FINITE DIFFERENCE METHOD FOR THE TWO-DIMENSIONAL BLACK–SCHOLES EQUATION WITH A HYBRID BOUNDARY CONDITION

YOUNGJIN HEO[1], HYUNSOO HAN[2], HANBYEOL JANG[2], YONGHO CHOI[3], AND JUNSEOK KIM[1†]

[1]DEPARTMENT OF MATHEMATICS, KOREA UNIVERSITY, SEOUL 02841, REPUBLIC OF KOREA
E-mail address: cfdkim@korea.ac.kr

[2]DEPARTMENT OF FINANCIAL ENGINEERING, KOREA UNIVERSITY, SEOUL 02841, REPUBLIC OF KOREA

[3]DEPARTMENT OF MATHEMATICS AND BIG DATA, DAEGU UNIVERSITY, GYEONGSAN-SI, GYEONGSANGBUK–DO 38453, REPUBLIC OF KOREA

ABSTRACT. In this paper, we develop an accurate explicit finite difference method for the two-dimensional Black–Scholes equation with a hybrid boundary condition. In general, the correlation term in multi-asset options is problematic in numerical treatments partially due to cross derivatives and numerical boundary conditions at the far field domain corners. In the proposed hybrid boundary condition, we use a linear boundary condition at the boundaries where at least one asset is zero. After updating the numerical solution by one time step, we reduce the computational domain so that we do not need boundary conditions. To demonstrate the accuracy and efficiency of the proposed algorithm, we calculate option prices and their Greeks for the two-asset European call and cash-or-nothing options. Computational results show that the proposed method is accurate and is very useful for nonlinear boundary conditions.

1. INTRODUCTION

An option is the right to buy or sell a basic product at a specific price in the future. It is important for financial companies or customers to know the price of option. There are many mathematical models and computation methods [1, 2, 3, 4, 5, 6, 7, 8] to determine option pricing. The Black–Scholes (BS) partial differential equation (PDE) [9] is commonly used to measure the option price and its Greeks.

In [10], the authors improved the model of Mortensen [11] and showed that a rich class of recovery rate scenarios could be incorporated into the model in a computationally manageable way. Paul Glasserman [12] focused on improving efficiency and discussed applications of the Monte Carlo simulation (MCS) [13, 14] on security pricing issues. Milstein [15] calculated the option price using the MCS and then calculated the Greeks. However, the MCS takes a lot of time because it has many execution time. Because the option value is different at each time, the fluctuation of the Greeks is too high.

One of the other numerical methods is the finite difference method (FDM) [16, 17, 18, 19]. Foulon [20] used semi-discretization of the Heston PDE by using FDM on non-uniform grids and solved the equation using the alternating direction implicit (ADI [21]) on two dimensions. In [22], the authors suggested operator splitting method (OSM) for solving the linear complementarity problems arising from the pricing of American options with two assets. In [23], the authors proposed the numerical contour integral method for free-boundary PDE to solve American volatility option pricing. In [25], the authors used the Laplace Transform Method with two free boundaries for pricing American CEV strangles Option.

The problematic part in solving a two-dimensional BS equation with an implicit method is the cross differential term. Therefore, the cross differential term is typically treated as an explicit manner. In particular, at the corner point, i.e., where both the underlying assets are maximum, the solution is highly dependent on the boundary treatment. In this paper, we extend the previous work [24] for the one-dimensional no boundary method to the two-dimensional BS equation:

$$\frac{\partial u}{\partial t} = -\frac{\sigma_1^2 x^2}{2}\frac{\partial^2 u}{\partial x^2} - \frac{\sigma_2^2 y^2}{2}\frac{\partial^2 u}{\partial y^2} - \rho\sigma_1\sigma_2 xy\frac{\partial^2 u}{\partial x\partial y} - rx\frac{\partial u}{\partial x} - ry\frac{\partial u}{\partial y} + ru \quad (1.1)$$
$$\text{for } (x,y,t) \in (0,\infty)\times(0,\infty)\times[0,T)$$

with the final payoff condition $u(x,y,T) = \text{payoff}(x,y)$ at expiry T, where x,y are the value of underlying assets, $u(x,y,t)$ is the option price, r is the risk–free rate, ρ is the correlation value between x,y and σ_1, σ_2 are the volatility of underlying assets, respectively. After applying the change of variable $\tau = T - t$ to Eq. (1.1) and truncate the infinite domain into a finite computational domain, we have the following natural initial boundary value problem

$$\frac{\partial u}{\partial \tau} = \frac{\sigma_1^2 x^2}{2}\frac{\partial^2 u}{\partial x^2} + \frac{\sigma_2^2 y^2}{2}\frac{\partial^2 u}{\partial y^2} + \rho\sigma_1\sigma_2 xy\frac{\partial^2 u}{\partial x\partial y} + rx\frac{\partial u}{\partial x} + ry\frac{\partial u}{\partial y} - ru \quad (1.2)$$
$$\text{for } (x,y,\tau) \in \Omega\times(0,T],$$

where $\Omega = (0, x_{max}) \times (0, y_{max})$.

The main purpose of this paper is to develop a hybrid boundary condition for the two-dimensional BS equation so that we can avoid extrapolation problem at the corner point and get stable numerical solutions.

This paper is organized as follows. In Section 2, we provide the numerical solution algorithm for the two-dimensional BS model. In Section 3, we present the computational results using the proposed hybrid boundary condition. Finally, conclusions are drawn in Section 4.

2. NUMERICAL SOLUTION

We discretize the BS equation on a non-uniform grid as shown in Fig. 1. Let $h_{i-1} = x_i - x_{i-1}$ and $h_{j-1} = y_j - y_{j-1}$ be the non-uniform space step sizes in the x- and y-directions, respectively. Here, $x_0 = y_0 = 0$. Let us denote $u(x_i, y_j, \tau^n)$ by u_{ij}^n, which is a numerical approximation of the solution, where $\tau^n = n\Delta\tau$ and $\Delta\tau$ is a time step. The two-dimensional
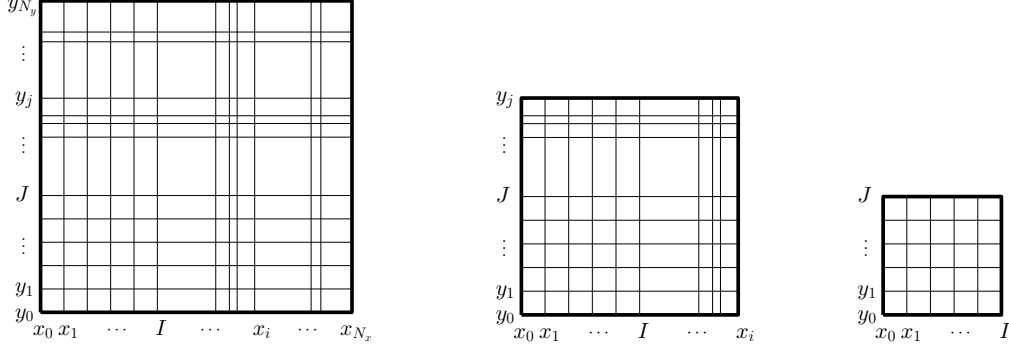
FIGURE 1. Temporal evolution of computational domain.

BS Eq. (1.2) is discretized as

$$\frac{u_{ij}^{n+1} - u_{ij}^n}{\Delta\tau} = \frac{(\sigma_1 x_i)^2}{2}\left(\frac{\partial^2 u}{\partial x^2}\right)_{ij}^n + \frac{(\sigma_2 y_j)^2}{2}\left(\frac{\partial^2 u}{\partial y^2}\right)_{ij}^n \tag{2.1}$$

$$+ \rho\sigma_1\sigma_2 x_i y_j\left(\frac{\partial^2 u}{\partial x \partial y}\right)_{ij}^n + r x_i\left(\frac{\partial u}{\partial x}\right)_{ij}^n + r y_j\left(\frac{\partial u}{\partial y}\right)_{ij}^n - r u_{ij}^n,$$

where

$$\left(\frac{\partial u}{\partial x}\right)_{ij}^n = \frac{h_{i-1}}{h_i(h_{i-1}+h_i)}u_{i+1,j}^n + \frac{h_i - h_{i-1}}{h_{i-1}h_i}u_{ij}^n - \frac{h_i}{h_{i-1}(h_{i-1}+h_i)}u_{i-1,j}^n,$$

$$\left(\frac{\partial u}{\partial y}\right)_{ij}^n = \frac{h_{j-1}}{h_j(h_{j-1}+h_j)}u_{i,j+1}^n + \frac{h_j - h_{j-1}}{h_{j-1}h_j}u_{ij}^n - \frac{h_j}{h_{j-1}(h_{j-1}+h_j)}u_{i,j-1}^n,$$

$$\left(\frac{\partial^2 u}{\partial x^2}\right)_{ij}^n = \frac{2}{h_i(h_{i-1}+h_i)}u_{i+1,j}^n - \frac{2}{h_{i-1}h_i}u_{ij}^n + \frac{2}{h_{i-1}(h_{i-1}+h_i)}u_{i-1,j}^n,$$

$$\left(\frac{\partial^2 u}{\partial y^2}\right)_{ij}^n = \frac{2}{h_j(h_{j-1}+h_j)}u_{i,j+1}^n - \frac{2}{h_{j-1}h_j}u_{ij}^n + \frac{2}{h_{j-1}(h_{j-1}+h_j)}u_{i,j-1}^n,$$

$$\left(\frac{\partial^2 u}{\partial x \partial y}\right)_{ij}^n = \frac{u_{ij}^n - u_{i,j-1}^n - u_{i-1,j}^n + u_{i-1,j-1}^n}{h_{i-1}h_{j-1}}.$$

Rewriting Eq. (2.1) gives

$$u_{ij}^{n+1} = \Delta\tau\left[\left(\frac{(\sigma_1 x_i)^2}{h_{i-1}(h_{i-1}+h_i)} - \frac{r x_i h_i}{h_{i-1}(h_{i-1}+h_i)} - \frac{\rho\sigma_1\sigma_2 x_i y_j}{h_{i-1}h_{j-1}}\right)u_{i-1,j}^n\right.$$

$$+ \left(\frac{(\sigma_2 y_j)^2}{h_{j-1}(h_{j-1}+h_j)} - \frac{r y_j h_j}{h_{j-1}(h_{j-1}+h_j)} - \frac{\rho\sigma_1\sigma_2 x_i y_j}{h_{i-1}h_{j-1}}\right)u_{i,j-1}^n$$

$$+ \left(\frac{1}{\Delta\tau} - \frac{(\sigma_1 x_i)^2}{h_{i-1}h_i} - \frac{(\sigma_2 y_j)^2}{h_{j-1}h_j} + \frac{\rho\sigma_1\sigma_2 x_i y_j}{h_{i-1}h_{j-1}} + \frac{h_i - h_{i-1}}{h_{i-1}h_i} + \frac{h_j - h_{j-1}}{h_{j-1}h_j} - r\right)u_{ij}^n$$

$$+\left(\frac{(\sigma_1 x_i)^2}{h_i(h_{i-1}+h_i)}+\frac{rx_i h_{i-1}}{h_i(h_{i-1}+h_i)}\right)u_{i+1,j}^n$$

$$+\left(\frac{(\sigma_2 y_j)^2}{h_j(h_{j-1}+h_j)}+\frac{ry_j h_{j-1}}{h_j(h_{j-1}+h_j)}\right)u_{i,j+1}^n+\frac{\rho\sigma_1\sigma_2 x_i y_j}{h_{i-1}h_{j-1}}u_{i-1,j-1}^n\Bigg], \qquad (2.2)$$

$$\text{for } i=1,...,N_x,\ j=1,...,N_y,\ \text{and } n=0,...,N_\tau-1,$$

where N_x, N_y, and $N_\tau = T/\Delta\tau$ are the number of grid points in the x-, y-, and τ- direction, respectively. To make the coefficient of u_{ij}^n be positive, the following should be satisfied:

$$\frac{1}{\Delta\tau} > \frac{(\sigma_1 x_i)^2}{h_{i-1}h_i}+\frac{(\sigma_2 y_j)^2}{h_{j-1}h_j}-\frac{\rho\sigma_1\sigma_2 x_i y_j}{h_{i-1}h_{j-1}}-\frac{h_i-h_{i-1}}{h_{i-1}h_i}-\frac{h_j-h_{j-1}}{h_{j-1}h_j}+r. \qquad (2.3)$$

Suppose the step size is uniform, i.e., $h_i = h_j = h$ for some constant h. Then, we have

$$\Delta\tau < f(x_i,y_j) = \frac{h^2}{(\sigma_1 x_i)^2+(\sigma_2 y_j)^2+rh^2-\rho\sigma_1\sigma_2 x_i y_j}.$$

In Fig. 2, (a) and (b) show a mesh plot and contour lines of $f(x_i, y_j)$, respectively, with $h = 10$, $\sigma_1 = \sigma_2 = 0.3$, $r = 0.015$, and $\rho = 0.3$. We can observe that the minimum value is obtained at the upper right domain corner.
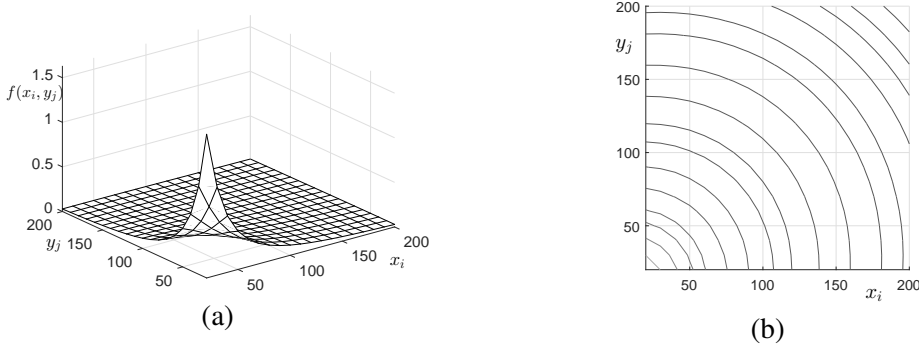

(a)

(b)

FIGURE 2. (a) Mesh plot and (b) contour lines of $f(x_i, y_j)$.

Let a temporary time step be

$$\Delta\tau_{tmp} = \frac{sh^2}{(\sigma_1 x_I)^2+(\sigma_2 y_J)^2+rh^2-\rho\sigma_1\sigma_2 x_I y_J},$$

where s $(0 < s < 1)$ is a safety factor and I, J are maximum indices of an interested region. Let $N_\tau = [T/\Delta\tau_{tmp}]+1$ be the total number of time steps, where $[k]$ means the largest integer not greater than k. We redefine the time step [24] as

$$\Delta\tau = T/N_\tau. \qquad (2.4)$$

Then, from Eq. (2.3) with $y_j = x_i$ and $h_j = h_i$, we have

$$h_i = \frac{1}{s} \times \frac{(\sigma_1^2 + \sigma_2^2)x_i^2/h_{i-1} + 2}{s/\Delta\tau + \rho\sigma_1\sigma_2 x_i^2/h_{i-1}^2 + 2/h_{i-1} - r}, \text{ for } i > I, \tag{2.5}$$

where $1/s$ is a safety factor. Uniform grid ($h = 4$) is used up to an index I and then non-uniform grid (2.5) is used. Fig. 3 shows h_i against index i with $s = 0.99$, $\sigma_1 = \sigma_2 = 0.3$, $r = 0.015$, $\rho = 0.3$, and $\Delta\tau = 0.0076$. Now, we describe the main process of the proposed
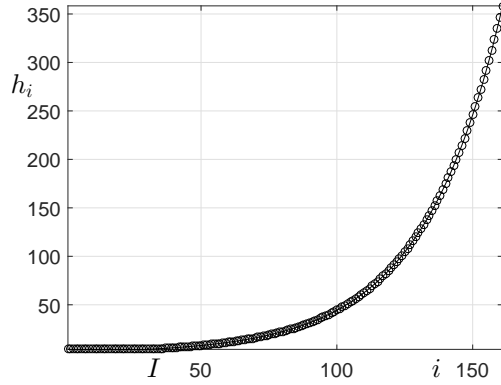

FIGURE 3. Grid size h_i.

algorithm. We set the grid size h_i using uniform grid and Eq. (2.5). Then, we solve the Eq. (2.2) by using a hybrid boundary condition: a linear boundary condition is applied at $x = 0$ or $y = 0$, and the computation domain is reduced by one grid in each direction at the other boundaries. Therefore, as we proceed the computation, the domain size decreases as shown in Fig. 1.

3. NUMERICAL EXPERIMENTS

We numerically compute the European call option and the cash-or-nothing option and then compare with each analytic solution to show the accuracy of our proposed method. We use the relative error to compare the results. The relative error is defined as follows:

$$error = V_{analytic} - V,$$

where $V_{analytic}$ is the analytic solution and V is the numerical solution. All tests were run on a 2.7 GHZ Intel PC with 4 GB of RAM loaded with MATLAB 2016a [26].

3.1. **European call option of two-asset.** First, let us consider a European call option [27] whose payoff is given as

$$u(x, y, 0) = \max\{x - K_1, y - K_2, 0\}, \tag{3.1}$$

where K_1 and K_2 are the strike prices of x and y, respectively. Fig. 4(a) shows the payoff function (3.1). Fig. 4(b)–(d) show the snapshots of the option price at $\tau = 0.7634, 0.8397$, and 1, respectively. Here, we used the following parameters: $s = 0.99$, $\sigma_1 = \sigma_2 = 0.3$, $r = 0.015$, $\rho = 0.3$, $T = 1$, $K_1 = K_2 = 100$, and $h = 4$. Using the above-mentioned time step (2.4)
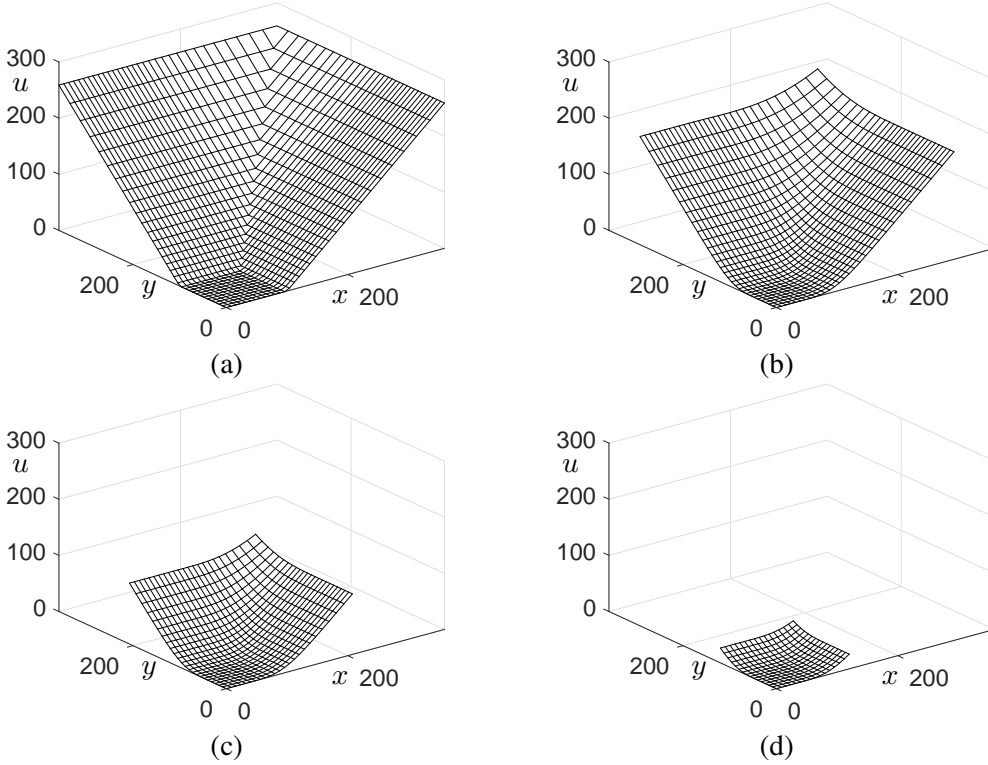

FIGURE 4. (a) Payoff of a European call option on the maximum of two-asset. (b)–(d) Snapshots of European call option at three different times, $\tau = 0.7634$, 0.8397, and 1, respectively.

and grid (2.5), we solve the European call option as shown in Fig. 4. The analytic solution is given in Appendix. We compared with operator splitting method(OSM) with conventional linear boundary condition, the method we present in the paper can get a more good result, which is shown in the Fig. 5. As shown in the Fig. 5(a), the price graph calculated by the OSM and conventional linear boundary condition shows that the value decreases toward the end of the domain. The conventional linear boundary condition has difficulty that generates the bad results at boundary when the high correlation in multi-dimensional problem or derivative with nonlinear payoff. However, in Fig. 5(b), the price graph that calculated by proposed method has good result. In addition, we compute the numerical Greeks and compare the difference with the analytic Greeks. The variables used here are the same as those used to calculate the
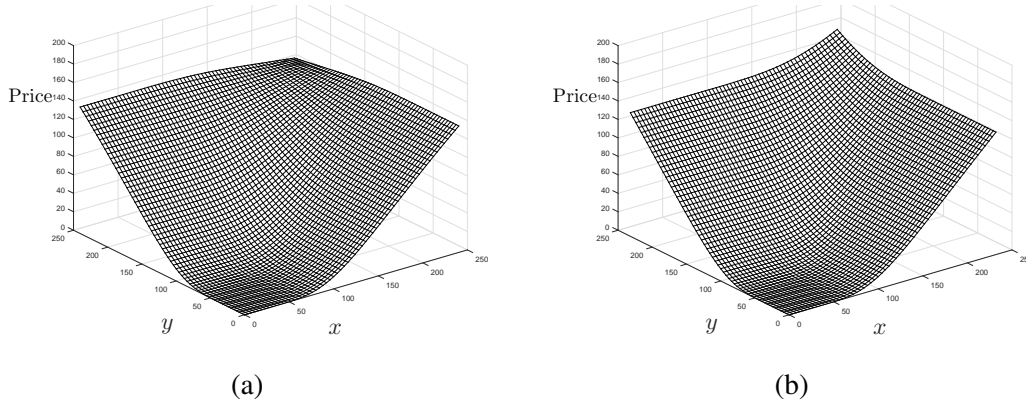
FIGURE 5. (a) European call option price using implicit finite difference scheme (b) European call option price using proposed scheme.

option price. The characters in Table 1 mean that: Delta (Δ) is the derivative of the option price relative to the underlying asset, $\frac{\partial u}{\partial x}$ and $\frac{\partial u}{\partial y}$. Gamma ($\Gamma$) is the second derivative of the option price, $\frac{\partial^2 u}{\partial x^2}$, $\frac{\partial^2 u}{\partial y^2}$, and $\frac{\partial^2 u}{\partial x \partial y}$. Theta ($\Theta$) is the derivative of the option price over time, $\frac{\partial u}{\partial t}$. Rho ($\mathcal{R}$) is the derivative of the option price with respect to the risk-free interest rate, $\frac{\partial u}{\partial r}$. Vega ($\mathcal{V}$) is the derivative of option price with respect to volatility, $\frac{\partial u}{\partial \sigma_1}$ and $\frac{\partial u}{\partial \sigma_2}$. We use finite difference schemes for evaluating the Greeks [28].

$$\Delta_x = \frac{\partial u}{\partial x} \approx \frac{u(x_{i+1}, y, \tau) - u(x_{i-1}, y, \tau)}{2h},$$

$$\Delta_y = \frac{\partial u}{\partial y} \approx \frac{u(x, y_{j+1}, \tau) - u(x, y_{j-1}, \tau)}{2h},$$

$$\Gamma_{xx} = \frac{\partial^2 u}{\partial x^2} \approx \frac{u(x_{i+1}, y, \tau) - 2u(x, y, \tau) + u(x_{i-1}, y, \tau)}{h^2},$$

$$\Gamma_{yy} = \frac{\partial^2 u}{\partial y^2} \approx \frac{u(x, y_{j+1}, \tau) - 2u(x, y, \tau) + u(x, y_{j-1}, \tau)}{h^2},$$

$$\Gamma_{xy} = \frac{\partial^2 u}{\partial x \partial y} \approx \frac{u(x_{i+1}, y_{j+1}, \tau) - u(x_{i-1}, y_{j+1}, \tau) - u(x_{i+1}, y_{j-1}, \tau) + u(x_{i-1}, y_{j-1}, \tau)}{4h^2},$$

$$\Theta = -\frac{\partial u}{\partial \tau} \approx -\frac{u(x, y, \tau + \Delta\tau) - u(x, y, \tau - \Delta\tau)}{2\Delta\tau},$$

$$\mathcal{R} = \frac{\partial u}{\partial r} \approx \frac{u(x, y, \tau; r + \Delta r) - u(x, y, \tau; r - \Delta r)}{2\Delta r},$$

$$\mathcal{V}_x = \frac{\partial u}{\partial \sigma_1} \approx \frac{u(x, y, \tau; \sigma_1 + \Delta\sigma_1) - u(x, y, \tau; \sigma_1 - \Delta\sigma_1)}{2\Delta\sigma_1},$$

$$\mathcal{V}_y = \frac{\partial u}{\partial \sigma_2} \approx \frac{u(x, y, \tau; \sigma_2 + \Delta\sigma_2) - u(x, y, \tau; \sigma_2 - \Delta\sigma_2)}{2\Delta\sigma_2}.$$

Table 1 lists the price and the Greeks of European call option and error at current price, $(x, y) = (100, 100)$.

TABLE 1. Price and the Greeks of European call option and error at current price, $(x, y) = (100, 100)$.

Case	u	Δ_x	Δ_y	Γ_{xx}	Γ_{yy}
Analytic solution	20.6131	0.4282	0.4284	0.0129	0.0129
Numerical solution	20.5899	0.4231	0.4267	0.0126	0.0128
Error	0.0232	0.0051	0.0017	0.0003	0.0001
Case	Γ_{xy}	Θ	$\mathcal{R}$	$\mathcal{V}_x$	$\mathcal{V}_y$
Analytic solution	-0.0061	10.9966	65.0680	33.3215	33.3215
Numerical solution	-0.0060	10.8658	49.2500	32.7857	32.9264
Error	0.0001	0.1308	15.818	0.5358	0.3951

3.2. **Two-asset Cash-or-Nothing Option.** Next, we consider the two-asset cash-or-nothing option [27]. The payoff of the option is given as

$$u(x, y, 0) = \begin{cases} \text{Cash} & \text{if} \quad x \geq K_1 \text{ and } y \geq K_2, \\ 0 & \text{otherwise}, \end{cases} \tag{3.2}$$

where K_1 and K_2 are the strike prices of x and y, respectively (see Fig. 6 (a)). The analytic solution is obtained from a closed-form solution, which is provided in Appendix. The parameters used are: $\sigma_1 = \sigma_2 = 0.3$, $r = 0.015$, $\rho = 0.3$, $T = 1$, Cash $= 100$, $K_1 = K_2 = 100$ and $h = 4$. Because cash-or-nothing option has a rapid evolution at the strike price point, we wrap the strick price point and solve Eq. (2.2) by using the proposed scheme. We calculate the cash-or-nothing option and its Greeks just as we did for the European call option. Fig. 6 shows the payoff function and snapshots of the solution at three different times. Fig. 7 show the prices of the options borrowed by the OSM and proposed methods, respectively. Unlike the Fig. 5, noticeable differences between Fig. 7(a) and Fig. 7(b) are not identifiable. The reason for this comes from the payoff structure of cash or nothing option. Table 2 lists the price and the Greeks of cash-or-nothing option and error at current price, $(x, y) = (100, 100)$.
Checking the results in Tables 1 and 2 confirm that the analytic solution and numerical solution values are similar in almost all cases. This is effective because you can get similar value to the analytic solution like the above results without solving linear boundary treatment problems.
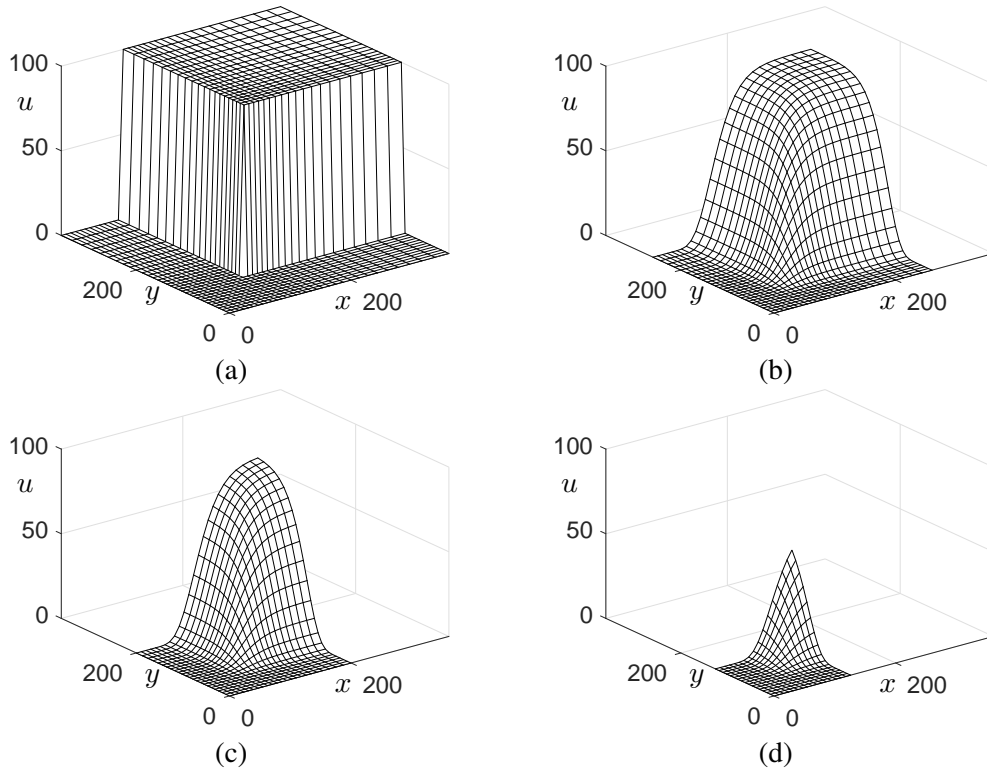
FIGURE 6. (a) Payoff of a cash-or-nothing option of two-asset. (b)–(d) Snap-shots of cash-or-nothing option at three different times, $\tau = 0.8451, 0.9014$, and 1, respectively.
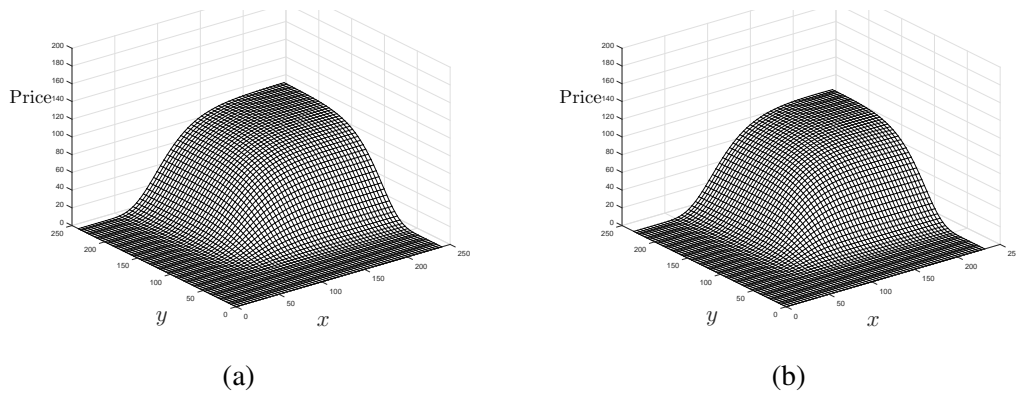

FIGURE 7. (a) European cash or nothing option price using implicit finite difference scheme (b) European cash or nothing option price using proposed scheme

TABLE 2. Price and the Greeks of cash-or-nothing option and error at current price, $(x, y) = (100, 100)$.

Case	u	Δ_x	Δ_y	Γ_{xx}	Γ_{yy}
Analytic solution	25.6305	0.6111	0.6111	-0.0187	-0.0187
Numerical solution	25.4787	0.5971	0.6001	-0.0188	-0.0183
Error	0.1518	0.0140	0.0110	0.0001	-0.0004
Case	Γ_{xy}	Θ	$\mathcal{R}$	$\mathcal{V}_x$	$\mathcal{V}_y$
Analytic solution	0.0224	-3.6166	96.8067	-12.4815	-12.4815
Numerical solution	0.0220	-2.4106	73.3093	-12.5675	-12.4459
Error	0.0004	-1.2060	23.4974	0.0860	-0.0356

4. CONCLUSION

In this article, we proposed the accurate explicit hybrid finite difference method for the two-dimensional BS equation. The proposed method can avoid the numerical boundary treatment problem of the cross derivatives at the domain corner. The main idea of the algorithm is reducing the computational domain by each direction so that we do not need boundary conditions. Numerical tests for the pricing and Greeks of the two-asset European options demonstrate the accuracy and efficiency of the proposed algorithm.

APPENDIX

The closed-form formula for the option price (3.1) given in [27] is as follows:

$$
\begin{aligned}
u(x, y, T) &= xM(d_1, d; \rho_1) + yM\big(d_2, -d + \sigma\sqrt{T}; \rho_2\big) \\
&\quad - Xe^{-rT}\Big[1 - M\big(-d_1 + \sigma_1\sqrt{T}, -d_2 + \sigma_2\sqrt{T}; \rho\big)\Big],
\end{aligned}
$$

where $d = \frac{\ln(x/y) + 0.5\sigma^2 T}{\sigma\sqrt{T}}$, $d_1 = \frac{\ln(x/X) + (r + 0.5\sigma_1^2)T}{\sigma_1\sqrt{T}}$, $d_2 = \frac{\ln(y/X) + (r + 0.5\sigma_2^2)T}{\sigma_2\sqrt{T}}$, $\sigma = \sqrt{\sigma_1^2 + \sigma_2^2 - 2\rho\sigma_1\sigma_2}$, $\rho_1 = (\sigma_1 - \rho\sigma_2)/\sigma$, and $\rho_2 = (\sigma_2 - \rho\sigma_1)/\sigma$.
$M(m, n; \rho)$ is a standard cumulative normal distribution function:

$$
M(m, n; \rho) = \frac{1}{2\pi\sqrt{1 - \rho^2}} \int_{-\infty}^{m} \int_{-\infty}^{n} e^{-\frac{\xi^2 - 2\rho\xi\eta + \eta^2}{2(1 - \rho^2)}} \, d\xi \, d\eta.
$$

```
        %%%%%%%%% Option value %%%%%%%%%
clear; sigma1=0.3; sigma2=0.3; r=0.015; rho=0.3; T=1; X=100;
x=100;y=100; sig=sqrt(sigma1^2+sigma2^2-2*rho*sigma1*sigma2);
rho1=(sigma1-rho*sigma2)/sig; rho2=(sigma2-rho*sigma1)/sig;
d=(log(x/y)+0.5*sig^2*T)/(sig*sqrt(T));
y1=(log(x/X)+0.5*sigma1^2*T)/(sig*sqrt(T));
y2=(log(y/X)+0.5*sigma2^2*T)/(sig*sqrt(T));
V=x*mvncdf([y1 d],[0 0],[1 rho1; rho1 1]) ...
```

```
+y*mvncdf([y2 -d+sig*sqrt(T)],[0 0],[1 rho2; rho2 1]) ...
-X*exp(-r*T)*(1-mvncdf([-y1+sigma1*sqrt(T) ...
-y2+sigma2*sqrt(T)],[0 0],[1 rho; rho 1]))
```

The closed-form formula for the option price (3.2) given in [27] is as follows:

$$u(x,y,\tau) = Ke^{-r\tau}M(\alpha,\beta;\rho),$$

where $\alpha = \frac{\ln(x/K_1)+(r-\sigma_1^2/2)\tau}{\sigma_1\sqrt{\tau}}, \beta = \frac{\ln(y/K_2)+(r-\sigma_2^2/2)\tau}{\sigma_2\sqrt{\tau}}$.

```
              %%%%%%%% Option value %%%%%%%%%
clear;tau=0.5;K1=100;K2=100; Cash=100;r=0.03;
sig1=0.3;sig2=0.3;rho=0.5;x=100;y=100;
a=(log(x/K1)+(r-sig1^2/2)*tau)/(sig1*sqrt(tau));
b=(log(y/K2)+(r-sig2^2/2)*tau)/(sig2*sqrt(tau));
V=Cash*exp(-r*tau)*mvncdf([a b],[0 0],[1 rho;rho 1])
```

REFERENCES

[1] J. Topper, *Financial Engineering with Finite Elements*, John Wiley & Sons., 2005.

[2] P. Wilmott, J. Dewynneand, and S. Howisson, *Option Pricing: Mathematical Models and Computation*, Oxford Press, Oxford, 1993.

[3] R. Seydel, *Tools for Computational Finance*, Springer-Verlag, Berlin, 2003.

[4] Y. Achdou and P. Olivier, *Computational methods for option pricing*, Society for Industrial and Applied Mathematics, 2005.

[5] M. Mrazek, J. Pospisil and T. Sobotka, *On calibration of stochastic and fractional stochastic volatility models*, European Journal of Operational Research, **254** (2016), 1036–1046.

[6] M. Alghalith, *Pricing the American options using the Black& Scholes pricing formula*, Physica A. 2018.

[7] N.Laskin, *Valuing options in shot noise market*, Physica A, **502** (2018), 518–533.

[8] L. Lin, Y. Li, and J. Wu, *The pricing of European options on two underlying assets with delays*, Physica A, **495** (2018), 143–151.

[9] F. Black and S. Myron, *The pricing of options and corporate liabilities*, Journal of political economy, **81** (1973), 637–654.

[10] A. Eckner, *Computational techniques for basic affine models of portfolio credit risk*, Journal of Computational Finance, **13** (2009) 63–102.

[11] A. Mortensen, *Semi-Analytical Valuation of Basket Credit Derivatives in Intensity-Based Models*, The journal of derivatives, **13** (2006), 8–26.

[12] P. Boyle, M. Broadie and P. Glasserman, *Monte Carlo methods for security pricing*, Journal of Economic Dynamics & Control, **21** (1997), 1267–1321.

[13] C.Z. Mooney, *Monte Carlo Simulation*, Sage, London, **116**, 1997.

[14] C.J. Wang and M.Y. Kao, *Optimal search for parameters in Monte Carlo simulation for derivative pricing*, European Journal of Operational Research, **249** (2016), 683–690.

[15] G.N. Milstein, M. V. Tretyakov, *Numerical analysis of Monte Carlo evaluation of Greeks by finite differences*, Journal of Computational Finance, **8** (2005), 1–34.

[16] Z. Cen and L. Anbo, *A robust and accurate finite difference method for a generalized Black-Scholes equation*, Journal of Computational and Applied Mathematics, **235**0 (2011), 3728–3733.

[17] D. Tavella and C. Randall, *Pricing Financial Instruments: the Finite Difference Method*, John Wiley & Sons. 2000.

[18] D.J. Duffy, *Finite Difference methods in financial engineering: a Partial Differential Equation approach*, John Wiley & Sons. 2006.

[19] J. Toivanen, *A high-order front-tracking finite diffrence method for pricing American options under jump-diffsion models*, The Journal of Computational Finance, **13** (2010), 61–79.

[20] K.J. Hout and S. Foulon, *ADI finite difference schemes for option pricing in the Heston model with correlation*, International Journal of Numerical Analysis & Modeling, **7** (2010), 303–320.

[21] S.P. Zhu and W.T. Chen, *A predictor-corrector scheme based on the ADI method for pricing American puts with stochastic volatility*, Computers & Mathematics with Applications, **62** (2011), 1–26.

[22] S. Ikonen and J. Toivanen, *Operator splitting methods for American option pricing*, Applied mathematics letters, **17** (2004), 809–814.

[23] Y. chen and J.Ma, *Numerical Contour Integral Methods for Free-Boundary Partial Differential Equations Arising in American Volatility Options Pricing*, Discrete Dynamics in Nature and Society, **2018** (2018), 61–74.

[24] D. Jeong, M. Yoo and J. Kim, *Finite Difference Method for the Black–Scholes Equation Without Boundary Conditions*, Computational Economics, (2017), 1–12.

[25] Z. Zhou and H. Wu, *Laplace Transform Method for Pricing American CEV Strangles Option with Two Free Boundaries*, Discrete Dynamics in Nature and Society, **2018** (2018).

[26] Using MATLAB, http://www.mathworks.com/, The MathWorks Inc.Natick, MA, 1998.

[27] E.G. Haug, *The complete guide to option pricing formulas*, McGraw-Hill Companies, 2007.

[28] S. Lee, Y. Li, Y. Choi, H. Hwang and J. Kim, *Accurate and efficient computations for the Greeks of European multi-asset options*, Journal of Korean Society for Industrial and Applied Mathmatics, **18** (2014), 61–74.