J. KSIAM Vol.15, No.1, 1–17, 2011

A COST-EFFECTIVE MODIFICATION OF THE TRINOMIAL METHOD FOR OPTION PRICING

KYOUNG-SOOK MOON[1] AND HONGJOONG KIM[2†]

[1]DEPARTMENT OF MATHEMATICS & INFORMATION, KYUNGWON UNIVERSITY, SEONGNAM-SI 461-701, KOREA
E-mail address: ksmoon@kyungwon.ac.kr

[2]DEPARTMENT OF MATHEMATICS, KOREA UNIVERSITY, SEOUL 136-701, KOREA
E-mail address: hongjoong@korea.ac.kr

ABSTRACT. A new method for option pricing based on the trinomial tree method is introduced. The new method calculates the local average of option prices around a node at each time, instead of computing prices at each node of the trinomial tree. Local averaging has a smoothing effect to reduce oscillations of the tree method and to speed up the convergence. The option price and the hedging parameters are then obtained by the compact scheme and the Richardson extrapolation. Computational results for the valuation of European and American vanilla and barrier options show superiority of the proposed scheme to several existing tree methods.

1. INTRODUCTION

In this paper, we derive a numerical scheme to estimate the price of the option and its hedging parameters based on a tree method. Most of the option models including American types have no closed price formulas. Therefore one should rely on numerical approximations for the valuation of option prices. The usual numerical schemes for option pricing problems are the Monte Carlo simulation, the tree methods and the finite difference and element methods [1, 12, 13, 15, 16, 20].

Among these schemes, the tree method is the most popular scheme for the valuation of various derivatives in the finance community due to its ease of implementation and wide application to exotic options. After Cox et al. [7] introduced the original two-jump binomial method, many researchers improved the method in various ways. Boyle [2] studied five-jump process for two asset model and later Boyle, Evnine and Gibbs [3] and Kamrad and Ritchken [14] proposed 2^d-jump and $2^d + 1$-jump models respectively for multi-state options with $d > 1$. For American options, Broadie and Detemple [5] introduced the Binomial Black and Scholes method with Richardson extrapolation (BBSR) scheme, which includes the Black-Scholes formula in the

Received by the editors December 22, 2010; Accepted December 31, 2010.
2000 *Mathematics Subject Classification.* 91B28, 65C20, 60H35.
Key words and phrases. option pricing, trinomial method, local average.
† Corresponding author.

binomial tree method one step before expiry and applies the Richardson extrapolation. Later Gaudenzi and Pressacco [11] extended the BBSR based on the interpolation. For barrier options, Boyle and Lau [4] modified the binomial method by choosing the correct number of time steps of tree to have a layer of nodes as close as possible to the barrier. Derman et al. [8] applied linear interpolation near the barrier. Ritchken [19] proposed trinomial method which chooses the parameters so that the layers of tree pass exactly through the barrier nodes. Later Cheuk and Vorst [6] developed a trinomial tree method which uses a time-dependent shift of the tree to match the barrier. Gaudenzi and Lepellere [10] modified the binomial tree method based on the interpolation techniques.

All of these methods successfully improved the original binomial method. It has also been well known that the approximate error in various classes of tree methods tends to zero as the number of time steps goes to infinity. However, a large number of steps may be required to obtain great accuracy when the original binomial method is applied to several exotic options due to the odd-even ripples in the error. In particular, some of the methods are focusing on special types of options and do not perform well when it is applied to other types of options.

In order to compute efficiently the values of general types of options and hedging parameters, we introduce a simple modification of the trinomial method by conducting two extra steps. The first preliminary processing calculates local averages of the payoff at expiry around each node of the tree. Then the ordinary backward process is performed with these averaged payoff. Finally we apply the compact scheme and the Richardson extrapolation as a post-processing. Note that the computational effort due to these pre- and post-processing is negligible compared to the original backward process. We carry out a computational study to compare many recent tree methods for the valuation of various types of options and hedging parameters including European and American vanilla and barrier options. These numerical experiments show that the proposed method is more accurate and efficient than other tree methods, see the detailed comparison in Section 4.

The outline of the paper is as follows. In Section 2 we first set up the problem to price European and American options and explain the original trinomial method. In Section 3 we explain new trinomial tree method based on the local averages (termed TTLA). In Section 4 we show the numerical experiments and the conclusions are drawn in Section 5. Brief description of the compact schemes is given in Appendix.

2. TRINOMIAL TREE METHOD

Let us consider the price of the underlying asset as a stochastic process $\{S_t\}_{t\in[0,T]}$ on a suitable probability space $(\Omega, \mathcal{F}, \mathbb{P})$. Let us assume that the evolution of the underlying asset satisfies the following stochastic differential equation:

$$\mathrm{d}S(t) = \mu S \mathrm{d}t + \sigma S \mathrm{d}W(t), \qquad 0 < t < T,$$

where the expected rate of return μ and the volatility σ are constant. T is the expiration date and $W(t)$ is a Wiener process. From the Ito formula [9, 17], the process $X(t) \equiv \ln(S(t)) - \left(\mu - \sigma^2/2\right) t$ satisfies

$$X(t) = X(0) + \sigma W(t), \qquad t > 0.$$

In the risk-neutral world, the value of the European option V^{exact} can be obtained by

$$V^{\text{exact}}(x,t) = e^{-r(T-t)} E\left[\Lambda(X(T)) | X(t) = x\right], \tag{2.1}$$

where r is the risk-free interest rate and $\Lambda(X(t))$ represents the payoff of the option at t. Without loss of generality, we denote again $X(t)$ by the risk neutral process with drift rate r. If we consider a continuous dividend yield, q, the drift rate becomes $r - q$.

Let us partition the interval $[0, T]$ into N bins of uniform length $\Delta t = T/N$, with endpoints $0 = t_0 < t_1 < ... < t_N = T$. The trinomial tree method [2] assumes that the asset price $S(t_n)$ at $t = t_n$ moves either up to $uS(t_n)$ for $u > 1$ or to $S(t_n)$ or down to $dS(t_n)$ for $d < 1$ with probabilities p_u, p_m, p_d, respectively, for $n = 0, 1, \ldots, N-1$. Various types of trinomial methods have different definitions for the probabilities or parameters such as u and d [2, 19, 6, 11, 10]. Let $X_j^n = X_0 + jh$ denote the value of the underlying asset after the change of variables at $t = t_n$ for $j = -n, \ldots, n$ and $V_j^n = V(X_j^n, t_n)$ denote the numerical approximation of $V^{\text{exact}}(x,t)$ in (2.1) at $(x,t) = (X_j^n, t_n)$. Then the algorithm of the trinomial method for the valuation of the European options is summarized in Algorithm 1.

Algorithm 1 Trinomial Method

for $j = -N$ to N **do**
  Compute the option prices at expiry, $V_j^N = \Lambda(X_j^N)$,
end for
for $n = N-1$ to 0 **do**
  for $j = -n$ to n **do**
    Compute the backward iteration for V_j^n,

$$V_j^n = e^{-r\Delta t}(p_u V_{j+1}^{n+1} + p_m V_j^{n+1} + p_d V_{j-1}^{n+1}) , \tag{2.2}$$

  end for
end for
Estimate the option price and the hedging parameters

The value of the American option, V^{Amer}, can be computed by

$$V^{\text{Amer}}(x,t) = \max_{\tau \in [t,T]} e^{-r(\tau - t)} E\left[\Lambda(X(\tau)) | X(t) = x\right],$$

where τ is an optimal exercise time. The value of the American option can be easily approximated using trinomial tree method. We can simply change the backward procedure (2.2) in Algorithm 1 by

$$V_j^n = \max\left\{e^{-r\Delta t}(p_u V_{j+1}^{n+1} + p_m V_j^{n+1} + p_d V_{j-1}^{n+1}), \Lambda(X_j^n)\right\} . \tag{2.3}$$

3. Modified trinomial tree method

When a tree method is applied for option pricing, since the values of parameters such as strike or barrier prices are estimated by the nearby nodes in the tree, there are representation errors due to overestimation or underestimation, see [16]. For example, there is a discrepancy between the exact value and its approximation when a strike price K is represented by a point A in Figure 1. In addition, when a barrier is changed from H_1 to H_2 as in Figure 1, the theoretical option price changes but the value from the tree method may not be changed. These errors decrease as the number of nodes of the tree increases. But this number cannot be arbitrarily large in practice.

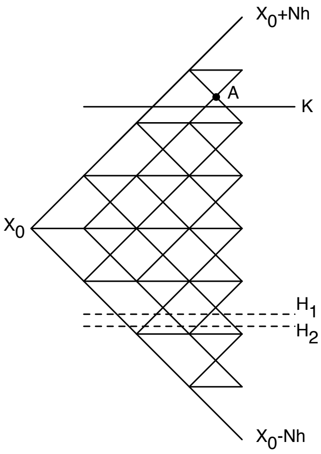

FIGURE 1. Trinomial tree method.

This study suggests an alternative efficient trinomial tree method to reduce the representation errors. We introduce local averages of option values and these averages smooth out the representation errors. Figure 2 shows an example plot of the approximation error $|V^{\text{exact}}(X_0, 0) - V(X_0, 0)|$ in the option prices from the standard trinomial method and the proposed method using local average as the number of steps of the tree increases. The error from the proposed scheme decreases much faster than the original method. The European vanilla call option with parameters in Section 4.2 is considered and the trinomial method by Ritchken [19] is used for the standard method. More in-depth experimental results are shown in Section 4.

Now let us explain in detail the algorithm of trinomial tree method using local averages (termed TTLA) for European vanilla option in Section 3.1. The modification for American option is presented in Section 3.2.

3.1. **Trinomial tree method using local averages (TTLA).** The standard trinomial method in Section 2 considers the option prices at points $X_0 + jh,\ j = -N, -(N-1), \ldots, N$ of the tree at expiry, and estimates the option price $V(x, t_n)$ from the prices at $t_n + \Delta t$ by

$$V(x, t_n) = e^{-r\Delta t}(p_u\, V(x+h, t_n+\Delta t) + p_m\, V(x, t_n+\Delta t) + p_d\, V(x-h, t_n+\Delta t))\ ,$$

as in Figure 3 (Left).

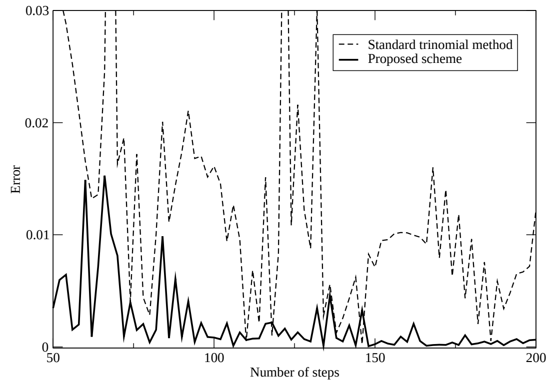

FIGURE 2. Errors of the option price from the standard trinomial method (dashed line) and the proposed TTLA scheme (solid line).

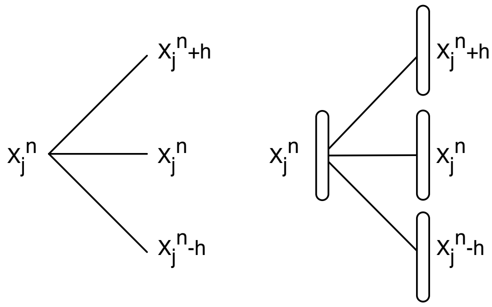

FIGURE 3. (Left) Standard trinomial method and (Right) the proposed trinomial method using local averages.

Let us first consider the interval $[X_0 - (N+1/2)h,\ X_0 + (N+1/2)h]$ and partition it into $2N+1$ intervals of uniform length h,

$$[X_0 - (N+1/2)h,\ X_0 - (N-1/2)h], \cdots, [X_0 + (N-1/2)h,\ X_0 + (N+1/2)h].$$

Then let us estimate the averages, $\overline{V}_j^N$, of option prices on each interval $[X_0+(j-1/2)h,\ X_0+(j+1/2)h]$ defined by

$$\overline{V}_j^N = \frac{1}{h}\int_{X_{j-1/2}^N}^{X_{j+1/2}^N} \Lambda(\xi)\mathrm{d}\xi, \qquad j = -N, \ldots, N, \tag{3.1}$$

where $X_j^N = X_0 + jh$ at expiry. From the definition of the local averages in (3.1) at expiry, we can derive the following property (See Figure 3 (Right)):

Property 1. *Assume that for a fixed time t_n the following relation holds at each point $\xi \in [X_j^n - h/2,\ X_j^n + h/2]$*

$$V(\xi, t_n) = e^{-r\Delta t}(p_u\, V(\xi + h, t_n + \Delta t) + p_m\, V(\xi, t_n + \Delta t) + p_d\, V(\xi - h, t_n + \Delta t)).$$

Then the average price $\overline{V}_j^n$ for the European vanilla option satisfies

$$\overline{V}_j^n = e^{-r\Delta t}\left(p_u\, \overline{V}_{j+1}^{n+1} + p_m\, \overline{V}_j^{n+1} + p_d\, \overline{V}_{j-1}^{n+1}\right), \tag{3.2}$$

for $j = -n, -n+1, ..., n$.

Therefore from the backward procedure (3.2) we finally get the average of the option price $\overline{V}_0^0$ at $t = 0$. The value $\overline{V}_0^0$ itself is an approximation of the option price $V^{\mathrm{exact}}(X_0, 0)$, however we can further reduce the approximation error $|V^{\mathrm{exact}}(X_0, 0) - \overline{V}_0^0|$ using the following compact scheme for the option price, $V(X_0, 0)$, and the delta, $\Delta(X_0, 0) = V'(X_0, 0)$. The detailed derivation of the formula is proved in Appendix.

$$\begin{aligned} V(X_0, 0) &= -\frac{1}{24}\overline{V}_{-1}^0 + \frac{13}{12}\overline{V}_0^0 - \frac{1}{24}\overline{V}_1^0\,, \\ \Delta(X_0, 0) &= \frac{\overline{V}_1^0 - \overline{V}_{-1}^0}{4h}, \end{aligned} \tag{3.3}$$

where the values $\overline{V}_{-1}^0 = \overline{V}(X_0 - h, 0)$ and $\overline{V}_1^0 = \overline{V}(X_0 + h, 0)$ are computed during the backward iteration. Since only two more points are needed to compute the formula (3.3) at each time step, the amount of extra work is negligible.

Finally in order to accelerate the rate of convergence we apply the Richardson extrapolation as a post-processing based on N and $N/2$ number of nodes after the option price and delta are obtained from the compact scheme (3.3). Note that N is chosen to be an even number for the extrapolation. Taking local average has a smoothing effect and this seems to be a partial reason for a positive result on the extrapolation. In addition, since the integral operator is continuous, the TTLA will converge at the same rate as the trinomial method [19]. More theoretical study will be performed. Consequently, the algorithm of the TTLA for the European vanilla option can be summarized as follows.

3.2. **American option.** The American option allows early exercise of the option and we need appropriate modification of the algorithm. If $V(x, t_n)$ denotes the result from the backward process of the tree method at time t_n, the American option price $V^{\mathrm{Amer}}(x, t_n)$ needs the following comparison step as in (2.3),

$$V^{\mathrm{Amer}}(x, t_n) = \max\{V(x, t_n), \Lambda(x, t_n)\}\,.$$

Note that for American put options there exists an optimal exercise boundary $S^* = S^*(t_n)$ such that

$$\max\{V(x, t_n), \Lambda(x, t_n)\} = \begin{cases} V(x, t_n) & \text{if } S^* \le x \\ \Lambda(x, t_n) & \text{if } x \le S^* \end{cases}.$$

Algorithm 2 Trinomial Method using Local Averages (TTLA)

for $j = -N-1$ to $N+1$ **do**
  Compute the local averages (3.1) of the option prices at expiry, $\overline{V}_j^N$,
end for
for $n = N-1$ to 0 **do**
  for $j = -n-1$ to $n+1$ **do**
    Compute the backward iteration (3.2) for $\overline{V}_j^n$,
  end for
end for
Estimate the option price and the hedging parameter delta (3.3)
Perform Richardson extrapolation

Thus, we have

$$\int_{X_j^n-h/2}^{X_j^n+h/2} \max\{V(\xi,t_n),\Lambda(\xi,t_n)\}\mathrm{d}\xi$$

$$= \begin{cases} \int_{X_j^n-h/2}^{X_j^n+h/2} V(\xi,t_n)\mathrm{d}\xi & \text{if } S^* \le X_j^n - h/2 \\ \int_{X_j^n-h/2}^{X_j^n+h/2} \Lambda(\xi,t_n)\mathrm{d}\xi & \text{if } X_j^n + h/2 \le S^* \\ \int_{X_j^n-h/2}^{S^*} \Lambda(\xi,t_n)d\xi + \int_{S^*}^{X_j^n+h/2} V(\xi,t_n)\mathrm{d}\xi & \text{if } X_j^n - h/2 < S^* < X_j^n + h/2 \end{cases},$$

$$= \begin{cases} \max\{\int_{X_j^n-h/2}^{X_j^n+h/2} V(\xi,t_n)\mathrm{d}\xi, \int_{X_j^n-h/2}^{X_j^n+h/2} \Lambda(\xi,t_n)\mathrm{d}\xi\} & \text{if } S^* \le X_j^n - h/2 \text{ or } X_j^n + h/2 \le S^* \\ \int_{X_j^n-h/2}^{S^*} \Lambda(\xi,t_n)\mathrm{d}\xi + \int_{S^*}^{X_j^n+h/2} V(\xi,t_n)\mathrm{d}\xi & \text{if } X_j^n - h/2 < S^* < X_j^n + h/2 \end{cases}.$$

That is, out of $2N+1$ intervals only one interval containing S^* gives a different value. If S^* is contained in some interval $[X_j^n - h/2,\ X_j^n + h/2]$, the error between $\int_{X_j^n-h/2}^{S^*} \Lambda(\xi,t_n)\mathrm{d}\xi + \int_{S^*}^{X_j^n+h/2} V(\xi,t_n)\mathrm{d}\xi$ and $\max\{\int_{X_j^n-h/2}^{X_j^n+h/2} \Lambda(\xi,t_n)\mathrm{d}\xi,\ \int_{X_j^n-h/2}^{X_j^n+h/2} V(\xi,t_n)\mathrm{d}\xi\}$ on this single interval is small when h is small, and converges to 0 as h decreases to 0. Let $\overline{\Lambda}_j^n = \frac{1}{h}\int_{X_j^n-h/2}^{X_j^n+h/2} \Lambda(\xi,t_n)\ \mathrm{d}\xi$. Then we have the following TTLA algorithm for American options. Algorithm 3 performs an additional update procedure after each backward iteration in Algorithm 2.

4. NUMERICAL COMPUTATIONS

We compare the proposed TTLA method with three trinomial methods by Boyle [2], Ritchken [19], and Cheuk and Vosrt [6] when these schemes are applied to European and American vanilla and barrier options. The comparisons with BIR [10] and BBSR [11] are added for the American options.

Algorithm 3 TTLA for American options

for $j = -N-1$ to $N+1$ **do**
  Compute the local averages (3.1) of the option prices at expiry, $\overline{V}_j^N$,
end for
for $n = N-1$ to 0 **do**
  for $j = -n-1$ to $n+1$ **do**
    Compute the backward iteration (3.2) for $\overline{V}_j^n$,
    Update $\overline{V}_j^n = \max\{\overline{V}_j^n, \overline{\Lambda}_j^n\}$,
  end for
end for
Compute the option price and the hedging parameter delta (3.3)
Perform Richardson extrapolation

4.1. **Sample selection.** A sample of 5000 option parameters was generated for valuation according to the following distributions introduced by Broadie and Detemple [5]: risk free interest rate r is uniformly distributed between 0 and 0.1; volatility σ is uniformly distributed between 0.1 and 0.6; strike price K is uniformly distributed between 70 and 130; time to maturity (years) T is uniformly distributed between 0.1 and 1 with probability 0.75 and uniformly distributed between 1 and 5 with probability 0.25; The initial price of the underlying S_0 is 100. In case of the barrier options, we consider the down-and-out European barrier call option whose barrier H is uniformly distributed between 55 and 85, and the up-and-out American barrier put option whose barrier H is uniformly distributed between 115 and 145. The rebate R for both barrier options is $R = 0$ and the dividend yield $q = 0$ for all cases. The exact formula in [12, 18] is used for the European options. For the American options, the benchmark price and hedging parameter delta are computed by the binomial model with $20,000$ steps.

The error measure is the root mean squared relative error (RMSRE),

$$\text{RMSRE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\frac{V_i - V_i^{\text{exact}}}{V_i^{\text{exact}}}\right)^2},$$

where V_i^{exact} is the true value, and V_i is the computed value. The summation is taken over options satisfying $V_i \geq 0.5$ as in [5], and $H \leq K$ for European down-and-out barrier options ($K \leq H$ for American up-and-out barrier options) as in [10]. Out of $5,000$ options, $4,494$ European vanilla options, $4,474$ European barrier options, $4,557$ American vanilla options and $4,225$ American barrier options satisfied the criterion. The computations were performed on Imac computer with 2.4 GHz Intel Core 2 Duo processor.

4.2. **European Vanilla option.** Figure 4 shows the RMSREs of the price vs. the CPU time (Top) and the number of time steps (Bottom) for the European vanilla call option. For a fixed computational cost (i.e. for a fixed CPU time or number of steps), the TTLA gives smaller

RMSRE than the others. In other words, for a fixed RMSRE, the TTLA takes the smallest computational time. Figure 4 (Below) also shows that the convergence of the TTLA is faster.

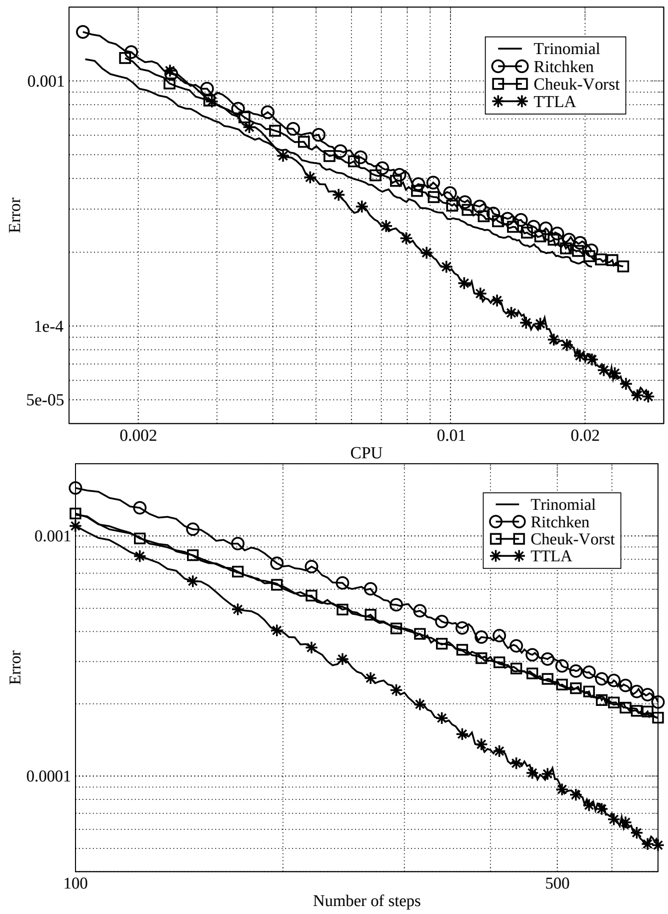

FIGURE 4. The RMSRE of the price vs. the CPU time (Top) and the number of time steps (Bottom) for the European vanilla call option.

Table 1 and Table 2 show the RMSREs of the option price and delta, respectively, from four different methods for the European vanilla call option with respect to the number of time steps of the tree. The numbers in the parenthesis are the computational CPU time measured in seconds. When the same number of steps are used, TTLA results in smallest errors for both option price and delta.

TABLE 1. The RMSRE of the option price for the European vanilla option as the number of steps increases. The numbers in the parenthesis are the CPU time in seconds.

Number of steps	Boyle	Ritchken	Cheuk-Vorst	TTLA
100	0.00123(0.00152)	0.00158(0.00150)	0.00124(0.00187)	0.00110(0.00235)
200	0.00060(0.00345)	0.00075(0.00343)	0.00062(0.00415)	0.00040(0.00498)
300	0.00041(0.00588)	0.00052(0.00585)	0.00041(0.00703)	0.00022(0.00823)
400	0.00030(0.00880)	0.00038(0.00878)	0.00031(0.01045)	0.00013(0.01214)
500	0.00025(0.01223)	0.00030(0.01219)	0.00025(0.01448)	0.00010(0.01663)
600	0.00020(0.01618)	0.00024(0.01617)	0.00020(0.01909)	0.00007(0.02180)
700	0.00017(0.02073)	0.00020(0.02066)	0.00018(0.02434)	0.00005(0.02769)

TABLE 2. The RMSRE of the delta for the European vanilla option as the number of steps increases. The numbers in the parenthesis are the CPU time in seconds.

Number of steps	Boyle	Ritchken	Cheuk-Vorst	TTLA
100	0.00147(0.00152)	0.00204(0.00150)	0.02542(0.00187)	0.00046(0.00235)
200	0.00073(0.00345)	0.00101(0.00343)	0.01822(0.00415)	0.00017(0.00498)
300	0.00049(0.00588)	0.00069(0.00585)	0.01477(0.00703)	0.00010(0.00823)
400	0.00037(0.00880)	0.00051(0.00878)	0.01283(0.01045)	0.00005(0.01214)
500	0.00030(0.01223)	0.00041(0.01219)	0.01162(0.01448)	0.00004(0.01663)
600	0.00024(0.01618)	0.00034(0.01617)	0.01061(0.01909)	0.00003(0.02180)
700	0.00021(0.02073)	0.00029(0.02066)	0.00971(0.02434)	0.00002(0.02769)

4.3. **European Barrier option.** Figure 5 shows the RMSREs of the price vs. the CPU time (Top) and the number of time steps (Bottom) for the European down-and-out barrier call option. The figure shows that when the number of nodes is small the method by Cheuk and Vorst gives better option price estimation than the other schemes. But as the number of nodes increases, the graph of TTLA is lower than the others, and TTLA results in higher degree of precision in option pricing than Cheuk and Vorst's method or others. In other words, when a sufficiently small error tolerance is considered, the TTLA takes the smallest computational time and hence

TABLE 3. The RMSRE of the option price for the European down-and-out barrier call option as the number of steps increases. The numbers in the parenthesis are the CPU time in seconds.

Number of steps	Boyle	Ritchken	Cheuk-Vorst	TTLA
100	0.04778(0.00204)	0.04111(0.00206)	0.02643(0.00240)	0.05262(0.00479)
200	0.03349(0.00478)	0.03472(0.00479)	0.01830(0.00551)	0.02105(0.01006)
300	0.03057(0.00834)	0.02881(0.00834)	0.01783(0.00948)	0.01555(0.01641)
400	0.02505(0.01258)	0.02445(0.01259)	0.01719(0.01427)	0.01131(0.02370)
500	0.02189(0.01766)	0.02221(0.01766)	0.01496(0.01999)	0.00991(0.03215)
600	0.02142(0.02350)	0.02007(0.02346)	0.01386(0.02647)	0.00895(0.04138)
700	0.02015(0.03020)	0.01786(0.03015)	0.01246(0.03388)	0.00820(0.05170)

is the most cost-effective. This numerical result shows that the convergence rate of TTLA is faster than the other schemes as in Figure 4.

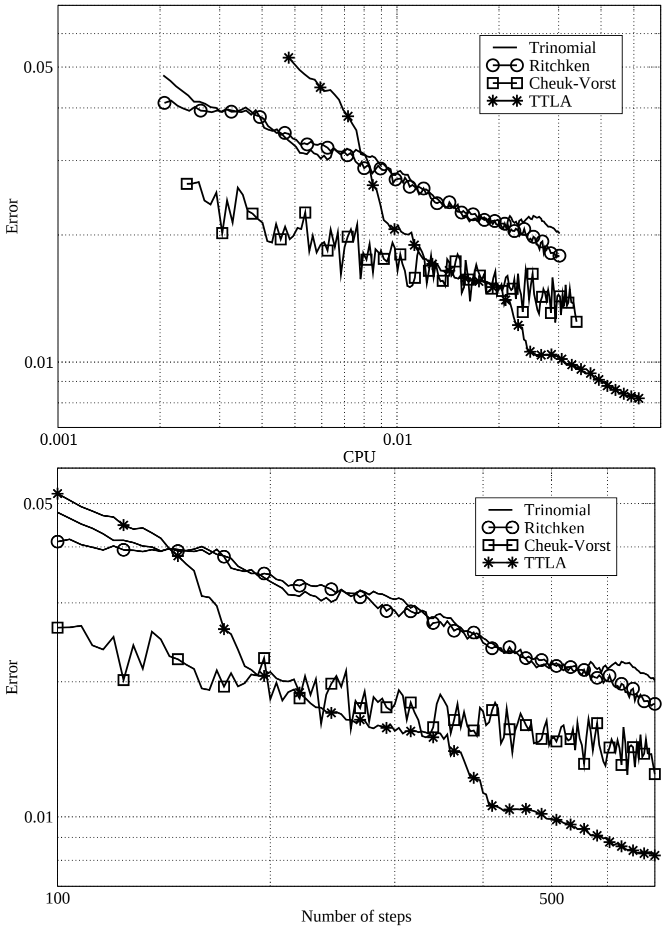

FIGURE 5. The RMSRE of the price and vs. the CPU time (Top) and the number of time steps (Bottom) for the European down-and-out barrier call option.

Table 3 and Table 4 show the RMSREs of the option price and delta, respectively, from four different methods for the European down-and-out barrier option. TTLA results in the smallest errors for both option price and delta as in Table 1 and Table 2.

TABLE 4. The RMSRE of the delta for the European down-and-out barrier call option as the number of steps increases. The numbers in the parenthesis are the CPU time in seconds.

Number of steps	Boyle	Ritchken	Cheuk-Vorst	TTLA
100	0.02148(0.00204)	0.01971(0.00206)	0.02775(0.00240)	0.01216(0.00479)
200	0.01545(0.00478)	0.01533(0.00479)	0.01992(0.00551)	0.00731(0.01006)
300	0.01290(0.00834)	0.01256(0.00834)	0.01707(0.00948)	0.00564(0.01641)
400	0.01118(0.01258)	0.01079(0.01259)	0.01446(0.01427)	0.00486(0.02370)
500	0.00989(0.01766)	0.00987(0.01766)	0.01356(0.01999)	0.00431(0.03215)
600	0.00913(0.02350)	0.00902(0.02346)	0.01247(0.02647)	0.00394(0.04138)
700	0.00865(0.03020)	0.00835(0.03015)	0.01184(0.03388)	0.00362(0.05170)

4.4. **American Vanilla option.** Figure 6 shows the RMSREs of the option price and delta for the American vanilla put option with respect to the CPU time. It can be seen that in case of the American vanilla option, the Ritchken and BBSR methods give very good results in terms of the CPU time for option price and delta. Figure 7 shows the RMSRE of the price in terms of the number of time steps, which shows that given a number of time steps, BBSR and TTLA methods result in better performance than the others and they show almost identical errors. Note that slopes of the graph from the TTLA method are slightly steeper than the others. Thus when the number of nodes is sufficiently large, the TTLA method is expected to give at least comparable results to those from the Ritchken or BBSR methods.

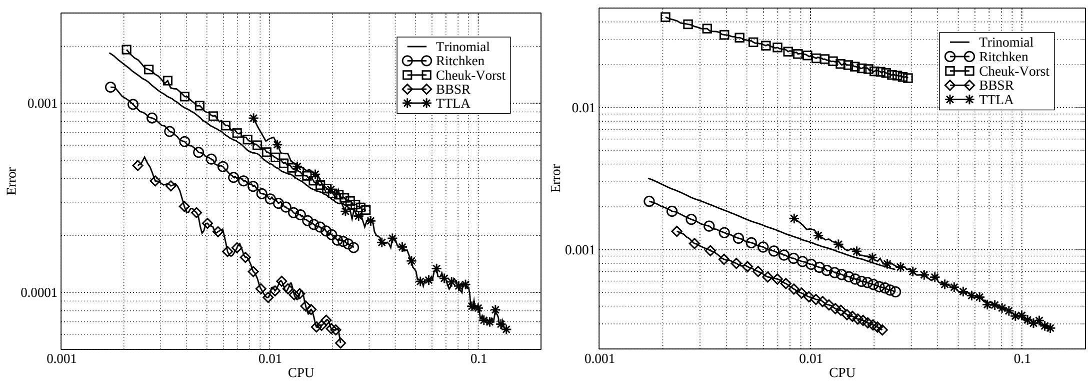

FIGURE 6. The RMSREs of (Left) the price and (Right) the delta vs. the CPU time for the American vanilla put option.

4.5. **American Barrier option.** Figure 8 shows the RMSREs of the option price and delta vs the CPU time for the American up-and-out barrier put option and Figure 9 shows the RMSRE of the price vs the number of time steps. When the American up-and-out barrier put option is considered, the Ritchken and Boyle methods result in good accuracy when the number of nodes

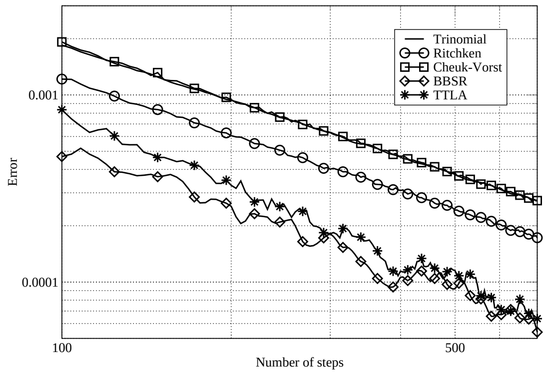

FIGURE 7. The RMSREs of the price vs. the number of time steps for the American vanilla put option.

is small. As the number increases, the result from the TTLA method shows better precision than the others for both option price and delta. The convergence rate for TTLA is faster than the others.

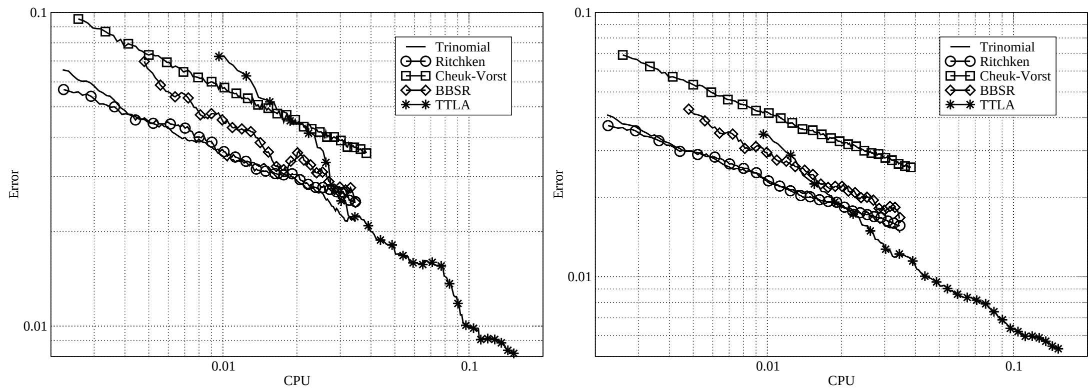

FIGURE 8. The RMSREs of (Left) the price and (Right) the delta vs. the CPU time for the American up-and-out barrier put option.

5. CONCLUSIONS

We modified the standard trinomial tree method based on local averages of the option price and compared its performance with other tree schemes in terms of their accuracy and efficiency. Given an error tolerance, the computational time for the proposed TTLA scheme is smaller than the others. That is, the TTLA method is more cost-effective than the other schemes. In particular, TTLA works very well for general types of options including European and American

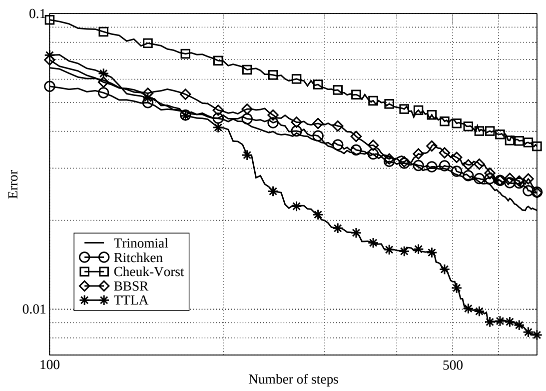

FIGURE 9. The RMSREs of the price vs. the number of time steps for the American up-and-out barrier put option.

vanilla and barrier options. Taking local averages has a smoothing effect to reduce oscillations of the tree method and it seems to accelerate the convergence rate. We could not prove this claim theoretically but computational experiments for several types of options numerically support this.

6. APPENDIX

6.1. **Compact schemes.** Suppose that a function $f(x)$ is continuously differentiable on an interval $I = [m, M]$ and $m < x_0 - 3h < x_0 - 2h < \cdots < x_0 + 3h < M$ for some $x_0 \in I$ and $h > 0$. Let $f_i = f(x_0 + ih), \quad i = -3, \ldots, 3$ and let us consider a following equation for $f(x)$ and $f'(x)$, where α, β, a, b, c are constants.

$$f'_0 + \alpha\left(f'_1 + f'_{-1}\right) + \beta\left(f'_2 + f'_{-2}\right) = a\,\frac{f_1 - f_{-1}}{2h} + b\,\frac{f_2 - f_{-2}}{4h} + c\,\frac{f_3 - f_{-3}}{6h}\ .$$

If $f(x)$ is sufficiently smooth, from the Taylor series expansion,

$$(LHS) = (1 + 2(\alpha + \beta))\,f'_0 + 2(\alpha + 4\beta)\frac{h^2}{2}f'''_0 + 2(\alpha + 16\beta)\frac{h^4}{4!}f_0^{(5)} + 2(\alpha + 64\beta)\frac{h^6}{6!}f_0^{(7)} + \cdots\ ,$$

and

$$(RHS) = (a + b + c)f'_0 + (a + 4b + 9c)\frac{h^2}{3!}f'''_0 + (a + 16b + 81c)\frac{h^4}{5!}f_0^{(5)} + (a + 64b + 729c)\frac{h^6}{7!}f_0^{(7)} + \cdots\ .$$

In particular, in order to have the 4th order accuracy, following two equations should be satisfied

$$a + b + c = 1 + 2\alpha + 2\beta\ ,$$
$$a + 4b + 9c = 6\alpha + 24\beta\ .$$

That is, we have

$$a = \frac{1}{3}(5c + 2\alpha - 16\beta + 4) ,$$
$$b = \frac{1}{3}(-8c + 4\alpha + 22\beta - 1) .$$

For example, if α, β and b are set to 0, $a = 9/8$ and $c = -1/8$ and the equation

$$f'_0 = \frac{9}{8}\,\frac{f_1 - f_{-1}}{2h} - \frac{1}{8}\,\frac{f_3 - f_{-3}}{6h} , \tag{6.1}$$

becomes 4th order accurate.

Lemma 2. *The compact scheme (6.1) gives the following approximation for the option price*

$$V(X_0, 0) = -\frac{1}{24}\overline{V}^0_{-1} + \frac{13}{12}\overline{V}^0_0 - \frac{1}{24}\overline{V}^0_1 .$$

Proof. Let $f(x) \equiv \int_{X_0-3h/2}^{x} V(\xi, 0)\mathrm{d}\xi$. Then,

$$f(X_0 + h/2) - f(X_0 - h/2) = \int_{X_0-h/2}^{X_0+h/2} V(\xi, 0)\mathrm{d}\xi = h\overline{V}^0_0 ,$$

and

$$\begin{aligned} & f(X_0 + 3h/2) - f(X_0 - 3h/2) \\ &= \int_{X_0-3h/2}^{X_0+3h/2} V(\xi, 0)\mathrm{d}\xi \\ &= \int_{X_0-3h/2}^{X_0-h/2} V(\xi, 0)\mathrm{d}\xi + \int_{X_0-h/2}^{X_0+h/2} V(\xi, 0)\mathrm{d}\xi + \int_{X_0+h/2}^{X_0+3h/2} V(\xi, 0)\mathrm{d}\xi \\ &= h\left(\overline{V}^0_{-1} + \overline{V}^0_0 + \overline{V}^0_1\right) . \end{aligned}$$

Since $f'(X_0) = V(X_0, 0)$, the equation (6.1) of the compact scheme for the first derivative implies the following,

$$\begin{aligned} V(X_0, 0) &= \frac{9}{8}\frac{h\overline{V}^0_0}{h} - \frac{1}{8}\frac{h\left(\overline{V}^0_{-1} + \overline{V}^0_0 + \overline{V}^0_1\right)}{3h} \\ &= \frac{9\overline{V}^0_0}{8} - \frac{\overline{V}^0_{-1} + \overline{V}^0_0 + \overline{V}^0_1}{24} \\ &= -\frac{1}{24}\overline{V}^0_{-1} + \frac{13}{12}\overline{V}^0_0 - \frac{1}{24}\overline{V}^0_1 . \qquad \square \end{aligned}$$

In a similar method, if $f(x)$ is continuously differentiable twice on I, we can derive an equation for $f(x)$ and $f''(x)$,

$$\begin{aligned} & f''_0 + \alpha\left(f''_1 + f''_{-1}\right) + \beta\left(f''_2 + f''_{-2}\right) \\ &= a\,\frac{f_1 - 2f_0 + f_{-1}}{h^2} + b\,\frac{f_2 - 2f_0 + f_{-2}}{4h^2} + c\,\frac{f_3 - 2f_0 + f_{-3}}{9h^2} . \end{aligned} \tag{6.2}$$

If $f(x)$ is sufficiently smooth, the Taylor series expansion gives

$$(LHS) = (1 + 2(\alpha + \beta))\, f_0'' + 2(\alpha + 4\beta)\frac{h^2}{2} f_0^{(4)} + 2(\alpha + 16\beta)\frac{h^4}{4!} f_0^{(6)} + \cdots \ ,$$

and

$$(RHS) = 2(a + b + c)\frac{1}{2} f_0'' + 2(a + 4b + 9c)\frac{h^2}{4!} f_0^{(4)} + 2(a + 16b + 81c)\frac{h^4}{6!} f_0^{(6)} + \cdots \ .$$

The 2nd order accuracy requires the following equation,

$$a + b + c = 1 + 2\alpha + 2\beta \ . \tag{6.3}$$

Lemma 3. *The compact scheme (6.3) gives the following approximation for delta*

$$\Delta(X_0, 0) = \frac{\overline{V}_1^0 - \overline{V}_{-1}^0}{2h} \ .$$

Proof. Let $f(x) \equiv \int_{X_0-3h/2}^{x} V(\xi, 0)\mathrm{d}\xi$. Then,

$$\begin{aligned}
f(X_0 - 3h/2) &= \int_{X_0-3h/2}^{X_0-3h/2} V(\xi, 0)\mathrm{d}\xi = 0 \ , \\
f(X_0 - h/2) &= \int_{X_0-3h/2}^{X_0-h/2} V(\xi, 0)\mathrm{d}\xi = h\overline{V}_{-1}^0 \ , \\
f(X_0 + h/2) &= \int_{X_0-3h/2}^{X_0+h/2} V(\xi, 0)\mathrm{d}\xi = h\left(\overline{V}_{-1}^0 + \overline{V}_0^0\right) \ , \\
f(X_0 + 3h/2) &= \int_{X_0-3h/2}^{X_0+3h/2} V(\xi, 0)\mathrm{d}\xi = h\left(\overline{V}_{-1}^0 + \overline{V}_0^0 + \overline{V}_1^0\right) \ .
\end{aligned}$$

Since $f''(X_0) = \Delta(X_0, 0)$, the equation (6.2) with $\alpha = \beta = b = 0$ gives

$$\begin{aligned}
&\Delta(X_0, 0) \\
&= a\,\frac{f(X_0 + h/2) - 2f(X_0) + f(X_0 - h/2)}{h^2/4} + c\,\frac{f(X_0 + 3h/2) - 2f(X_0) + f(X_0 - 3h/2)}{9h^2/4} \\
&= \frac{4}{h^2}\left(ah(2\overline{V}_{-1}^0 + \overline{V}_0^0) + \frac{ch}{9}\left(\overline{V}_{-1}^0 + \overline{V}_0^0 + \overline{V}_1^0\right) - 2\left(a + \frac{c}{9}\right) f(X_0)\right) \ .
\end{aligned}$$

$f(X_0)$ is unknown and the $f(X_0)$ term disappears when $a + c/9 = 0$. Since the condition (6.3) gives $a + c = 1$, we have

$$a = -\frac{1}{8}, \qquad c = \frac{9}{8} \ .$$

That is, we have the following approximation for delta,

$$\Delta(X_0,0) = \frac{4}{h}\left(-\frac{1}{8}(2\overline{V}^0_{-1} + \overline{V}^0_0) + \frac{1}{8}\left(\overline{V}^0_{-1} + \overline{V}^0_0 + \overline{V}^0_1\right)\right)$$
$$= \frac{\overline{V}^0_1 - \overline{V}^0_{-1}}{2h}. \qquad \square$$

ACKNOWLEDGMENTS

This research of Kim was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2010-0012584). The work of Moon was supported by the Kyungwon University Research Fund in 2011. The authors would like to thank the very helpful comments from anonymous referees.

REFERENCES

[1] T. Achdou, O. Pironneau. *Computational methods for option pricing*. SIAM Philadelphia, 2005.
[2] P. Boyle. A lattice framework for option pricing with two state variables. *Journal of Financial and Quantitative Analysis*, 23(1):1–12, 1988.
[3] P. Boyle, J. Evnine, and S. Gibbs. Numerical evaluation of multivariate contingent claims. *The Review of Financial Studies*, 2(2):241–250, 1989.
[4] P. Boyle, and S. H. Lau. Bumping up against the barrier with the binomial method. *Journal of Derivatives*, 1:6–14, 1994.
[5] M. Broadie and J. Detemple. American option valuation: new bounds, approximations, and a comparison of existing methods. *Review of Financial Studies*, 9:1211–1250, 1996.
[6] T. H. F. Cheuk and T. C. F. Vorst. Complex barrier options. *Journal of Derivatives*, 4:8–22, 1996.
[7] J. Cox, S. Ross, and M. Rubinstein. Option pricing: A simplified approach. *Journal of Financial Economics*, 7:229–263, 1979.
[8] E. Derman, I. Kani, D. Ergener and I. Bardhan. Enhanced Numerical Methods for Options with Barriers. *Financial Analysts Journal*, 51(6):65–74, 1995.
[9] S. Figlewski and B. Gao. The adaptive mesh model: a new approach to efficient option pricing. *Journal of Financial Economics*, 53:313–351, 1999.
[10] M. Gaudenzi and M. A. Lepellere. Pricing and hedging American barrier options by a modified binomial method. *International Journal of Theoretical and Applied Finance*, 9(4):533–553, 2006.
[11] M. Gaudenzi and F. Pressacco. An efficient binomial method for pricing american options. *Decisions in Economics and finance*, 26:1–17, 2003.
[12] E. G. Haug. *The complete guide to option pricing formulas*. McGraw-Hill, 1997.
[13] D.J. Higham. *An introduction to financial option valuation*. Cambridge University Press, 2004.
[14] B. Kamrad and P. Ritchken. Multinomial approximating models for options with k state variables. *Management science*, 37(12), 1991.
[15] Y. K. Kwok. *Mathematical models of financial derivatives*. Springer-Verlag, Singapore, 1998.
[16] Y. D. Lyuu. *Financial Engineering and Computation*. Cambridge, 2002.
[17] B. Øksendal. *Stochastic differential equations*. Springer, Berlin, 1998.
[18] E. Reimer and M. Rubinstein. Unscrambling the binary code. *Risk Magazine*, 4, 1991.
[19] P. Ritchken. On pricing barrier options. *Journal of Derivatives*, 3:19–28, 1995.
[20] P. Wilmott, J. Dewynne, S. Howison. *Option Pricing: Mathematical Models and Computation*. Oxford Financial Press, 1993.