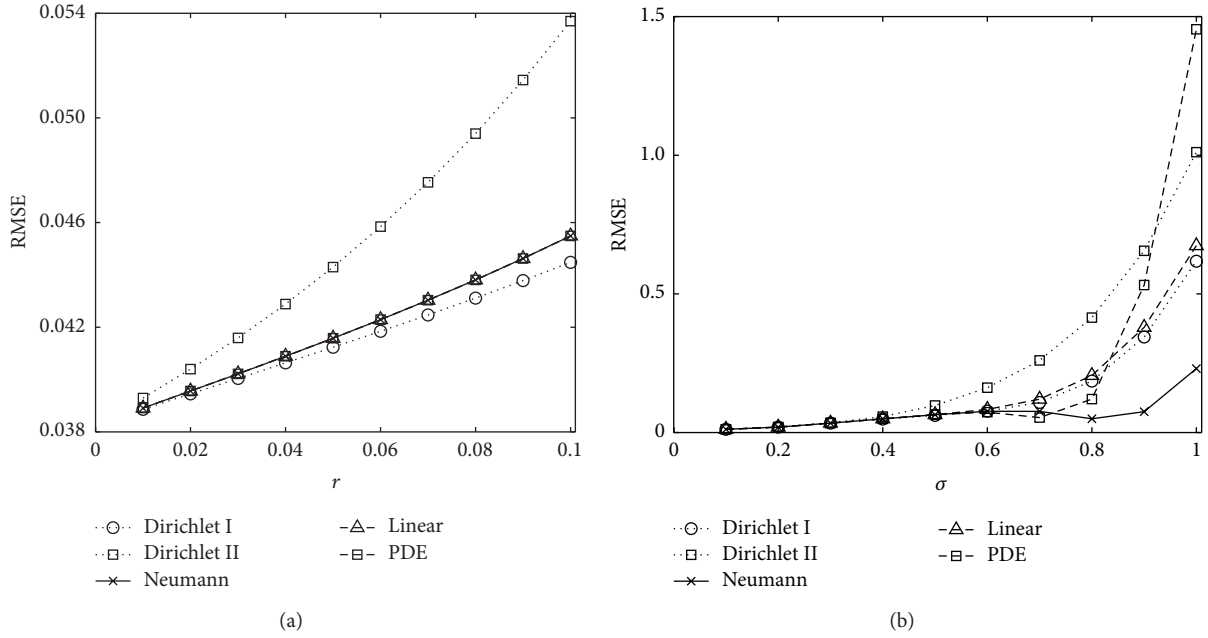


FIGURE 5: RMSE of various BCs versus (a) r and (b) σ with European call option.

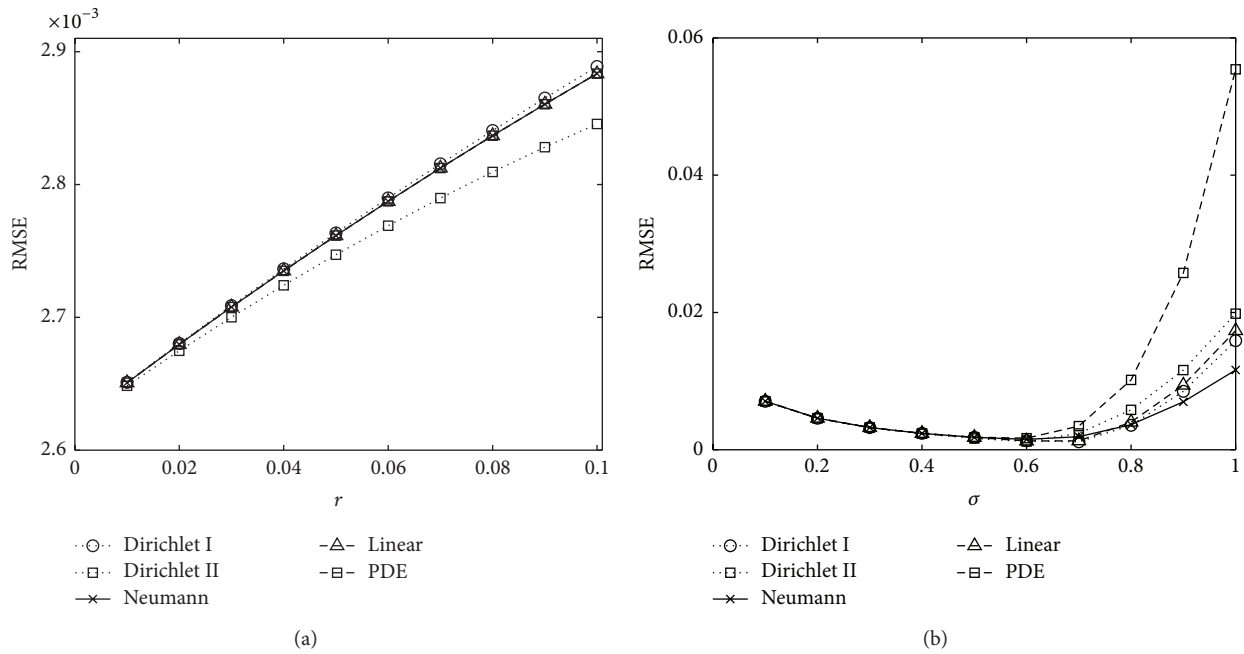


FIGURE 6: RMSE of various BCs versus (a) r and (b) σ with cash-or-nothing option.

conditions for the numerical solutions of the BS partial differential equation. We used a finite difference method to numerically solve the BS equation with the five different BCs. To show the efficiency of the given boundary condition, several numerical examples such as a convergence test, domain size effect, and parameter effect are presented. Numerical results suggested that the linear boundary condition is accurate and efficient among the other boundary conditions. As a future

research, we will investigate the BCs on the multidimensional BS equations.

Appendix

Evaluation of European call option price from call: we describe the formula, in MATLAB code, as shown in Algorithm 1.

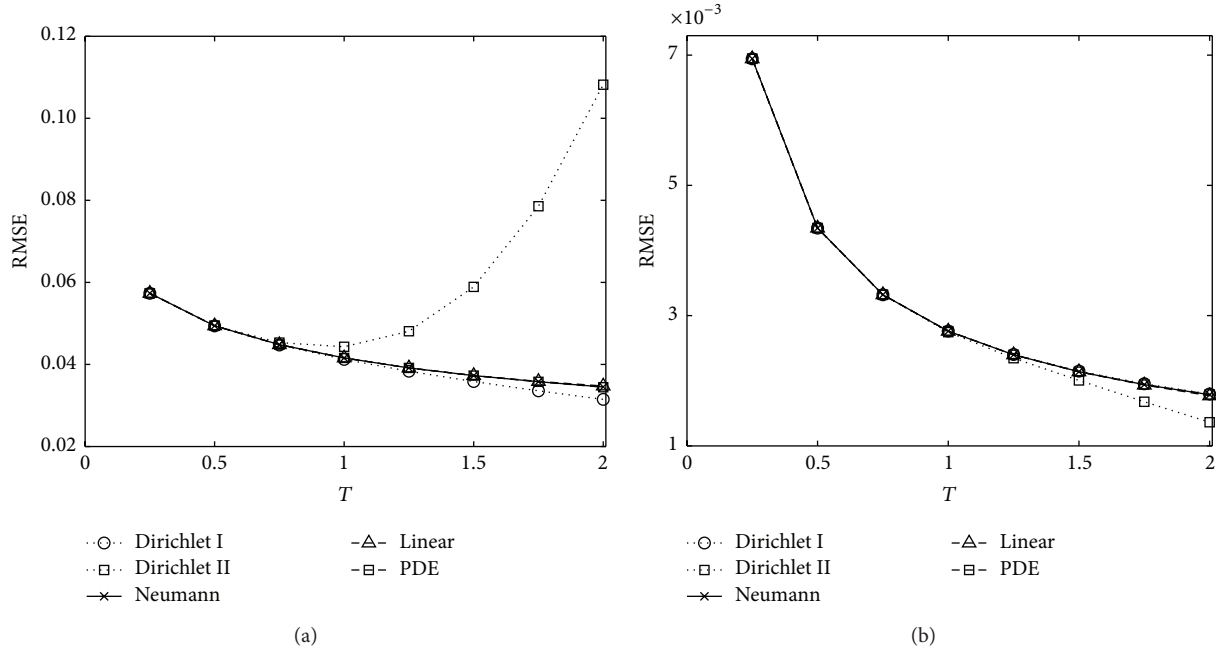


FIGURE 7: RMSE of various BCs versus T with (a) European call option and (b) cash-or-nothing option.

```

% European call option
clear;clc;clf;K=100;L=3*K;T=1.0;sigma=0.35;r=0.05;Nx=L+1;dt=0.01;
Nt=round(T/dt); x=linspace(0,L,Nx); h=x(2)-x(1);payoff=max(x-K,0);
d1=(log(x/K)+(r+sigma^2/2)*T)/(sigma*sqrt(T));d2=d1-sigma*sqrt(T);
exact=x.*normcdf(d1)-K*exp(-r*T).*normcdf(d2);grid on; hold on
plot(x,payoff,'k-',x,exact,'k-','LineWidth',1) axis image,
axis ([0 300 -12.5 220])
    
```

ALGORITHM 1

```

% Cash-or-nothing options
clear;clc;clf;K=100;L=3*K;T=1;sigma=0.35;r=0.05;Nx=L+1;dt=0.01;Nt=round(T/dt);
x=linspace(0,L,Nx);h=x(2)-x(1);
for i=1:Nx
if (x(i) <= K)
payoff(i)=0.0;
else payoff(i)=1.0;
end
end
d=(log(x/K)+(r-sigma^2/2)*T)/(sigma*sqrt(T));exact=1.0*exp(-r*T).*normcdf(d); grid on;
hold on;plot(x,payoff,'k-',x,exact,'r-','LineWidth',1)axis ([0 300 -0.05 1.05])
    
```

ALGORITHM 2

In a similar way, we can obtain the closed form of cash-or-nothing option, for MATLAB code as shown in Algorithm 2.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

The first author (Darae Jeong) was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2014R1A6A3A01009812). The second author (Seungsuk Seo) and the corresponding author

(Junseok Kim) were supported by The Small and Medium Business Administration (SMBA). The authors are grateful to the referees whose valuable suggestions and comments significantly improved the quality of this paper.

References

- [1] P. Wilmott, J. Dewynne, and S. Howison, *Option Pricing: Mathematical Models and Computation*, Oxford Financial Press, Oxford, UK, 1993.
- [2] M. J. Brennan and E. S. Schwartz, "Finite difference methods and jump processes arising in the pricing of contingent claims: a synthesis," *The Journal of Financial and Quantitative Analysis*, vol. 13, no. 3, pp. 461–474, 1978.
- [3] H. Han and X. Wu, "A fast numerical method for the Black-Scholes equation of American options," *SIAM Journal on Numerical Analysis*, vol. 41, no. 6, pp. 2081–2095, 2003.
- [4] J. Huang, Z. Cen, and A. Le, "A finite difference scheme for pricing American put options under Kou's jump-diffusion model," *Journal of Function Spaces and Applications*, vol. 2013, Article ID 651573, 11 pages, 2013.
- [5] O. Kudryavtsev, "Finite difference methods for option pricing under Lévy processes: wiener-Hopf factorization approach," *The Scientific World Journal*, vol. 2013, Article ID 963625, 12 pages, 2013.
- [6] X. Ruan, W. Zhu, S. Li, and J. Huang, "Option pricing under risk-minimization criterion in an incomplete market with the finite difference method," *Mathematical Problems in Engineering*, vol. 2013, Article ID 165727, 9 pages, 2013.
- [7] D. Jeong and J. Kim, "A comparison study of ADI and operator splitting methods on option pricing models," *Journal of Computational and Applied Mathematics*, vol. 247, pp. 162–171, 2013.
- [8] D. Jeong, T. Ha, M. Kim, J. Shin, I.-H. Yoon, and J. Kim, "An adaptive finite difference method using far-field boundary conditions for the Black-Scholes equation," *Bulletin of the Korean Mathematical Society*, vol. 51, no. 4, pp. 1087–1100, 2014.
- [9] Y. Achdou and N. Tchou, "Variational analysis for the Black and Scholes equation with stochastic volatility," *M2AN: Mathematical Modelling and Numerical Analysis*, vol. 36, no. 3, pp. 373–395, 2002.
- [10] A. Ern, S. Villeneuve, and A. Zanette, "Adaptive finite element methods for local volatility European option pricing," *International Journal of Theoretical and Applied Finance*, vol. 7, no. 6, pp. 659–684, 2004.
- [11] N. Rambeerich, D. Y. Tangman, M. R. Lollchund, and M. Bhuruth, "High-order computational methods for option valuation under multifactor models," *European Journal of Operational Research*, vol. 224, no. 1, pp. 219–226, 2013.
- [12] P. A. Forsyth and K. R. Vetzal, "Quadratic convergence for valuing American options using a penalty method," *SIAM Journal on Scientific Computing*, vol. 23, no. 6, pp. 2095–2122, 2002.
- [13] S. Wang, "A novel fitted finite volume method for the Black-Scholes equation governing option pricing," *IMA Journal of Numerical Analysis*, vol. 24, no. 4, pp. 699–720, 2004.
- [14] S. Wang, S. Zhang, and Z. Fang, "A superconvergent fitted finite volume method for Black-Scholes equations governing European and American option valuation," *Numerical Methods for Partial Differential Equations*, 2014.
- [15] S. N. I. Ibrahim, J. G. O'Hara, and N. Constantinou, "Pricing extendible options using the fast Fourier transform," *Mathematical Problems in Engineering*, vol. 2014, Article ID 831470, 7 pages, 2014.
- [16] S.-M. Zhang and L.-H. Wang, "A fast Fourier transform technique for pricing European options with stochastic volatility and jump risk," *Mathematical Problems in Engineering*, vol. 2012, Article ID 761637, 17 pages, 2012.
- [17] T. Sakuma and Y. Yamada, "Application of homotopy analysis method to option pricing under Lévy processes," *Asia-Pacific Financial Markets*, vol. 21, no. 1, pp. 1–14, 2014.
- [18] B. J. Kim, Y.-K. Ma, and H. J. Choe, "A simple numerical method for pricing an American put option," *Journal of Applied Mathematics*, vol. 2013, Article ID 128025, 7 pages, 2013.
- [19] D. J. Duffy, *Finite Difference Methods in Financial Engineering*, John Wiley & Sons, New York, NY, USA, 2006.
- [20] D. Jeong, I. S. Wee, and J. Kim, "An operator splitting method for pricing the ELS option," *Journal of KSIAM*, vol. 14, no. 3, pp. 175–187, 2010.
- [21] D. Jeong, J. Kim, and I.-S. Wee, "An accurate and efficient numerical method for Black-Scholes equations," *Communications of the Korea Mathematical Society*, vol. 24, no. 4, pp. 617–628, 2009.
- [22] R. Seydel, *Tools for Computational Finance*, Springer, Berlin, Germany, 2003.
- [23] D. Tavella and C. Randall, *Pricing Financial Instruments—The Finite Difference Method*, John Wiley & Sons, New York, NY, USA, 2000.
- [24] J. Topper, *Financial Engineering with Finite Elements*, John Wiley & Sons, Chichester, UK, 2005.
- [25] F. Black and M. Scholes, "The pricing of options and corporate liabilities," *Journal of Political Economy*, vol. 81, no. 3, pp. 637–659, 1973.
- [26] R. Kangro and R. Nicolaides, "Far field boundary conditions for Black-Scholes equations," *SIAM Journal on Numerical Analysis*, vol. 38, no. 4, pp. 1357–1368, 2000.
- [27] H. Windcliff, P. A. Forsyth, and K. R. Vetzal, "Analysis of the stability of the linear boundary condition for the Black-Scholes equation," *Journal of Computational Finance*, vol. 8, pp. 65–92, 2004.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

