Hindawi Publishing Corporation
Discrete Dynamics in Nature and Society
Volume 2016, Article ID 1586786, 12 pages
http://dx.doi.org/10.1155/2016/1586786

Research Article

Accurate and Efficient Computations of the Greeks for Options Near Expiry Using the Black-Scholes Equations

Darae Jeong,[1] Minhyun Yoo,[2] and Junseok Kim[1]

[1]*Department of Mathematics, Korea University, Seoul 136-713, Republic of Korea*
[2]*Department of Financial Engineering, Korea University, Seoul 136-701, Republic of Korea*

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

Received 6 January 2016; Accepted 13 March 2016

Academic Editor: Francisco R. Villatoro

We investigate the accurate computations for the Greeks using the numerical solutions of the Black-Scholes partial differential equation. In particular, we study the behaviors of the Greeks close to the maturity time and in the neighborhood around the strike price. The Black-Scholes equation is discretized using a nonuniform finite difference method. We propose a new adaptive time-stepping algorithm based on local truncation error. As a test problem for our numerical method, we consider a European cash-or-nothing call option. To show the effect of the adaptive stepping strategy, we calculate option price and its Greeks with various tolerances. Several numerical results confirm that the proposed method is fast, accurate, and practical in computing option price and the Greeks.

1. Introduction

In this paper, we investigate the accurate and efficient computations for the Greeks using the numerical solutions of the Black-Scholes (BS) partial differential equation (PDE) [1]. Let s_i and t denote the price of the underlying asset and time, respectively. Let $\mathbf{s} = (s_1, s_2, \ldots, s_n)$ be the set of n underlying assets. Then, the value of an option $u(\mathbf{s}, t)$ is governed by the following n-dimensional BS equation [2].

For $(\mathbf{s}, t) \in \mathbb{R}_+^d \times [0, T)$,

$$\frac{\partial u(\mathbf{s},t)}{\partial t} + \sum_{i=1}^{d} r s_i \frac{\partial u(\mathbf{s},t)}{\partial s_i} + \frac{1}{2}\sum_{i,j=1}^{d} \rho_{ij}\sigma_i\sigma_j s_i s_j \frac{\partial^2 u(\mathbf{s},t)}{\partial s_i \partial s_j} - r u(\mathbf{s},t) = 0 \tag{1}$$

with a final condition $u(\mathbf{s}, T) = \Lambda(\mathbf{s})$, where r is the constant riskless interest rate, σ_i are volatility values of s_i, and ρ_{ij} are the asset correlations between s_i and s_j. $\Lambda(\mathbf{s})$ is the payoff function at maturity T. There are three classical techniques which are the finite difference method (FDM) [3–13], the finite element method [14], and the finite volume method [15] for the numerical solutions of the BS PDE.

Sensitivities of option price, the so-called Greeks of option values, are derivatives with respect to market variables or model parameters. In this paper, we will focus on the behaviors of the Greeks close to the maturity time and in the neighborhood of the strike price. The outline of the paper is as follows. In Section 2, we present the numerical solution of the BS PDE. In Section 3, we show the results of the several numerical experiments and the conclusions are drawn in Section 4.

2. Numerical Solution

In this section, we present a numerical scheme and its solution for the one-dimensional BS equation. By introducing $\tau = T - t$ which means the time to expiry, the one-dimensional version of (1) becomes the following initial value problem:

$$\frac{\partial u(x,\tau)}{\partial \tau} = \frac{1}{2}(\sigma x)^2 \frac{\partial^2 u(x,\tau)}{\partial x^2} + rx\frac{\partial u}{\partial x} - r u(x,\tau), \quad \text{for } (x,\tau) \in \Omega \times (0, T] \tag{2}$$

with an initial condition $u(x, 0)$ for $x \in \Omega = (0, x_{\max})$. Here, $x_{\max}$ is a sufficiently large asset price. The BS equation (2) is

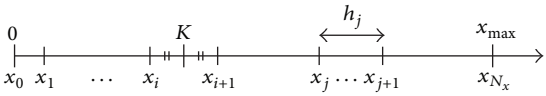

Figure 1: A nonuniform grid with spatial step sizes $h_j = x_{j+1} - x_j$.

discretized on a grid defined by $x_0 = 0$ and $x_{i+1} = x_i + h_i$ for $i = 0, \ldots, N_x - 1$, where N_x is the number of grid intervals and h_i is the grid spacing (see Figure 1). We assume that $x_{N_x} = x_{\max}$ and the ghost point $x_{N_x+1} = x_{\max} + h_{N_x-1}$.

Let $u_i^n \approx u(x_i, n\Delta\tau)$ be the numerical approximate solution, where $\Delta\tau = T/N_\tau$ is the time step size and N_τ is the total number of time steps. By applying implicit Euler's scheme to (2), we have

$$\begin{aligned}\frac{u_i^{n+1} - u_i^n}{\Delta\tau} &= \frac{\sigma^2 x_i^2}{2}\left[\frac{2u_{i-1}^{n+1}}{h_{i-1}\left(h_{i-1}+h_i\right)} - \frac{2u_i^{n+1}}{h_{i-1}h_i}\right.\\ &\quad\left. + \frac{2u_{i+1}^{n+1}}{h_i\left(h_{i-1}+h_i\right)}\right] + rx_i\left[\frac{-h_i u_{i-1}^{n+1}}{h_{i-1}\left(h_{i-1}+h_i\right)}\right.\\ &\quad\left. + \frac{\left(h_i - h_{i-1}\right)u_i^{n+1}}{h_{i-1}h_i} + \frac{h_{i-1}u_{i+1}^{n+1}}{h_i\left(h_{i-1}+h_i\right)}\right] - ru_i^{n+1},\end{aligned} \tag{3}$$

for $i = 1, \ldots, N_x$ and $n = 0, \ldots, N_\tau - 1$. Then, we can rewrite (3) as

$$\begin{aligned}&\frac{rx_ih_i - \left(\sigma x_i\right)^2}{h_{i-1}\left(h_{i-1}+h_i\right)}u_{i-1}^{n+1}\\ &\quad + \left[\frac{\left(\sigma x_i\right)^2 - rx_i\left(h_i - h_{i-1}\right)}{h_{i-1}h_i} + r + \frac{1}{\Delta\tau}\right]u_i^{n+1}\\ &\quad - \frac{rx_ih_{i-1} + \left(\sigma x_i\right)^2}{h_i\left(h_{i-1}+h_i\right)}u_{i+1}^{n+1} = \frac{u_i^n}{\Delta\tau}.\end{aligned} \tag{4}$$

In this paper, we restrict our attention to European call options. Since the option price at $x = 0$ approaches zero, we impose the zero Dirichlet boundary condition as $u(0, \tau) = 0$. Also, the option value at sufficiently large asset price is asymptotically linear. Therefore, we use the linear boundary condition [16–18] at $x = x_{\max}$ as $(\partial^2 u/\partial x^2)(x_{\max}, \tau) = 0$; that is, $u_{N_x+1}^{n+1} = 2u_{N_x}^{n+1} - u_{N_x-1}^{n+1}$. The linear system from (4) is solved by the Thomas algorithm [19].

2.1. Adaptive Time-Stepping Strategy. The numerical solutions of the BS PDE near maturity are very sensitive to the size of the time step used. In this paper, for the sake of efficiency and accuracy of the numerical solution, we consider an adaptive time-stepping strategy [20]. In this strategy, the time step is chosen by using criteria in accordance with a truncation error. Before we start, we consider the relation between the exact solution and the numerical approximation in terms of $\Delta\tau$. We denote the exact solution for an advance from τ to $\tau + 2\Delta\tau$ by $u(x, \tau + 2\Delta\tau)$ and the two approximate solutions by v (one step with $2\Delta\tau$) and w (two $\Delta\tau$ steps). In this study, since we use the fully implicit scheme for time derivative, the numerical solution of (4) has a first-order accuracy with respect to time. For the numerical solution with one step by $2\Delta\tau$,

$$\begin{aligned}u(x, \tau + 2\Delta\tau) &= u(x, \tau) + 2\Delta\tau u_\tau(x, \tau)\\ &\quad + \frac{1}{2}\left(2\Delta\tau\right)^2 u_{\tau\tau}(x, \tau) + O\left(\Delta\tau^3\right)\\ &= v(x, \tau + 2\Delta\tau) + \frac{1}{2}\left(2\Delta\tau\right)^2 u_{\tau\tau}(x, \tau)\\ &\quad + O\left(\Delta\tau^3\right).\end{aligned} \tag{5}$$

And, for the numerical solutions with two steps by $\Delta\tau$,

$$\begin{aligned}u(x, \tau + \Delta\tau) &= u(x, \tau) + \Delta\tau u_\tau(x, \tau)\\ &\quad + \frac{1}{2}\left(\Delta\tau\right)^2 u_{\tau\tau}(x, \tau) + O\left(\Delta\tau^3\right),\end{aligned} \tag{6}$$

$$\begin{aligned}u(x, \tau + 2\Delta\tau) &= u(x, \tau + \Delta\tau) + \Delta\tau u_\tau(x, \tau + \Delta\tau)\\ &\quad + \frac{1}{2}\left(\Delta\tau\right)^2 u_{\tau\tau}(x, \tau + \Delta\tau)\\ &\quad + O\left(\Delta\tau^3\right).\end{aligned} \tag{7}$$

By adding (6) and (7), we obtain

$$\begin{aligned}u(x, \tau + 2\Delta\tau) &= u(x, \tau)\\ &\quad + \Delta\tau\left[u_\tau(x, \tau) + u_\tau(x, \tau + \Delta\tau)\right]\\ &\quad + \left(\Delta\tau\right)^2\frac{u_{\tau\tau}(x, \tau) + u_{\tau\tau}(x, \tau + \Delta\tau)}{2}\\ &\quad + O\left(\Delta\tau^3\right)\\ &= w(x, \tau + 2\Delta\tau)\\ &\quad + \left(\Delta\tau\right)^2\frac{u_{\tau\tau}(x, \tau) + u_{\tau\tau}(x, \tau + \Delta\tau)}{2}\\ &\quad + O\left(\Delta\tau^3\right).\end{aligned} \tag{8}$$

Let E be the difference between the two numerical estimates; that is,

$$\begin{aligned}E &= w(x, \tau + 2\Delta\tau) - v(x, \tau + 2\Delta\tau)\\ &= \left(\Delta\tau\right)^2\phi + O\left(\Delta\tau^3\right),\end{aligned} \tag{9}$$

where ϕ means the constant value whose order of magnitude is $u_{\tau\tau}(x, \tau)$. As shown in Figure 2, there is a difference of two numerical solutions by one step with $2\Delta\tau$ and two steps with $\Delta\tau$. In general, the numerical approximation by two steps with $\Delta\tau$ is more accurate than by one step with $2\Delta\tau$.

In this study, we apply the adaptive time step strategy which is based on the local truncation error. First, we set the maximum and minimum time step sizes to avoid using too large or small time step, $\Delta\tau_{\max}$ and $\Delta\tau_{\min}$. Next, let $\Delta\tau_0$ be an initial time step size. With a given numerical solution u^n and

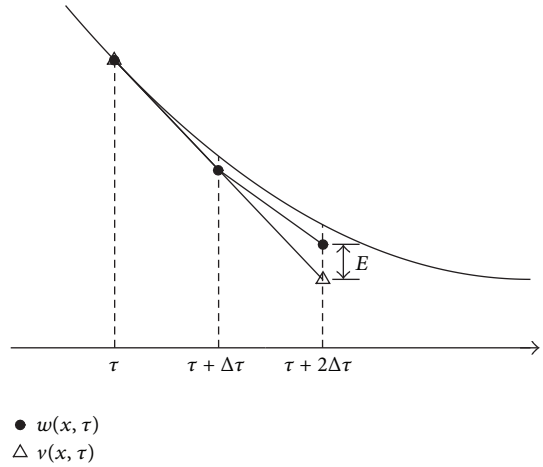

Figure 2: Schematic illustration of the numerical solutions $w(x,\tau)$ and $v(x,\tau)$ by two different time step sizes in Euler scheme. Here, $w(x,\tau)$ denotes the numerical solution by one step with $2\Delta\tau$ and $v(\tau)$ denotes the numerical solution by two steps with $\Delta\tau$.

a time step $\Delta\tau = \Delta\tau_0$, we solve (4) twice to get u^{n+2}. Next, with the given numerical solution u^n and a twice larger time step $\Delta\tau = 2\Delta\tau_0$, we solve (4) to get v^{n+2}. We define the time step scaled error as

$$E_{\text{tr}} = \left\| u^{n+2} - v^{n+2} \right\|_2, \tag{10}$$

where u^{n+2} and v^{n+2} are the numerical solution with times $\Delta\tau_0$ and $2\Delta\tau_0$, respectively. If the error is below the given tolerance, then we set the $(n+2)$th numerical solution as u^{n+2}. Otherwise, we solve (4) using each time step $\Delta\tau_0/2$ and then check the scaled error. This process repeats until the scaled error meets the given tolerance. In our strategy, the next time step size is automatically determined by the given tolerance tol and the error E_{tr} as $\Delta\tau_{\text{new}} = \Delta\tau_0 \times \text{tol}/E_{\text{tr}}$. If $E_{\text{tr}} < \text{tol}$, that is, $\text{tol}/E_{\text{tr}} > 1$, the new time step size is larger than the old one. Otherwise, which is $E_{\text{tr}} > \text{tol}$, the new one is smaller than the old one.

The adaptive time-stepping strategy can be summarized in Algorithm 1.

3. Numerical Experiments

In this section, for numerical experiments, we consider a European cash-or-nothing option which pays an amount C at maturity if option is in-the-money state. The payoff function is given by

$$u(x,0) = \begin{cases} C & \text{if } x > K, \\ 0 & \text{otherwise}, \end{cases} \tag{11}$$

where K is the strike price and C denotes the return value. The closed-form solution [21] for the option is given as

$$u(x,\tau) = Ce^{-r\tau}N(d), \quad \text{for } \forall (x,\tau) \in [0,L] \times [0,T], \tag{12}$$

where $d = [\ln(x/K) + (r - 0.5\sigma^2)\tau]/(\sigma\sqrt{\tau})$ and $N(d) = (1/\sqrt{2\pi}) \int_{-\infty}^{d} \exp(-x^2/2)\mathrm{d}x$ is the cumulative distribution function for the standard normal distribution [1]. In the Appendix, MATLAB code for the closed-form solution is presented.

Figure 3 shows the cash-or-nothing option prices at $\tau = 0$, $\tau = 1/365$, and $\tau = 2/365$ on $\Omega = [0, 300]$. Here, the option prices at $\tau = 1/365$ and $\tau = 2/365$ are obtained by (12). And we use strike price $K = 100$, cash $C = 100$, the risk-free interest rate $r = 0.03$, and volatility $\sigma = 0.3$. As shown in Figure 3, the option price has dropped drastically for the first time step. Therefore, we need to take smaller time step sizes in early times since the solution rapidly changes.

In the following sections, unless otherwise specified, we use strike price $K = 100$, cash $C = 100$, the risk-free interest rate $r = 0.03$, the volatility $\sigma = 0.3$, and $L = 250$. All computations are performed using MATLAB version 8 [22].

3.1. Convergence Test. First, we present the performance of the numerical scheme with respect to uniform spatial and temporal step sizes. For measurement of accuracy of the numerical scheme, we compute the absolute error $\mathbf{e} = |u_{\text{exact}} - u|$, where u_{exact} and u denote the exact and the numerical solutions for cash-or-nothing option, respectively. For consistent comparison of the accuracy, we evaluate the absolute error at $x = 100.5$ that is the position near the predetermined strike price $K = 100$. Figure 4 shows the spatial grid structure near K with respect to h. And the marked circle ($x = 100.5$) denotes the point where the absolute error is measured.

In this study, we are concerned with the option value and its Greeks near the expiry. Therefore, we only consider the cash-or-nothing option at the day before the option is expired; that is, $\tau = 1/365$. Table 1 represents the absolute error between the numerical and exact solutions with respect to spatial and temporal step sizes.

As shown in Table 1, the absolute error decreases as h and $\Delta\tau$ decrease. Also, we can see these results in Figure 5.

In the results, we confirm that the absolute error is more influenced by temporal step size $\Delta\tau$ than by spatial step size h. In particular, the numerical results with $h \leq 1/3$ are

Require: Set the initial conditions u^0 and v^0, the expiry time T, the maximum and minimum time steps $\Delta\tau_{\max} = T$ and $\Delta\tau_{\min} = T/86400$, tolerance tol, truncation error E_{tr}, safety factor $s = 0.8$, time $\tau = 0$, iteration number $n = 0$.

While $\tau < T$ **do**
  $n = n + 1$
  if $n = 1$ **then**
    Set $\Delta\tau_0 = \text{tol} \times \Delta\tau_{\max}$.
  else
    Set $\Delta\tau_0 = s \times \Delta\tau_0 \times \text{tol}/E_{\text{tr}}$.
  end if
  if $\tau + \Delta\tau_0 > T$ **then**
    Set $\Delta\tau_0 = T - \tau$.
    Solve (4) with time step $\Delta\tau_0$ and set its approximation as u^n.
  end if
  Solve (4) twice with time step $\Delta\tau_0$ and set its approximation as u^*.
  Solve (4) with time step $2\Delta\tau_0$ and set its approximation as v^*.
  Calculate $E_{\text{tr}} = \sqrt{\sum_{x_i \in \overline{\Omega}} (u_i^* - v_i^*)^2 / M}$ where $\overline{\Omega} = [0.9K, 1.1K]$ and M is the total number of $x_i \in \overline{\Omega}$.
  while $E_{\text{tr}} > \text{tol}$ **do**
    Set $\Delta\tau_0 = \Delta\tau_0/2$.
    if $\Delta\tau_0 < \Delta\tau_{\min}$ **then**
      Set $\Delta\tau_0 = 2\Delta\tau_0$.
    end if
    Solve (4) twice with $\Delta\tau_0$ and set its approximation as u^*.
    Solve (4) with $2\Delta\tau_0$ and set its approximation as v^*.
    Calculate $E_{\text{tr}} = \sqrt{\sum_{x_i \in \overline{\Omega}} (u_i^* - v_i^*)^2 / M}$.
  end while
  Set $\tau = \tau + \Delta\tau_0$.
  Set $u^n = u^*$ and $v^n = u^*$.
end while

Algorithm 1: Algorithm of adaptive time-stepping strategy.

Table 1: Comparison of absolute error $|u_{\text{exact}} - u|$ between the numerical and exact solutions at $x = 100.5$ and $\tau = 1/365$ with respect to h and $\Delta\tau$.

h	$\Delta\tau$					
	1/365	1/730	1/1460	1/2920	1/5840	1/11680
1	8.638266	5.403661	3.560908	2.618485	2.155975	1.929931
1/3	6.541975	3.457916	1.959400	1.288850	0.979618	0.831513
1/9	6.276309	3.262838	1.829017	1.187406	0.889680	0.746603
1/27	6.246320	3.241839	1.815166	1.176523	0.879980	0.737425
1/81	6.242982	3.239515	1.813635	1.175318	0.878906	0.736408

sufficiently accurate. Therefore, if we control time step size $\Delta\tau$, we can obtain a more accurate numerical value. In the next section, we will present several numerical results with $h = 1/3$ when we use the adaptive time-stepping strategy.

3.2. Adaptive Time-Stepping Scheme. Now, we consider an adaptive time-stepping strategy to efficiently solve the BS PDE. In this strategy, the time step size is determined at every τ^n along with the truncation error of the numerical solution as we described before.

As the default parameters, we set the initial time step size $\Delta\tau_0 = \text{tol} \times \Delta\tau_{\max}$, where $\Delta\tau_{\max} = 1/365$ and $\Delta\tau_{\min} = \Delta\tau_{\max}/(24{\cdot}3600)$. Here, $\Delta\tau_{\max} = 1/365$ and $\Delta\tau_{\min} = 1/(365{\cdot}24{\cdot}3600)$ represent one day and one second, respectively. Also, we use the spatial step size $h = 1/3$ according to results in Table 1.

Table 2 represents the root mean square error (RMSE) of the numerical solution which is measured in the area $[0.9K, 1.1K]$. Here, RMSE is calculated by $\sqrt{\sum_{i=1}^{N} (u_i - u_i^{\text{ex}})^2 / N}$, where u and u^{ex} denote the numerical and exact solutions. In Table 2, $\min \Delta\tau_0$ and $\max \Delta\tau_0$ represent the minimum and maximum time step sizes which are used during the numerical iteration. And N_τ denotes the total number of iterations during one day, $T = 1/365$. Note that the RMSE by only one time step size $\Delta\tau = 1/365$ is 2.20437414. For comparison, we evaluate the ratio of 2.20437414 to the RMSE by adaptive time-stepping strategy.

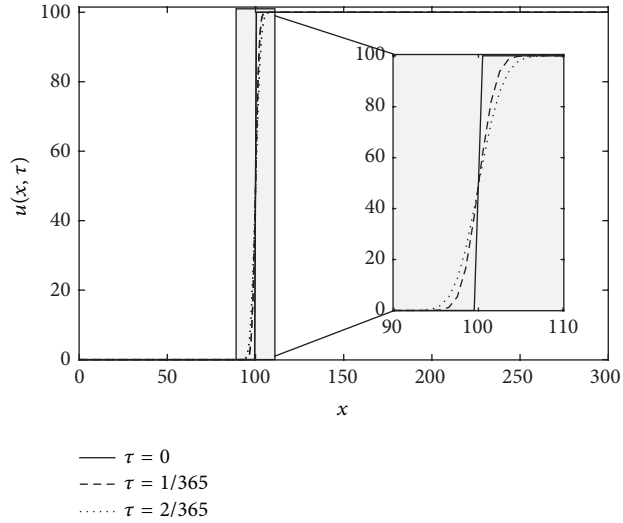

Figure 3: Values $u(x, \tau)$ of cash-or-nothing option at $\tau = 0$, $\tau = 1/365$, and $\tau = 2/365$.

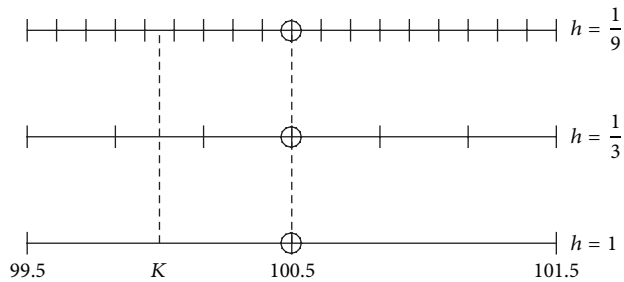

Figure 4: Construction of the spatial grid near the strike price $K = 100$ with respect to h. Here, the marked circle denotes the point where the absolute error is measured.

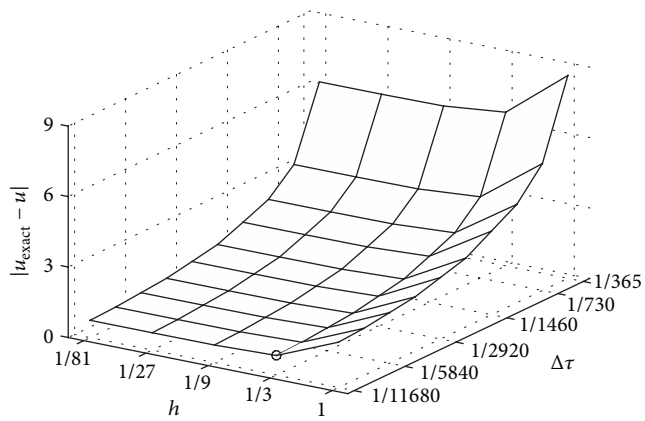

Figure 5: Absolute errors between numerical and exact solutions on $x = 100.5$ and $\tau = 1/365$ with respect to spatial and temporal step sizes.

As shown in Table 2, the numerical solution by the adaptive time-stepping method is about 40 times more accurate than that by the one time step.

Figure 6 shows the time step size $\Delta\tau$ against time τ over one day, $T = 1/365$, in adaptive time-stepping strategy with tol $= 1.0E - 4$. In Figure 6, early adaptive time step is very small because the errors between numerical and exact solutions are greatly generated during the total time $T = 1/365$.

Next, we perform numerical tests around maturity and compare the numerical solution with the exact solution. Let us define u_{ex} and u as the analytic solution by (12) and numerical solution by adaptive time strategy, respectively. Figure 7(a) shows that the usage of time step as $\Delta\tau = 1/365$ near

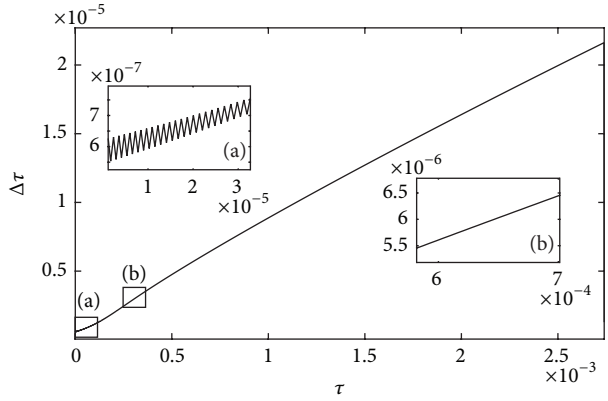

FIGURE 6: Graph of time step size $\Delta\tau$ against time τ with tolerance $1.0E-4$.

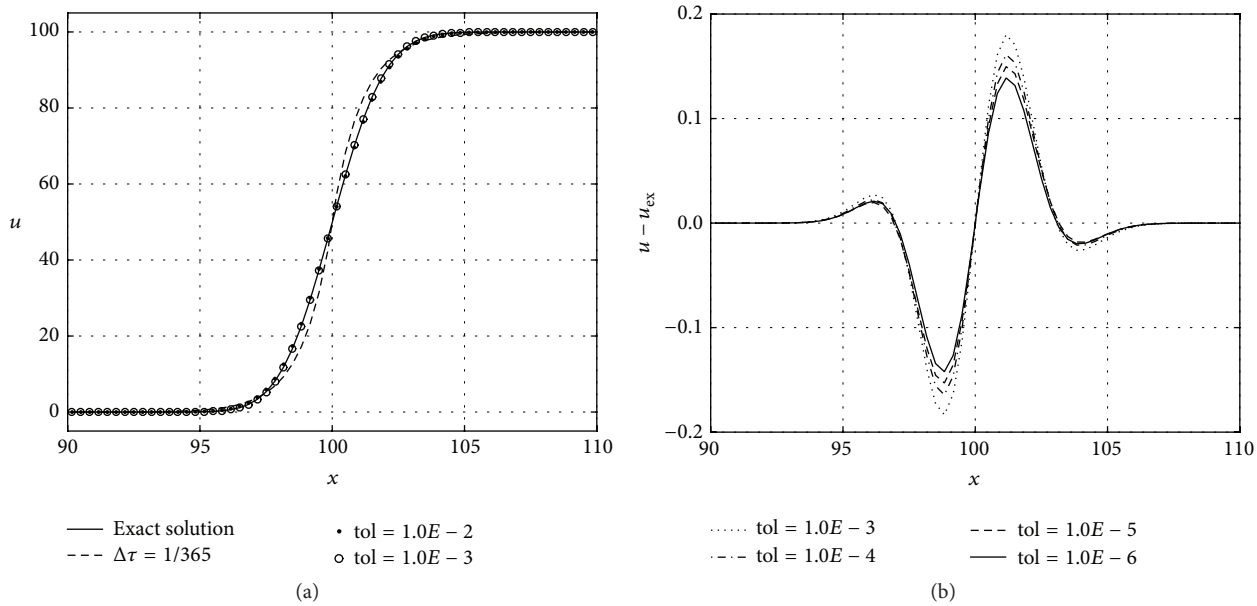

FIGURE 7: (a) Option prices $u(x, 1/365)$ as a function of the asset price x by closed-form formula, numerical approximation for only one time step size $\Delta\tau = 1/365$, and adaptive time step strategy. (b) Errors $u - u_{ex}$ as a function of the asset price x by adaptive time-stepping strategy with respect to tol.

maturity makes inaccurate solution. Figure 7(b) illustrates $u-u_{ex}$ with respect to the changes of tol.

We note that as the tol decreases, the corresponding numerical solution remains accuracy, however, for a longer time. Therefore, the tolerance, tol, can be determined by considering speed and accuracy trade-off.

3.3. The Greeks. The Greeks are defined as changes in option value relative to changes in each independent variable. For example, Delta is the first derivative of the option value with respect to the underlying asset. Gamma is the second derivative of the value with respect to the asset. Theta is the first derivative of the option with respect to time. Vega is the option's sensitivity to changes in the volatility. Rho is the first derivative of the option price with respect to interest rates. For more details about the Greeks, we refer the reader to [21].

In general, the payoff structure of the option is very stiff because it has a discontinuity around the strike price. Therefore, there exists a difficulty to numerically measure an accurate value of the Greeks. However, as we use the proposed adaptive time-stepping strategy, we can control the accuracy of the Greeks.

Next, we evaluate the option sensitivities of the cash-or-nothing option at $x = 100$. In particular, because of the given grid points as shown in Figure 1, we use the linear interpolation to calculate the Greeks at $x = 100$ in some numerical tests. For comparison, we also evaluate the Greeks by the exact formula using MATLAB code which is presented in the Appendix.

3.3.1. Delta. Delta (Δ) is defined as the rate of change of the option value with respect to small changes in the underlying

Table 2: RMSE by adaptive time stepping strategy with respect to different tolerance.

Tolerance	$\min \Delta\tau_0$	$\max \Delta\tau_0$	N_τ	RMSE	Ratio
$1.0E-2$	$5.94366E-6$	$2.04576E-4$	46	0.321825625	6.89882
$1.0E-3$	$1.28176E-6$	$4.77865E-5$	213	0.067703445	32.79326
$1.0E-4$	$5.47945E-7$	$2.1621E-5$	480	0.060855408	36.48347
$1.0E-5$	$5.47945E-8$	$4.82658E-6$	2150	0.056760465	39.11554
$1.0E-6$	$5.47945E-9$	$2.15918E-6$	4801	0.052404827	42.36664
$1.0E-7$	$5.47945E-10$	$6.82985E-7$	14214	0.054571982	40.68418
$1.0E-8$	$5.47945E-11$	$2.16019E-7$	39776	0.055265093	40.17394

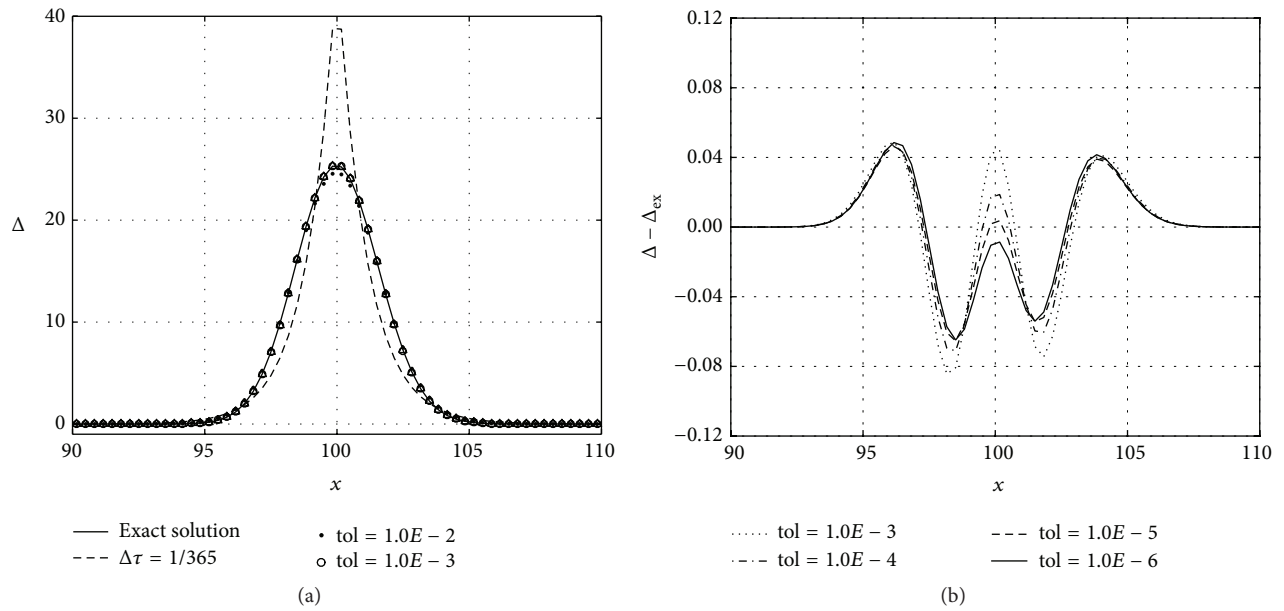

Figure 8: (a) Delta Δ at time $T = 1/365$ as a function of the asset price x by closed-form formula, numerical approximation for only one time step size $\Delta\tau = 1/365$, and adaptive time step strategy. (b) Errors $\Delta - \Delta_{ex}$ as a function of the asset price x by adaptive time-stepping strategy with respect to tol.

asset price. By differentiating (12), we have the following exact formula:

$$\Delta_{ex} = \frac{\partial u}{\partial x} = \frac{Ce^{-r\tau}N'(d)}{\sigma x\sqrt{\tau}}. \tag{13}$$

Also, we can define the Delta by using the finite difference discretizations at x_i. For example,

$$\begin{aligned}\Delta_i^n &:= \left(\frac{\partial u}{\partial x}\right)_i^n \\ &= -\frac{h_i u_{i-1}^n}{h_{i-1}(h_{i-1}+h_i)} + \frac{(h_i - h_{i-1})u_i^n}{h_{i-1}h_i} \\ &\quad + \frac{h_{i-1}u_{i+1}^n}{h_i(h_{i-1}+h_i)}.\end{aligned} \tag{14}$$

Figure 8(a) shows the Delta by closed-form formula (13), numerical approximation for only one time step size $\Delta\tau = 1/365$, and adaptive time step strategy. To represent the accuracy of the Delta with respect to tol, we investigate the error between closed-form solution and numerical approximation by our proposed method in Figure 8(b). Through these results, we can see that the adaptive time step strategy reduces the numerical error generated around the strike price $K = 100$.

Table 3 shows the Delta and its error by the adaptive time step strategy with respect to tol. As shown in Table 3, the Delta by our proposed method is more accurate as tol is smaller.

3.3.2. Gamma. Gamma (Γ) is the rate of change of the Delta with respect to changes in the underlying asset price. In other words, Gamma is defined by the second partial derivative of the portfolio value with respect to underlying asset price. According to closed-form solution (12), we get the exact formula for Gamma as

$$\Gamma_{ex} = \frac{\partial^2 u}{\partial x^2} = -Ce^{-r\tau}\frac{(d+\sigma\sqrt{\tau})N'(d)}{\sigma^2 x^2 \tau}. \tag{15}$$

Table 3: Delta Δ and its error $\Delta - \Delta_{ex}$ at $x = 100$ and $T = 1/365$. Here, Delta by the exact formula is $\Delta_{ex} = 25.40376$.

tol	$1.0E-2$	$1.0E-3$	$1.0E-4$	$1.0E-5$	$1.0E-6$	$1.0E-7$
Δ	24.7890	25.5980	25.5713	25.5553	25.5425	25.5481
$\Delta - \Delta_{ex}$	−0.61472	0.19425	0.16751	0.15149	0.13876	0.14432

Table 4: Gamma Γ and its error $\Gamma - \Gamma_{ex}$ at $x = 100$ and $T = 1/365$. Here, $\Gamma_{ex} = -0.21170$.

tol	$1.0E-2$	$1.0E-3$	$1.0E-4$	$1.0E-5$	$1.0E-6$	$1.0E-7$
Γ	−0.19796	−0.20396	−0.20378	−0.20368	−0.20359	−0.20363
$\Gamma - \Gamma_{ex}$	0.01374	0.00774	0.00792	0.00802	0.00811	0.00807

Table 5: Theta Θ and its error $\Theta - \Theta_{ex}$ at $x = 100$ and $T = 1/365$. Here, $\Theta_{ex} = 20.54956$.

tol	$1.0E-2$	$1.0E-3$	$1.0E-4$	$1.0E-5$	$1.0E-6$	$1.0E-7$
Θ	19.41235	19.96587	19.88387	19.94125	19.93973	19.93678
$\Theta - \Theta_{ex}$	−1.13721	−0.58369	−0.66570	−0.60832	−0.60984	−0.61279

As a different approach, Gamma is described by applying numerical discretization at x_i as follows:

$$\Gamma_i^n \coloneqq \left(\frac{\partial^2 u}{\partial x^2}\right)_i^n = \frac{2u_{i-1}^n}{h_{i-1}\left(h_{i-1}+h_i\right)} - \frac{2u_i^n}{h_{i-1}h_i} + \frac{2u_{i+1}^n}{h_i\left(h_{i-1}+h_i\right)}. \tag{16}$$

Figure 9 shows the Gamma at time $T = 1/365$ as a function of the asset price x by closed-form formula, numerical approximation for only one time step size $\Delta\tau = 1/365$, and adaptive time step strategy. As shown in Figure 9, while the numerical solution by one time step $\Delta\tau = 1/365$ has a great deviation from the exact formula (15), numerical approximations by adaptive time step strategy and exact solution of Gamma are in good agreement. Also, from Figure 9(b), we observe that the accuracy of the Gamma can be controlled by tol in our proposed method.

Table 4 displays the numerical value by our proposed method and its corresponding errors of the Gamma at $x = 100$. As tolerance is smaller, we can obtain a convergent value of Gamma.

3.3.3. Theta. Theta (Θ) is the rate of change of the option value with respect to changes in the time to maturity, whose exact formula on cash-or-nothing option is given as

$$\Theta_{ex} = \frac{\partial u}{\partial t} = -\frac{\partial u}{\partial \tau} = Ce^{-r\tau}\left[rN(d) + N'(d)\left(\frac{d+\sigma\sqrt{\tau}}{2\tau} - \frac{r}{\sigma\sqrt{\tau}}\right)\right]. \tag{17}$$

Also, the numerical Theta is described as

$$\Theta_i^n \coloneqq \left(\frac{\partial u}{\partial \tau}\right)_i^n = \frac{u_i^{n+1} - u_i^{n-1}}{2\Delta\tau}. \tag{18}$$

Here, we calculate the Theta by using the central difference approximation as

$$\Theta_i = \frac{u\left(x_i, T + \Delta\tau_{\text{end}}\right) - u\left(x_i, T - \Delta\tau_{\text{end}}\right)}{2\Delta\tau_{\text{end}}}, \tag{19}$$

where $\Delta\tau_{\text{end}} = 0.01T$.

In Figure 10(a), we represent Θ by each one of the methods. As a result, numerical results by the adaptive time algorithm are superior to the other ones. Also, we can see that numerical results of Θ converge as tol is smaller (see Figure 10(b) and Table 5).

3.3.4. Vega. Vega (ν) is the rate of the option value with respect to small changes in the volatility of the underlying asset. By the closed-form solution of cash-or-nothing option,

$$\nu_{ex} = \frac{\partial u}{\partial \sigma} = Ce^{-r\tau}\left(\frac{d}{\sigma} + \sqrt{\tau}\right)N'(d). \tag{20}$$

We can write the Vega by using the discretizations as

$$\nu_i^n \coloneqq \left(\frac{\partial u}{\partial \sigma}\right)_i^n = \frac{u_i^n(\sigma + \Delta\sigma) - u_i^n(\sigma - \Delta\sigma)}{2\Delta\sigma}, \tag{21}$$

where $\Delta\sigma = 0.01$.

As shown in Figure 11(a), we obtain a reasonable approximation for the Vega ν when we use the adaptive time step strategy. Moreover, the difference of ν and ν_{ex} decays as smaller tolerance level tol is determined.

The results in Table 6 show that our proposed method could be improved taking a lower tolerance parameter tol.

3.3.5. Rho. Rho (ρ) is the rate of change of the option value with respect to small changes in the riskless interest rate r. The exact Rho formula on cash-or-nothing option is derived from (12) as

$$\rho_{ex} = \frac{\partial u}{\partial r} = Ce^{-r\tau}\left(-\tau N(d) + \frac{\sqrt{\tau}}{\sigma}N'(d)\right). \tag{22}$$

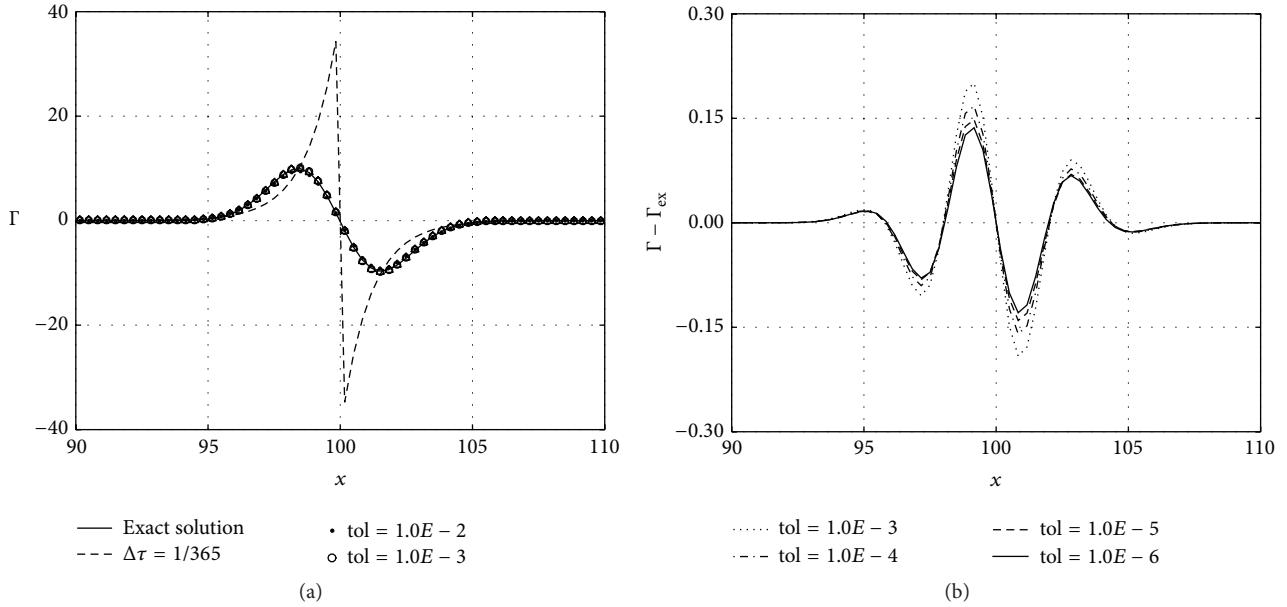

FIGURE 9: (a) Gamma Γ at time $T = 1/365$ as a function of the asset price x by closed-form formula, numerical approximation for only one time step size $\Delta\tau = 1/365$, and adaptive time step strategy. (b) Errors $\Gamma - \Gamma_{ex}$ as a function of the asset price x by adaptive time-stepping strategy with respect to tol.

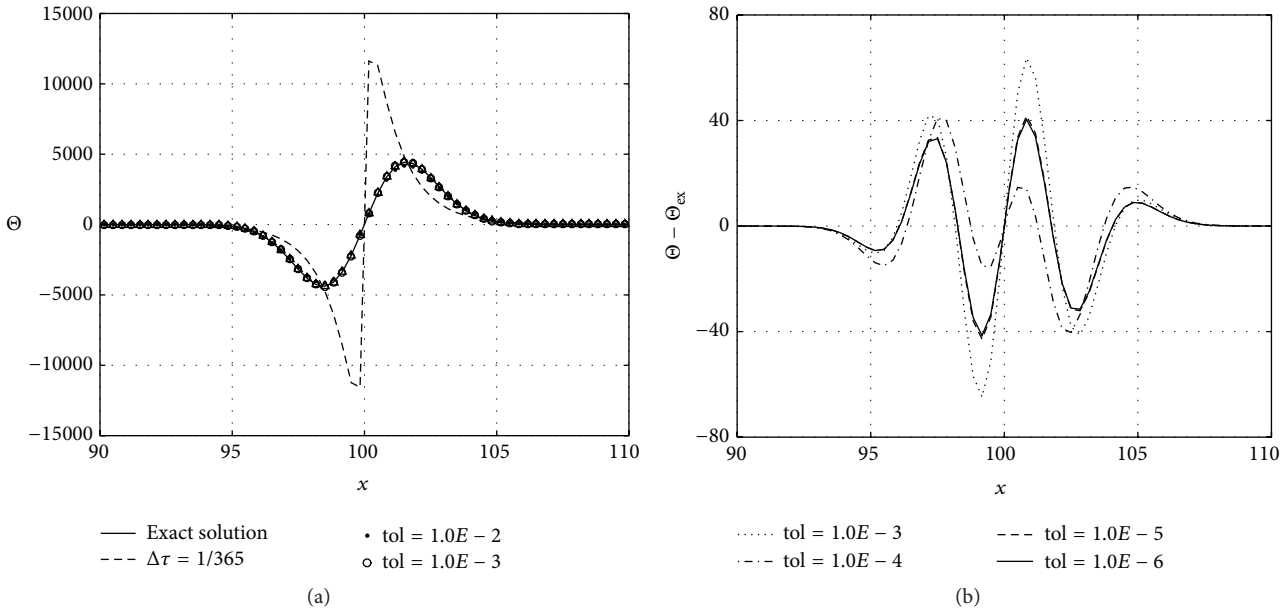

FIGURE 10: (a) Theta Θ at time $T = 1/365$ as a function of the asset price x by closed-form formula, numerical approximation for only one time step size $\Delta\tau = 1/365$, and adaptive time step strategy. (b) Errors $\Theta - \Theta_{ex}$ as a function of the asset price x by adaptive time-stepping strategy with respect to tol.

Rho is evaluated by numerical discretization as follows:

$$\rho_i^n := \left(\frac{\partial u}{\partial r}\right)_i^n = \frac{u_i^n(r+\Delta r) - u_i^n(r-\Delta r)}{2\Delta r}, \quad (23)$$

where $\Delta r = 0.001$.

Similar to previous tests, we obtain good results for Rho (ρ) by the adaptive time step algorithm as shown in Figure 12 and Table 7. Particularly, when we compare results by our proposed method with one time step simulation, we see the superiority in terms of accuracy.

3.4. Comparison with Adaptive and Uniform Time Step. In this section, we investigate the difference of numerical solutions with adaptive and uniform time step size. For comparison, we find the numerical results by uniform time step, which have similar RMSE to the results by the adaptive

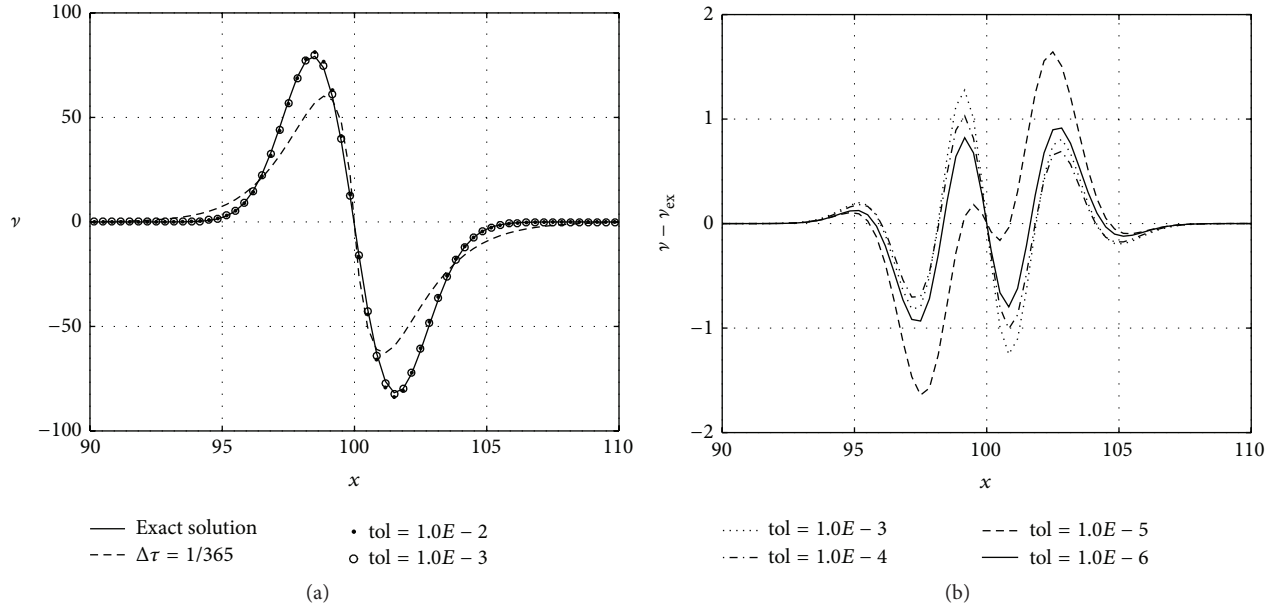

FIGURE 11: (a) Vega ν at time $T = 1/365$ as a function of the asset price x by closed-form formula, numerical approximation for only one time step size $\Delta\tau = 1/365$, and adaptive time step strategy. (b) Errors $\nu - \nu_{\text{ex}}$ as a function of the asset price x by adaptive time-stepping strategy with respect to tol.

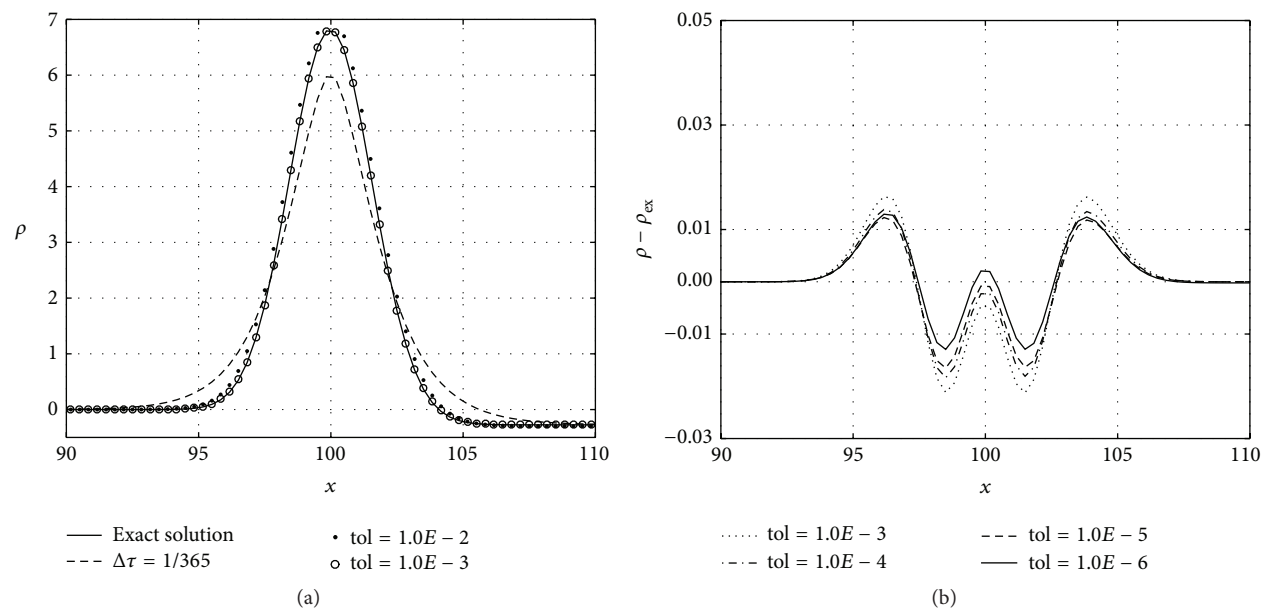

FIGURE 12: (a) Rho ρ at time $T = 1/365$ as a function of the asset price x by closed-form formula, numerical approximation for only one time step size $\Delta\tau = 1/365$, and adaptive time step strategy. (b) Errors $\rho - \rho_{\text{ex}}$ as a function of the asset price x by adaptive time-stepping strategy with respect to tol.

TABLE 6: Vega ν and its error $\nu - \nu_{\text{ex}}$ at $x = 100$ and $T = 1/365$. Here, $\nu_{\text{ex}} = -1.73998$.

tol	$1.0E-2$	$1.0E-3$	$1.0E-4$	$1.0E-5$	$1.0E-6$	$1.0E-7$
ν	−1.72129	−1.72138	−1.72165	−1.71663	−1.72054	−1.72260
$\nu - \nu_{\text{ex}}$	0.01870	0.01860	0.01833	0.02335	0.01945	0.01739

```
clear all; close all; clc; clf;
K = 100; L = 2.5*K; T = 1/365; sigma = 0.3; r = 0.03;
cash = 100; Nx = L+1; x = linspace(0,L,Nx); h = x(2)-x(1);
payoff(1:Nx) = 0.0; payoff(x >= K) = cash;
%% Price %%
d1 = (log(x/K) + (r+sigma^2/2)*T)/(sigma*sqrt(T));
d2 = d1 - (sigma*sqrt(T));
exact = cash*exp(-r*T).*normcdf(d1);
figure(1); hold on; grid on;
plot(x,payoff,'k-',x,exact,'r-'); axis([0 L -5 1.05*cash]);
%% Delta %%
del = cash*exp(-r*T).*normpdf(d1)./(sigma*sqrt(T)*x);
figure(2); hold on; grid on;
plot(del,'r-','LineWidth',1); axis([0 L -5 1.1*max(del)]);
%% Gamma %%
gam = -cash*exp(-r*T) * (d1.*normpdf(d2))./((sigma*x).^2*T);
figure(3); hold on; grid on; plot(gam,'r-','LineWidth',1);
axis([0 L -1.1*max(gam) 1.1*max(gam)]);
%% theta %%
the = cash*exp(-r*T)*(r*normcdf(d2) + normpdf(d2)...
    .* (d1/(2*T) - r/(sigma*sqrt(T))));
figure(5); hold on; grid on; plot(the,'r-','LineWidth',1);
axis([0 L -1.1*max(the) 1.1*max(the)]);
%% Vega %%
veg = -cash*exp(-r*T)*(normpdf(d2)).*d1/sigma;
figure(6); hold on; grid on; plot(veg,'r-','LineWidth',1);
axis([0 L -1.1*max(veg) 1.1*max(veg)]);
%% Rho %%
rho = cash*exp(-r*T)*((-T*normcdf(d2)) + sqrt(T)/sigma*(normpdf(d2)));
figure(4); hold on; grid on; plot(rho,'r-','LineWidth',1);
axis([0 L 3*min(rho) 1.1*max(rho)]);
```

Code 1: MATLAB code.

Table 7: Rho ρ and its error $\rho - \rho_{ex}$ at $x = 100$ and $T = 1/365$. Here, $\rho_{ex} = 6.82325$.

tol	$1.0E-2$	$1.0E-3$	$1.0E-4$	$1.0E-5$	$1.0E-6$	$1.0E-7$
ρ	7.02722	6.77944	6.78185	6.78329	6.78617	6.78449
$\rho - \rho_{ex}$	0.20394	−0.04384	−0.04143	−0.03999	−0.03711	−0.03879

time step method. Table 8 shows RMSE, N_τ, and CPU time with adaptive and uniform time step methods. As shown in Table 8, the uniform time step method is more efficient than the adaptive time step method at large tolerance (tol $\geq$ $1.0E-5$). However, we can see that when tolerance tol is below $1.0E-6$, the results by adaptive time step are better than those by uniform time step. Furthermore, we obtain more accurate results by adaptive time method than by uniform method.

4. Conclusion

In this paper, we presented the accurate numerical method for the Greeks close to the maturity time and in the neighborhood around the strike price. To reduce the difference between the numerical solution and the exact solution of the Greeks, we proposed an adaptive time step size strategy which is based on the truncation error. As a test problem, we considered a European cash-or-nothing call option. In order to show the effect of the adaptive time step, we had numerical tests for Delta, Gamma, Vega, Rho, and Theta on a variety of tolerances. Also, we compared the numerical results with the numerical solution by only one evolution having time step $\Delta\tau = T$. As a result, we obtained that the proposed method is accurate and practical in computing option price and its Greeks close to the maturity time.

Appendix

MATLAB Code

The MATLAB [22] script is composed of cash-or-nothing option value and its Greeks by the closed-form solutions of Black-Scholes equation (see Code 1).

Competing Interests

The authors declare that they have no competing interests.

Table 8: Comparison with adaptive and uniform time step methods.

	tol	RMSE	N_τ	CPU time
Adaptive	$1.0E-2$	0.32183	46	0.02790
Uniform		0.33442	9	0.00415
Adaptive	$1.0E-3$	0.06770	213	0.11714
Uniform		0.06774	207	0.02237
Adaptive	$1.0E-4$	0.06086	480	0.15395
Uniform		0.06086	476	0.04640
Adaptive	$1.0E-5$	0.05676	2150	1.04035
Uniform		0.05676	2144	0.20096
Adaptive	$1.0E-6$	0.05240	4801	1.41838
Uniform		0.05559	10000000	902.92370
Adaptive	$1.0E-7$	0.05457	14214	4.24171
Uniform		0.05559	10000000	902.92370
Adaptive	$1.0E-8$	0.05527	39776	12.89830
Uniform		0.05559	10000000	902.92370

Acknowledgments

The first author (Darae Jeong) was supported by a Korea University grant. The corresponding author (Junseok Kim) was supported by a subproject of the project Research for Applications of Mathematical Principles (no. C21501) and supported by the National Institute for Mathematical Sciences (NIMS).

References

[1] F. Black and M. Scholes, "The pricing of options and corporate liabilities," *The Journal of Political Economy*, vol. 81, no. 3, pp. 637–654, 1973.

[2] P. Wilmott, J. Dewynne, and S. Howison, *Option Pricing: Mathematical Models and Computation*, Oxford University Press, Oxford, UK, 1993.

[3] Z. Cen, A. Le, and A. Xu, "Finite difference scheme with a moving mesh for pricing Asian options," *Applied Mathematics and Computation*, vol. 219, no. 16, pp. 8667–8675, 2013.

[4] 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, no. 1, pp. 162–171, 2013.

[5] D. Jeong, J. Kim, and I.-S. Wee, "An accurate and efficient numerical method for Black-Scholes equations," *Communications of the Korean Mathematical Society*, vol. 24, no. 4, pp. 617–628, 2009.

[6] D. Jeong, I. S. Wee, and J. Kim, "An operator splitting method for pricing the ELS option," *Journal of the Korean Society for Industrial and Applied Mathematics*, vol. 14, no. 3, pp. 175–187, 2010.

[7] 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.

[8] G. González-Parra, A. J. Arenas, and B. M. Chen-Charpentier, "Positive numerical solution for a nonarbitrage liquidity model using nonstandard finite difference schemes," *Numerical Methods for Partial Differential Equations*, vol. 30, no. 1, pp. 210–221, 2014.

[9] 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.

[10] K. J. In't Hout and S. Foulon, "ADI finite difference schemes for option pricing in the Heston model with correlation," *International Journal of Numerical Analysis and Modeling*, vol. 7, no. 2, pp. 303–320, 2010.

[11] A. Q. M. Khaliq, D. A. Voss, and K. Kazmi, "Adaptive θ-methods for pricing American options," *Journal of Computational and Applied Mathematics*, vol. 222, no. 1, pp. 210–227, 2008.

[12] J. Persson and L. von Sydow, "Pricing European multi-asset options using a space-time adaptive FD-method," *Computing and Visualization in Science*, vol. 10, no. 4, pp. 173–183, 2007.

[13] A. Tagliani and M. Milev, "Laplace Transform and finite difference methods for the Black-Scholes equation," *Applied Mathematics and Computation*, vol. 220, pp. 649–658, 2013.

[14] A. Golbabai, L. V. Ballestra, and D. Ahmadian, "A highly accurate finite element method to price discrete double barrier options," *Computational Economics*, vol. 44, no. 2, pp. 153–173, 2014.

[15] R. Zvan, K. R. Vetzal, and P. A. Forsyth, "PDE methods for pricing barrier options," *Journal of Economic Dynamics & Control*, vol. 24, no. 11-12, pp. 1563–1590, 2000.

[16] D. Tavella and C. Randall, *Pricing Financial Instruments: The Finite Difference Method*, John Wiley & Sons, New York, NY, USA, 2000.

[17] 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.

[18] D. Jeong, S. Seo, H. Hwang, D. Lee, Y. Choi, and J. Kim, "Accuracy, robustness, and efficiency of the linear boundary condition for the Black-Scholes equations," *Discrete Dynamics in Nature and Society*, vol. 2015, Article ID 359028, 10 pages, 2015.

[19] W. F. Ames, *Numerical Methods for Partial Differential Equations*, Academic Press, New York, NY, USA, 2014.

[20] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in C (Vol. 2)*, Cambridge University Press, Cambridge, UK, 1996.

[21] E. G. Haug, *The Complete Guide to Option Pricing Formulas*, McGraw-Hill Companies, New York, NY, USA, 2007.

[22] The MathWorks, *MATLAB*, The MathWorks, Natick, Mass, USA, 2010, http://www.mathworks.com/.