Journal of Computational and Applied Mathematics 249 (2013) 107–119

Contents lists available at SciVerse ScienceDirect

Journal of Computational and Applied Mathematics

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

Spectral binomial tree: New algorithms for pricing barrier options☆

Yoshifumi Muroi [a,*], Takashi Yamada [b]

[a] Graduate School of Economics and Management, Tohoku University, 27-1 Kawauchi, Aoba-Ku, Sendai City, 980-8576, Japan

[b] Department of Computational Intelligence and Systems Science, Tokyo Institute of Technology, 4259 Nagatsuta-cho, Midori-ku, Yokohama City, 226-8502, Japan

HIGHLIGHTS

- Binomial tree is popular but slow to converge to calculate barrier option prices.
- We incorporate the spectral expansion method into it for pricing barrier options.
- The original idea comes from the eigenexpansion approach in PDEs.
- It can compute double barrier options with one billion steps within 0.07 s.
- The prices are always the same as those by conventional binomial trees.

ARTICLE INFO

Article history:
Received 27 March 2012
Received in revised form 1 October 2012

Keywords:
Spectral methods
Tridiagonal matrix
Eigenvalue problems
Double barrier options
Rebates

ABSTRACT

This paper introduces new and significantly fast algorithms to evaluate the price of double barrier options using binomial trees. To compute the price of double barrier options accurately, trees with large numbers of steps must be used, which is time consuming. In order to overcome this weakness, we develop new computational algorithms based on the spectral expansion method. The original idea of this method is coming from the eigenexpansion approach in PDEs. We show that this method enables us to compute double barrier options within 0.07 s, even if we use binomial trees with one billion steps. Moreover, this algorithm is easy to implement. In addition, the prices obtained by the proposed approach are always the same as those obtained by conventional binomial trees and show a good approximation to those by earlier studies.

1. Introduction

This paper introduces new and fast algorithms to evaluate the price of double barrier options by the binomial tree approach. Barrier options are the most frequently traded exotic options. They are popular because their prices are lower than those of regular European options. Among these, double barrier options have two absorbing boundaries: an upper boundary and a lower boundary. The payoff of barrier options is the same as that of regular European options as long as the price of underlying assets does not cross the knock-out boundaries before maturity date. However, if the underlying asset price process reaches one of these boundaries before maturity date, the options become worthless.

The pricing of barrier options was studied by Merton [1] in the early 1970s and there have been a wide variety of previous studies on this topic. The pricing formulas for double barrier options have been also studied since the early

☆ The authors are grateful for valuable comments and suggestions from one anonymous referee. We also thank Prof. Akira Yamazaki (Hosei University, Japan) for practical comments.

* Corresponding author. Tel.: +81 22 795 6308; fax: +81 22 795 6308.
E-mail addresses: ymuroi@econ.tohoku.ac.jp (Y. Muroi), tyamada@trn.dis.titech.ac.jp (T. Yamada).

0377-0427/$ – see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.cam.2012.10.036

1990s: Kunitomo and Ikeda [2] found the pricing formulas for double barrier options in the Black and Scholes model using the reflection principle infinitely many times. Geman and Yor [3] considered the Laplace transform approach to evaluate double barrier options. To invert the Laplace transform, Pelsser [4] used the residue theorem in complex analysis, by which Pelsser derived series expressions for the prices of double barrier options and rebates. Davydov and Linetsky [5] used the eigenexpansion approach to evaluate the price of double barrier options. They derived the price of barrier options not only in the Black–Scholes model but also in the constant elasticity of variance model (CEV model). (CEV model was introduced by Cox [6] to capture the leverage effect. See also Boyle and Tian [7] for pricing barrier options in the CEV model, for example.)

In addition to these previous studies of the derivation of the closed form formulas for double barrier options, numerical schemes for pricing of double barrier options are still important. If there is an algorithm for pricing double barrier options, the prices of double barrier options with different payoff structures can be computed with the same computer program. However, the binomial tree approach converges extremely slowly. Thus, an efficient numerical scheme for pricing barrier options must be developed. For example, Lyuu [8] and Dai et al. [9] introduced the combinatorics approach for barrier option pricing. In contrast, we employ the spectral expansion approach, which comes from the eigenexpansion methods in PDEs. The method of separating variables on the binomial tree makes it possible to obtain an eigenvalue problem for a tridiagonal matrix with closed form solutions We show that this enables us to reduce significantly in the time of computation of the price of double barrier options.

The remainder of this paper is organized as follows: The pricing methods of double barrier options by the classical binomial tree approach are reviewed in Section 2. The spectral expansion methods for the binomial tree are explained in Section 3. The pricing methods of rebates at a hit and knock-out options with a single barrier are also introduced in this section. Numerical demonstrations are presented and the efficiency of our approach is shown in Section 4. The last section contains concluding remarks.

2. Binomial tree approach

In this section, we briefly explain the classical binomial tree model of Cox et al. [10]. The basic idea for the computation of double barrier options by the binomial tree approach is also presented.

The binomial tree is a computational method for pricing options on securities whose price process is governed by the geometric Brownian motion,

$$dP_t = P_t(rdt + \sigma dZ_t), \qquad P_0 = s, \tag{2.1}$$

where $\{Z_t\}$ is a standard Brownian motion under the risk-neutral measure Q. It is a popular numerical method in financial engineering and has been explained in many textbooks such as Hull [11].

We will review the computation algorithms for pricing European options. A model is constructed with N periods and maturity date of the options is fixed as $T = N\Delta t$. The underlying asset price at time $i\Delta t$ $(i = 0, 1, \ldots, N-1)$ is denoted by $S_{i\Delta t}$ and the initial price of the underlying asset by $S_0 = s$. If the underlying asset value at time $i\Delta t$ $(i = 0, 1, \ldots, N-1)$ is given, the price for the underlying asset will move to $uS_{i\Delta t}$ or $dS_{i\Delta t}$ at time $(i+1)\Delta t$ where new constants, d and u, satisfy, $d < 1 < u$. We also impose an additional assumption $u = 1/d$. The probabilities of moving from $S_{i\Delta t}$ to $uS_{i\Delta t}$ (upward) or $dS_{i\Delta t}$ (downward) are given by p and q $(=1-p)$ respectively. We fix the parameters u, d, and p so that the expectation and variance are consistent with the geometric Brownian motion (2.1). If higher order terms than $O(\Delta t)$ are neglected, parameters u, d, and p, are given by

$$u = e^{\sigma\sqrt{\Delta t}}, \qquad d = e^{-\sigma\sqrt{\Delta t}}, \qquad p = \frac{e^{r\Delta t} - d}{u - d}.$$

If the underlying asset price moves upward j times and downward $n-j$ times until $n\Delta t$, the underlying asset price at time $n\Delta t$ is given by $S_{n\Delta t} = su^j d^{n-j} = se^{\sigma(2j-n)\sqrt{\Delta t}}$. The price of European options with payoff function $\Phi(x)$ and maturity date T is given by

$$C = e^{-rN\Delta t}\sum_{j=0}^{N} {}_NC_j p^j(1-p)^{N-j}\Phi(su^jd^{N-j}) = e^{-rT}\sum_{j=0}^{N} {}_NC_j p^j(1-p)^{N-j}\Phi(su^{2j-N}).$$

European options with a payoff functions $\Phi(x) = (x-K)^+$ and $\Phi(x) = (K-x)^+$ are called European call options and European put options, respectively. The price of options at time $i\Delta t$ is denoted by $C(i\Delta t, su^j)$ when the underlying asset price is given by su^j at time $i\Delta t$. The price of options is given by the backward induction algorithm:

$$C(i\Delta t, su^j) = e^{-r\Delta t}\{pC((i+1)\Delta t, su^{j+1}) + qC((i+1)\Delta t, su^{j-1})\}, \tag{2.2}$$

where the price of options at maturity date is given by $C(N\Delta t, su^j) = \Phi(su^j)$. This method is called the binomial tree approach, a popular algorithm for option pricing. However, it is computationally burdensome when using a tree with many steps.

Double knock-out options are contingent claims which pay the promised payoff at maturity date as long as the underlying asset price does not cross the knock-out boundaries. On the other hand, if the underlying asset price crosses one of the

Table 1
Double barrier option price by binomial trees.

Meshes	Price	Error (%)
50	0.126017	98.26
100	0.0921141	44.92
500	0.0724787	14.03
1000	0.0746472	17.44
Exact	0.0635619	

boundaries, options become worthless. If there is only a single boundary higher (resp. lower) than the underlying asset price, it is called up-and-out (resp. down-and-out) options.

The pricing of double knock-out options by the binomial tree approach will be explained. Consider double barrier options with an upper boundary m and a lower boundary l. Choose two integers, L and M, both of which satisfy the conditions,

$$su^{(L-1)} \le l < su^L \quad \text{and} \quad su^M < m \le su^{(M+1)}.$$

Once the underlying asset price reaches one of these boundaries, options become worthless. This is represented by the boundary conditions,

$$C(i\Delta t, su^{L-1}) = C(i\Delta t, su^{M+1}) = 0 \tag{2.3}$$

for an arbitrary integer $1 \le i \le N$. In addition to regular European options, the price of options at maturity date is given by

$$C(N\Delta t, su^j) = \begin{cases} \Phi(su^j) & (L \le j \le M) \\ 0 & (\text{otherwise}) \end{cases} \tag{2.4}$$

and the price of double barrier options is obtained by using the backward induction algorithm (2.2). This method is straightforward; however, alone it converges too slowly to compute the price of barrier options. We call this method the "conventional binomial tree".

In order to acquire reliable results, N must be extremely large. Consider the price of double knock-out call options with maturity date, $T = 1$, the exercise price, $K = 100$, and the lower and upper boundaries, $l = 70$ and $m = 130$. The volatility and the initial price of underlying assets are given by $\sigma = 0.4$ and $s = 100$ respectively. Assume also that the spot interest rate is given by $r = 0.1$. Table 1 shows the price of double barrier options computed by $N = 50$, 100, 500 and 1000 mesh binomial trees. This table indicates that the convergence speed of this method is slow, even with conventional binomial trees with relatively fine meshes; The errors are still large. That is why tree approach is not commonly used to evaluate barrier options. In the next section, we will provide new algorithms to compute double barrier options that overcome these shortcomings.

3. Spectral expansion

We introduce a newly suggested approach that has its origin in the spectral expansion approach of partial differential equations, to compute double barrier options with binomial trees. The spectral expansion approach is deeply related to the diagonalization of the symmetric matrix in linear algebra. This method enables us to compute the price of double barrier options with significant efficiency even with a binomial tree with huge mesh size. The pricing rebate contracts and knock-out options with one-sided barriers are also dealt with in this section, using the spectral expansion approach.

3.1. Pricing double barrier options

The computational algorithms for double barrier options are given in (2.2)–(2.4). Formula (2.2) is further computed as

$$\begin{aligned} C(i\Delta t,\ su^j) &= e^{-r\Delta t}\{pC((i+1)\Delta t,\ su^{j+1}) + qC((i+1)\Delta t,\ su^{j-1})\} \\ &= e^{-r\Delta t}(4pq)^{1/2}\left(\frac{1}{2}\left(\frac{p}{q}\right)^{1/2} C((i+1)\Delta t,\ su^{j+1}) + \frac{1}{2}\left(\frac{p}{q}\right)^{-1/2} C((i+1)\Delta t,\ su^{j-1})\right). \end{aligned}$$

This corresponds to the discrete version of the Girsanov formula for a one period model. Introduce a new variable $D(i, j)$, defined as

$$D(i,\ j) = \left(\frac{p}{q}\right)^{j/2} C(i\Delta t,\ su^j). \tag{3.5}$$

This yields the computational algorithm for the function $D(i, j)$, given by

$$D(i,\ j) = e^{-r\Delta t}(4pq)^{1/2}\left(\frac{1}{2}D(i+1,\ j+1) + \frac{1}{2}D(i+1,\ j-1)\right) \tag{3.6}$$

$$D(i, L-1) = D(i, M+1) = 0, \tag{3.7}$$

with the terminal condition

$$D(N,\ j) = \left(\frac{p}{q}\right)^{j/2} \Phi(su^j). \tag{3.8}$$

The initial price of double barrier options is thus given by

$$C = C(0, s) = D(0,\ 0).$$

The price of double barrier options is computed using the backward induction algorithm. Let us assume that the function $D(i,\ j)$ is represented by the product of two functions, $h(i)$ and $g(j)$, *i.e.* the function $D(i,\ j)$ is represented as $D(i,\ j) = h(i)g(j)$. Note that the function $h(\cdot)$ depends on the time only and that the function $g(\cdot)$ depends on the price of underlying assets only. This formula is plugged into (3.6), which yields

$$\begin{aligned} h(i)g(j) &= e^{-r\Delta t}(4pq)^{\frac{1}{2}} \left\{\frac{1}{2}h(i+1)g(j+1) + \frac{1}{2}h(i+1)g(j-1)\right\} \\ &= e^{-r\Delta t}(4pq)^{\frac{1}{2}} h(i+1) \left\{\frac{1}{2}g(j+1) + \frac{1}{2}g(j-1)\right\} \end{aligned}$$

with the boundary condition,

$$g(M+1) = g(L-1) = 0.$$

These calculations yield,

$$\begin{aligned} &\frac{h(i)}{h(i+1)} = e^{-r\Delta t}(4pq)^{\frac{1}{2}} \frac{\frac{1}{2}g(j+1) + \frac{1}{2}g(j-1)}{g(j)} \\ &g(M+1) = g(L-1) = 0. \end{aligned} \tag{3.9}$$

Note that the left hand side of formula (3.9) depends on the time only and the right-hand side of (3.9) depends on the price of underlying asset only. This relation reveals that both sides of formula (3.9) are really a constant. We denote this constant by λ'. These formulas are then rewritten as

$$h(i+1) = \lambda' h(i) \tag{3.10}$$

$$A\mathbf{G} = \lambda\mathbf{G}, \tag{3.11}$$

where a new constant λ is an eigenvalue of the matrix A; This variable is related to λ' by

$$\lambda = \frac{1}{e^{-r\Delta t}(4pq)^{\frac{1}{2}}}\lambda'. \tag{3.12}$$

A matrix A and a vector $\mathbf{G}$ are given by

$$A = \begin{pmatrix} 0 & \frac{1}{2} & & & & \\ \frac{1}{2} & 0 & \frac{1}{2} & & & \\ & & \ddots & \ddots & \ddots & \\ & & & \frac{1}{2} & 0 & \frac{1}{2} \\ & & & & \frac{1}{2} & 0 \end{pmatrix}, \qquad \mathbf{G} = \begin{pmatrix} g(L) \\ g(L+1) \\ \vdots \\ g(M-1) \\ g(M) \end{pmatrix}.$$

If we normalize the function $h(j)$ so that $h(N) = 1$ must be satisfied, formula (3.10) yields the closed-form expression for the function $h(\cdot)$ as

$$h(i) = \lambda'^{N-i}.$$

To obtain the eigenvalue for the matrix A, we use the following well-known fact in computational linear algebra. See Meyer [12], for example.

Fact 3.1. *Consider an $n \times n$ real valued tridiagonal matrix,*

$$\begin{pmatrix} b & c & & & & \\ a & b & c & & & \\ & & & & & \\ & \ddots & \ddots & \ddots & & \\ & & & & & \\ & & & a & b & c \\ & & & & a & b \end{pmatrix},$$

then the eigenvalue λ_k and the corresponding eigenvector $\mathbf{x}_k$ are given by

$$\lambda_k = b + 2\sqrt{ac}\cos\frac{k\pi}{n+1} \quad (1 \le k \le n)$$
$$\mathbf{x}_k = {}^t\left(\sin\frac{k\pi}{n+1}, \sqrt{\frac{a}{c}}\sin\frac{2k\pi}{n+1}, \ldots, \left(\sqrt{\frac{a}{c}}\right)^{n-1}\sin\frac{nk\pi}{n+1}\right).$$

The dimension of vector $\mathbf{G}$ is denoted by R, *i.e.* $R = M - L + 1$. The eigenvalue problem (3.11) has an eigenvalue of

$$\lambda_k = \cos\frac{k\pi}{R+1} \quad (1 \le k \le R)$$

and a corresponding eigenvector of

$$\mathbf{G}_k = {}^t\left(\sin\frac{k\pi}{R+1}, \sin\frac{2k\pi}{R+1}, \ldots, \sin\frac{Rk\pi}{R+1}\right).$$

Let us calculate the length of the eigenvector $\|\mathbf{G}_k\|$. It is given by

$$\begin{aligned} \|\mathbf{G}_k\|^2 &= \sin^2\frac{k\pi}{R+1} + \sin^2\frac{2k\pi}{R+1} + \cdots + \sin^2\frac{Rk\pi}{R+1} \\ &= \frac{1}{2}\left(1 - \cos\frac{2k\pi}{R+1}\right) + \cdots + \frac{1}{2}\left(1 - \cos\frac{2Rk\pi}{R+1}\right) \\ &= \frac{R}{2} - \frac{1}{2}\left(\cos\frac{2k\pi}{R+1} + \cdots + \cos\frac{2Rk\pi}{R+1}\right). \end{aligned}$$

Introduce a new variable $S = \cos\frac{2k\pi}{R+1} + \cdots + \cos\frac{2Rk\pi}{R+1}$. The identity

$$1 + \exp\left(\sqrt{-1}\frac{2k\pi}{R+1}\right) + \cdots + \exp\left(\sqrt{-1}\frac{2Rk\pi}{R+1}\right) = 0$$

yields to

$$S = \mathrm{Re}\left[\exp\left(\sqrt{-1}\frac{2k\pi}{R+1}\right) + \cdots + \exp\left(\sqrt{-1}\frac{2Rk\pi}{R+1}\right)\right] = -1.$$

This yields the result

$$\|\mathbf{G}_k\|^2 = \frac{R+1}{2}.$$

Let $\mathbf{f}_k$ be a normalized vector of the vector $\mathbf{G}_k$ given by

$$\mathbf{f}_k = \mathbf{G}_k/\|\mathbf{G}_k\| = \sqrt{\frac{2}{R+1}}\mathbf{G}_k.$$

The vector $\mathbf{f}_k$ is also an eigenvector for the matrix A corresponding to the eigenvalue λ_k. Since the matrix A is symmetric, the vectors $\{\mathbf{f}_k\}_{k=1}^R$ are the complete orthonormal basis for space $\mathbf{R}^R$. Since the constant λ_k is related to the eigenvalue of matrix A by formula (3.12), the constant λ' in (3.10) is given by

$$\lambda'_k = e^{-r\Delta t}(4pq)^{\frac{1}{2}}\lambda_k.$$

The function $h(\cdot)$ corresponding to the eigenvalue λ_k is denoted by $h_k(\cdot)$ and is given by

$$h_k(i) = \left(e^{-r\Delta t}(4pq)^{\frac{1}{2}}\cos\frac{k\pi}{R+1}\right)^{N-i}.$$

A new vector $\mathbf{Y}_k(i)$ is defined by

$$\mathbf{Y}_k(i) = h_k(i)\mathbf{f}_k$$

and the $(L+j+1)$-th element of this vector is denoted by $Y_k(i,j)$. These elements $Y_k(i,j)$ satisfy the algorithm (3.6) and boundary condition (3.7). Introduce R constants $a_1, \ldots, a_R$. Even if $D(i,j)$ has a representation,

$$D(i,j) = \sum_{k=1}^{R} a_k Y_k(i,j),$$

$D(i,j)$ still satisfies the algorithm (3.6) and the boundary condition (3.7). Constants a_k $(k = 1, \ldots, R)$ are determined so that the terminal condition (3.8) is satisfied:

$$\sum_{k=1}^{R} a_k Y_k(N,j) = D(N,j).$$

Introduce a new vector $\mathbf{D}(i)$ as

$$\mathbf{D}(i) = {}^t(D(i,L), \ldots, D(i,M)).$$

Since the function h_k is normalized as $h_k(N) = 1$, there is a special formula for a vector $\mathbf{D}(N)$:

$$\mathbf{D}(N) = \sum_{k=1}^{R} a_k h_k(N)\mathbf{f}_k = \sum_{k=1}^{R} a_k \mathbf{f}_k.$$

Define also a new vector, $\mathbf{\Psi}$, by

$$\mathbf{\Psi} = {}^t\left(\left(\frac{p}{q}\right)^{L/2} \Phi\left(su^{\frac{L}{2}}\right), \left(\frac{p}{q}\right)^{\frac{L+1}{2}} \Phi\left(su^{\frac{L+1}{2}}\right), \ldots, \left(\frac{p}{q}\right)^{\frac{M}{2}} \Phi\left(su^{\frac{M}{2}}\right)\right). \tag{3.13}$$

Because of formula (3.5), this vector is identical to $\mathbf{D}(N)$; $\mathbf{D}(N) = \mathbf{\Psi}$. Remember that $\{\mathbf{f}_k\}_{k=1}^{R}$ is an orthonormal basis for $\mathbf{R}^R$; If the constant a_k is fixed at

$$a_k = \mathbf{D}(N) \cdot \mathbf{f}_k = \mathbf{\Psi} \cdot \mathbf{f}_k, \tag{3.14}$$

the constant $D(i,j)$ satisfies the algorithm (3.6), boundary condition (3.7) and terminal condition (3.8). Using the relation (3.5) yields the price of double barrier options $C(i, su^j)$ and the initial price of double barrier options $D(0,0)(=C(0,s))$. The initial price of double barrier options is given by $D(0,0)$, which is the $(L+1)$-th elements vector $\mathbf{D}(0)$: $(L+1)$-th element of vector $\mathbf{D}(0)$

$$\mathbf{D}(0) = {}^t(D(0,L), \ldots, D(0,M)) = \sum_{k=1}^{R} (\lambda'_k)^N a_k \mathbf{f}_k. \tag{3.15}$$

Note that the eigenvalue λ'_k $(1 \le k \le R)$ satisfies the condition $-1 \le \lambda'_k \le 1$ and that if the relation $|\lambda'_k| \approx 1$ is not satisfied, we have an approximation formula $h_k(0) \approx 0$. This means that the price of double barrier options is accurate even if not all of the elements in the summation (3.15) are added. In our numerical demonstrations, we ignore the elements in summation (3.15) if the relation

$$|\lambda'_k|^N < \frac{\epsilon e^{rT}}{(4pq)^{N/2}}$$

is satisfied. This condition is equivalent to

$$\log|\lambda'_k| < \frac{rT + \log\epsilon}{N} - \frac{1}{2}\log 4pq.$$

This process reduces our computations significantly and computing becomes much faster. In our numerical studies in the next section, we fix this constant at $\epsilon = 10^{-10}$, following Pelsser [4].

3.2. Computation of Greeks

Greeks are quantities that represent the sensitivity of the price of derivative securities with respect to changes in the prices of underlying assets or parameters. We consider two kinds of Greeks, delta and gamma. Delta measures the sensitivity of the option price with respect to changes in the price of underlying assets. It is defined by the first derivative of the option value function with respect to the price of underlying assets. Another Greek, gamma, measures the sensitivity of delta with respect to changes in the price of underlying assets. It is thus defined by second derivative of the option value function with respect to the price of underlying assets. These quantities are important for risk management for option holders and writers. Computation of Greeks using the binomial tree approach was initiated by Pelsser and Vorst [13].

The extended (N-step) binomial tree is an $N+2$ binomial tree starting from $-2\Delta t$. The extended binomial tree approach is popular for practitioners because delta and gamma can be obtained accurately and easily. Extended binomial delta and gamma are given by

$$\Delta = \frac{C(0, su^2) - C(0, sd^2)}{su^2 - sd^2}$$

$$\Gamma = \frac{\frac{C(0,su^2)-C(0,s)}{su^2-s} - \frac{C(0,s)-C(0,sd^2)}{s-sd^2}}{su - sd}.$$

We will use these formulas for numerical demonstrations in Section 4 to compute delta and gamma for double barrier options.

3.3. Pricing rebates

It is common in double knock-out options to offer a rebate as soon as the underlying asset price hits one of the barriers. In this subsection, we compute the price of a rebate at a hit using the spectral expansion approach. Pricing a rebate at hit with the spectral expansion approach has been independently considered by Pelsser [4] and Davydov and Linetsky [14].

Consider rebates with upper and lower boundaries. The lower and upper boundaries are fixed at l and m, respectively. Rebates have a cashflow only if the underlying asset price process reaches one of the boundaries before maturity date, $T(= N\Delta t)$. Assume that the first hitting time for the lower or upper boundary is equal to $i\Delta t$ $(i = 1, \ldots, N-1)$. We then assume that rebates have a cashflow, $\phi_l(i)$ (resp. $\phi_m(i)$), as soon as the underlying asset price reaches the lower (resp. upper) boundary. This contingent claim is evaluated using the spectral binomial tree approach.

Consider a contingent claim that pays one dollar if and only if the underlying asset price reaches su^h at time $I\Delta t$ before it hits the upper or lower boundary. This is an Arrow–Debreu security for the binomial tree model. The price of this contingent claim is denoted by $c_h^I(i\Delta t, su^j)$ if the underlying asset price is at su^j at time $i\Delta t$. Consider another contingent claim that pays one dollar if and only if the underlying asset price process first crosses the lower (resp. upper) boundary at time $I\Delta t$. The price of this security is denoted by $R_m^I = e^{-r\Delta t} p c_M^{I-1}(0, s)$ (resp. $R_l^I = e^{-r\Delta t} q c_L^{I-1}(0, s)$). Using this setup, the price of a rebate is given by

$$\begin{aligned} C &= \sum_{n=M,M+2,\ldots,N'} \phi_m(n) R_m^n + \sum_{n=-L,-L+2,\ldots,N''} \phi_l(n) R_l^n \\ &= pe^{-r\Delta t} \sum_{n=M,M+2,\ldots,N'} \phi_m(n+1) c_M^{n-1}(0, s) + qe^{-r\Delta t} \sum_{n=-L,-L+2,\ldots,N''} \phi_l(n+1) c_L^{n-1}(0, s) \end{aligned}$$

where N' and N'' are $N-1$ or N and must be chosen properly.

The price of rebates can be calculated if we propose the numerical method of $c_L^n(0, s)$ and $c_M^n(0, s)$. However, $c_L^n(0, s)$ and $c_M^n(0, s)$ represent the prices of contingent claims with maturity date $n\Delta t$ and payoff functions $\Phi = {}^t(0, \ldots, 0, 1)$ and $\Phi = {}^t(1, 0, \ldots, 0)$, respectively. This enables us to compute $c_L^n(0, s)$ and $c_M^n(0, s)$ by the same manner as the spectral methods presented in Section 3.1. We discuss the computational method of $c_M^n(0, s)$ only. Transformation (3.5) yields

$$\Psi = {}^t\left(0, \ldots, \left(\frac{p}{q}\right)^{\frac{M}{2}}\right)$$

and (3.14) yields the coefficient a_k, given by

$$a_k = \Psi \cdot \mathbf{f}_k = \left(\frac{p}{q}\right)^{\frac{M}{2}} \sqrt{\frac{2}{R+1}} \sin\left(\frac{Rk\pi}{R+1}\right).$$

Introduce a new vector $\mathbf{D}_M^n$ defined by $\mathbf{D}_M^n = ((\frac{p}{q})^L c_M^n(0, su^L), \ldots, (\frac{p}{q})^M c_M^n(0, su^M))$, where $c_M^n(0, su^i)$ represents the price of Arrow–Debreu security. Formula (3.15) yields

$$\begin{aligned} \mathbf{D}_M^n &= \sum_{k=1}^{R} (\lambda_k)^n a_k \mathbf{f}_k \quad M+1 \le I \\ &= {}^t(0, \ldots, 0) \quad I \le M \end{aligned}$$

where λ_k' is given by $\lambda_k' = e^{-r\Delta t}(4pq)^{\frac{1}{2}} \lambda_k = e^{-r\Delta t}(4pq)^{\frac{1}{2}} \cos\frac{k\pi}{R+1}$. This leads to

$$\begin{aligned} c_M^n(0, s) &= \sum_{k=1}^{R} (\lambda_k')^n a_k \left\{\sqrt{\frac{2}{R+1}} \sin\frac{(1-L)k\pi}{R+1}\right\} \\ &= \frac{2p^{M/2}}{(R+1)q^{M/2}} \sum_{k=1}^{R} (\lambda_k')^n \sin\left(\frac{Rk\pi}{R+1}\right) \sin\frac{(1-L)k\pi}{R+1}. \end{aligned}$$

Similar calculation yields

$$c_L^n(0,s) = \frac{2p^{L/2}}{(R+1)q^{L/2}} \sum_{k=1}^{R} (\lambda_k')^n \sin\left(\frac{k\pi}{R+1}\right) \sin\frac{(1-L)k\pi}{R+1}.$$

Example 3.1. As a special case, we consider rebates whose payoff for upper boundary $\phi_m(i)$ is a constant, F, and the payoff for lower boundary $\phi_l(i)$ equals 0. There is a closed form formula for the price of this contingent claim given by

$$\begin{aligned} C_u &= \frac{2Fe^{-r\Delta t}p^{\frac{M+2}{2}}}{(R+1)q^{\frac{M}{2}}} \sum_{n=0}^{N'} \sum_{k=1}^{R} (\lambda_k')^{M+2n} \sin\left(\frac{Rk\pi}{R+1}\right) \sin\frac{(1-L)k\pi}{R+1} \\ &= \frac{2Fe^{-r\Delta t}p^{\frac{M+2}{2}}}{(R+1)q^{\frac{M}{2}}} \sum_{k=1}^{R} \lambda_k'^M \frac{1-\lambda_k'^{2(N'+1)}}{1-\lambda_k'^2} \sin\left(\frac{Rk\pi}{R+1}\right) \sin\frac{(1-L)k\pi}{R+1} \end{aligned} \tag{3.16}$$

where N' is an integer that is properly chosen by $M+2N' = N-2$ or $M+2N' = N-1$. The price of rebates which has payoff F for the lower boundary and nothing for the upper boundary is also given by

$$C_d = \frac{2Fe^{-r\Delta t}p^{\frac{L}{2}}}{(R+1)q^{\frac{L-2}{2}}} \sum_{k=1}^{R} \lambda_k'^{-L} \frac{1-\lambda_k'^{2(N'+1)}}{1-\lambda_k'^2} \sin\left(\frac{k\pi}{R+1}\right) \sin\frac{(1-L)k\pi}{R+1}, \tag{3.17}$$

where N' is an integer which is properly chosen by $-L+2N' = N-2$ or $-L'+2N' = N-1$.

Example 3.2. It is straightforward to derive the price of rebates with a one-sided barrier. The price of rebates only with a lower boundary can be obtained if we substitute M into $N-1$ in pricing formula (3.16); The price of rebates only with an upper boundary can be obtained if we substitute L into $1-N$ in pricing formula (3.17).

Example 3.3. The risk-neutral probability to reach the underlying asset price to su^{2h} at time $I\Delta t$ without crossing the upper or lower boundary is also computed by the spectral method. Define stopping times τ_L and τ_U by $\tau_L = \inf\{i\Delta t > 0 | S_{i\Delta t} < l\}$ and $\tau_U = \inf\{i\Delta t > 0 | S_{i\Delta t} > m\}$ respectively. The formula

$$\begin{aligned} c_h^I(0,s) &= E[e^{-rI\Delta t} 1_{\{S_{I\Delta t} = su^{2h}\}} 1_{\{l < S_{1\Delta t}, \ldots, S_{I\Delta t} < m\}}] \\ &= e^{-rI\Delta t} Q[S_{I\Delta t} = su^{2h}, l < S_{1\Delta t}, \ldots, S_{I\Delta t} < m] \end{aligned}$$

leads to

$$Q[S_{I\Delta t} = su^{2h}, l < S_{1\Delta t}, \ldots, S_{I\Delta t} < m] = e^{rI\Delta t} c_h^I(0,s).$$

This yields probability $Q[\tau_U \le T, \tau_U \le \tau_L]$ as

$$Q[\tau_U \le T, \tau_U \le \tau_L] = p \sum_{n=M,M+2,\ldots,N'} e^{r(n-1)\Delta t} c_M^{n-1}(0,s) = pe^{-r\Delta t} \sum_{n=M,M+2,\ldots,N'} e^{-r(n-1)\Delta t} e^{rn\Delta t} c_M^{n-1}(0,s), \tag{3.18}$$

where N' is chosen properly so that $N' = N-1$ or $N' = N$ is satisfied. This quantity is derived by computing the price of rebates with an upper cashflow $\phi_M(n) = \phi_L(n) = e^{rn\Delta t}$, given by

$$Q[\tau_U \le T, \tau_U \le \tau_L] = \frac{2p^{\frac{M+2}{2}}}{(R+1)q^{\frac{M}{2}}} \sum_{k=1}^{R} \tilde\lambda_k^M \frac{1-\tilde\lambda_k^{2(N'+1)}}{1-\tilde\lambda_k^2} \sin\left(\frac{Rk\pi}{R+1}\right) \sin\frac{(1-L)k\pi}{R+1}, \tag{3.19}$$

where $\tilde\lambda_k$ is given by $\tilde\lambda_k = e^{r\Delta t}\lambda_k' = (4pq)^{\frac{1}{2}}\lambda_k$.

Similar calculation leads to

$$Q[\tau_L \le T, \tau_L \le \tau_U] = \frac{2e^{-r\Delta t}p^{\frac{L}{2}}}{(R+1)q^{\frac{L-2}{2}}} \sum_{k=1}^{R} \tilde\lambda_k^{-L} \frac{1-\tilde\lambda_k^{2(N'+1)}}{1-\tilde\lambda_k^2} \sin\left(\frac{k\pi}{R+1}\right) \sin\frac{(1-L)k\pi}{R+1}.$$

The one-sided barrier hitting probability $Q[\tau_U \le T]$ can be obtained if we substitute M into $N-1$ in formula (3.18), and probability $Q[\tau_L \le T]$ can be obtained if we substitute L into $1-N$ in formula (3.19).

As a double barrier options case, if M and $-L$ are sufficiently large constants, the approximation $|\lambda_k'|^M \approx 0$ ($|\lambda_k'|^{-L} \approx 0$) is satisfied as long as $|\lambda_k'| \approx 1$ is not satisfied. Then, even if we do not summarize all elements, we can obtain the price of rebates accurately. In our numerical studies to determine the price of rebates, for example, we used only the eigenvalues which satisfied the condition

$$|\lambda_k'|^M < \frac{\epsilon e^{rT}}{(4pq)^{M/2}},$$

where a small constant ϵ is again fixed at 10^{-10}. This condition is equivalent to

$$\log|\lambda'_k| < \frac{rT + \log\epsilon}{M} - \frac{1}{2}\log 4pq.$$

3.4. Pricing knock-out options with a single boundary

One of the simplest ideas to evaluate the price of barrier options with a one-sided lower (resp. upper) knock-out boundary in the binomial tree setting is to equip an artificial upper (resp. lower) boundary. This method enables us to compute the price of barrier options with a one-sided knock-out boundary easily. However, we seek more direct ways to evaluate the price of one-sided knock-out options.

Consider knock-out options with a lower boundary l. We assume that the price of these options is denoted by C. Also, consider double barrier options with the lower and upper barriers fixed at l and $u = se^{N\sigma\sqrt{\Delta t}}$, respectively. The computational method of the initial price of these options is given in Section 2 and is denoted by $C(0,0)(= D(0,0))$. Then the price of one-sided knock-out options, C, is given by

$$C = e^{-rT}q^N\Phi(se^{-N\sigma\sqrt{\Delta t}}) + D(0,0).$$

Similar discussion leads to the price of knock-out options with an upper boundary. There is an important remark for this method; In order to evaluate the price of one-sided barrier options, the weight for the eigenvector, $\mathbf{f_k}$ has to be determined and it is given by a_k. This quantity is computed from the transformed payoff function (3.8),

$$D(N,j) = \left(\frac{p}{q}\right)^{j/2}\Phi(se^{j\sigma\sqrt{\Delta}}).$$

Theoretically, even if the transformed payoff $D(N,j)$ takes a huge value, the price of one-sided barrier options can always be determined. However, if $D(N,j)$ is explosively large, the numerical value of knock-out options can be unstable because of the rounding errors, and we do not recommend evaluating the price of one-sided options in such cases. This means that we do not recommend evaluating the price of down-and-out call options by direct applications of the spectral methods, although the price of down-and-out put options can be computed in this way. In order to calculate the price of down-and-out call options, we use the put-call parity for barrier options. Consider European call and put options with a single lower boundary l. Maturity date for these options is fixed at T and the strike price of these options is given by K. The prices of these call and put options are denoted by C_L and P_L, respectively. Then the put-call parity for knock-out options with a single lower boundary is given by

$$\begin{aligned}
C_L - P_L &= E[e^{-rT}(S_T-K)^+1_{\{\tau_L>T\}}] - E[e^{-rT}(K-S_T)^+1_{\{\tau_L>T\}}]\\
&= E[e^{-rT}(S_T-K)1_{\{\tau_L>T\}}]\\
&= E[e^{-rT}(S_T-K)] - E[e^{-rT}(S_T-K)1_{\{\tau_L\le T\}}]\\
&= s - e^{-rT}K - E[E[e^{-rT}S_T|\mathcal{F}_{\tau_L}]1_{\{\tau_L\le T\}}] + e^{-rT}KE[1_{\{\tau_L\le T\}}]\\
&= s - e^{-rT}K - l'E[e^{-r\tau_L}1_{\{\tau_L\le T\}}] + e^{-rT}KQ[\tau_L\le T],
\end{aligned}$$

where the constant l' is given by $l' = s\exp((L-1)\sigma\sqrt{\Delta t})$. This relation leads to the price of call options with a single lower boundary, given by

$$C_L = P_L + s - e^{-rT}K - l'E[e^{-r\tau_L}1_{\{\tau_L\le T\}}] + e^{-rT}KQ[\tau_L\le T]. \tag{3.20}$$

Note that each quantity in the right-hand side can be safely computed by the spectral method.

4. Numerical studies

The computational results are presented here. We also give the results of the conventional binomial trees and closed form pricing formulas in the continuous time model (Black and Scholes model) for comparison. All results in this section were computed in double precision on a Core i7-920 2.66 GHz PC with 6 GB of RAM. All source codes were written in C++ and complied and optimized with GNU g++ version 4.1.2.

For our numerical demonstrations, the parameters $S = K = 100$, $r = 0.10$, $\sigma = 0.30$, $T = 1$ are used. The boundaries sets are $(L,U) = (90,110), (80,120), (70,130), (60,140)$ and $(50,150)$. The obtained prices for options and rebates will be given to six significant figures.

4.1. Price of double knock-out options

The computational results for double knock-out call options are shown in Table 2. They are computed by the spectral binomial tree approach, the conventional binomial tree approach and two kinds of closed form formulas derived by Kunitomo and Ikeda [2] and Pelsser [4] by changing time to maturity, volatility, and boundary set. Note that totally 21 terms $n = 0, +1, -1, +2, -2, \ldots, +10, -10$ are used for computation for Kunitomo and Ikeda [2] and the proposed truncation condition is employed for Pelsser [4]. We used the same numbers of meshes as in conventional binomial trees.

Table 2
Double knock-out call ($T = 1, \sigma = 0.30$).

L	U	Spectral	Binomial	K–I	Pelsser
50	150	5.43579	5.43579	5.41147	5.41261
60	140	3.28887	3.28887	3.27620	3.27730
70	130	1.46092	1.46092	1.45910	1.46001
80	120	0.232026	0.232026	0.228733	0.229067
90	110	3.50770E−05	3.50770E−05	0.000670253	0.0000306240

K–I: Kunitomo and Ikeda [2], Pelsser: Pelsser [4].

Table 3
Greeks of double knock-out calls ($T = 1, \sigma = 0.30$).

L	U	Spectral with P–V		K–I	
		Delta	Gamma	Delta	Gamma
50	150	0.0427778	−0.00854504	0.0420971	−0.00852582
60	140	−0.00732051	−0.00611914	−0.00752531	−0.00610313
70	130	−0.0202385	−0.00361347	−0.0202341	−0.00361345
80	120	−0.00424989	−0.00130294	−0.00421280	−0.00129129
90	110	−6.19964E−07	−8.37735E−07	3.71721E−06	−8.82349E−07

P–V: Pelsser and Vorst [13], K–I: Kunitomo and Ikeda [2].

As presented in Table 2 and later, there are no differences between the prices computed by the spectral binomial tree approach and the conventional binomial tree approach. On the other hand, the prices are slightly different from the prices derived by the closed formulas in the continuous time model. These results are not very different, even if we calculate the price of double barrier options with other maturity dates and volatilities. In order to get more accurate price, we have to use the binomial trees with finer meshes. This will be addressed in Section 4.4.

Table 3 exhibits the Greeks (delta and gamma) for one-year double knock-out options with the same parameter set as in Table 2. We compute Greeks, delta and gamma, by the extended binomial tree approach of Pelsser and Vorst [13] incorporating with the spectral binomial tree approach. We also give deltas and gammas obtained by taking derivatives directly with respect to the closed form formulas of double barrier options obtained by Kunitomo and Ikeda [2] for comparison.[1] By the extended binomial tree approach of Pelsser and Vorst [13], both the Greeks take closer values to those by Kunitomo and Ikeda [2].

4.2. Price of rebates

Table 4 shows the prices of rebates whose face value F is equal to one dollar. We calculate rebates with a pay-off at "upper boundary and lower boundary". We introduce three kinds of rebates: "upper boundary", "lower boundary" and "upperboundary + lowerboundary". The first one, "upper boundary", shows the prices of rebates that have a cashflow as soon as the underlying asset price process hits the upper boundary, and expire if it hists one of the boundaries. The pricing formula for this security is given by (3.16). The second one, "lower boundary", provides the prices of rebates that have a cashflow as soon as the underlying asset price process hits the lower boundary and expire when it hits one of the boundaries. The pricing formula for this security is given by (3.17). The third one, "upper boundary+lower boundary", shows the prices of rebates that have a cashflow as soon as the underlying asset price hits one of double barriers. The pricing formula for such a security is given by the sum of the prices of the above two securities. The numerical prices using the conventional binomial tree approach and two kinds of the analytic formulas derived by Pelsser [4] and Ikeda [15] are also given for comparison. Since the truncation condition for Pelsser [4] is not given, we used the first 100 terms for calculation. For Ikeda [15], the first six terms, $n = 0, 1, \ldots, 5$, were used.[2]

The prices computed by the spectral binomial tree approach are the same as those computed by the conventional binomial tree approach for all cases. On the other hand, compared to the results of Pelsser [4] and Ikeda [15], the differences are all within 1% range.

4.3. Price of knock-out options with a single boundary

Tables 5 and 6 give the prices of down-and-out and up-and-out options respectively. They are calculated by the spectral binomial tree approach, the conventional binomial tree approach, and the closed form formula. As discussed in the previous

[1] The sign of delta by spectral binomial tree is opposite to that by Kunitomo and Ikeda [2] for the case $l = 90$ and $m = 110$. This is because delta at $s = 100$ is too tiny to evaluate, *i.e.* $\Delta \sim 10^{-6}$, with numerical algorithms.

[2] Ikeda [15] derived the closed form pricing formula for rebates. He showed that the pricing formula for rebates is represented by the infinite weighted sum of normal distribution functions.

Table 4
Rebate at a hit (double barrier: $T = 1, \sigma = 0.30$).

(Upper boundary)					
	U	Spectral	Binomial	Pelsser	Ikeda
	150	0.209506	0.209506	0.210320	0.210334
	140	0.300883	0.300883	0.301493	0.301510
	130	0.422575	0.422575	0.422646	0.422665
	120	0.545151	0.545151	0.545745	0.545760
	110	0.548252	0.548252	0.549574	0.549571
(Lower boundary)					
L		Spectral	Binomial	Pelsser	Ikeda
50		0.0123966	0.0123966	0.0124604	0.0124617
60		0.0596789	0.0596789	0.0597374	0.0597419
70		0.173680	0.173680	0.173815	0.173826
80		0.339430	0.339430	0.339655	0.339666
90		0.440583	0.440583	0.439390	0.439387
(Upper boundary + lower boundary)					
L	U	Spectral	Binomial	Pelsser	Ikeda
50	150	0.221902	0.221902	0.222780	0.222796
60	140	0.360562	0.360562	0.361230	0.361252
70	130	0.596255	0.596255	0.596461	0.596491
80	120	0.884581	0.884581	0.885400	0.885426
90	110	0.988835	0.988835	0.988964	0.988958

Ikeda: Ikeda [15], Pelsser: Pelsser [4].

Table 5
Down-and-out call/put ($T = 1, \sigma = 0.30$). (The prices of call options are derived by put-call-parity.)

(Call)				(Put)			
L	Spectral	Binomial	BS	L	Spectral	Binomial	BS
50	16.7341	16.7341	16.7336	50	6.62401	6.62401	6.62075
60	16.7314	16.7314	16.7309	60	5.00451	5.00451	5.00212
70	16.6380	16.6380	16.6372	70	2.56420	2.56420	2.56119
80	15.6769	15.6769	15.6715	80	0.683487	0.683487	0.679586
90	11.3341	11.3341	11.3147	90	0.0475408	0.0475408	0.0460698

BS: analytic price in the BS model.

Table 6
Up-and-out call/put ($T = 1, \sigma = 0.3$).

(Call)				(Put)			
U	Spectral	Binomial	BS	U	Spectral	Binomial	BS
110	0.0369689	0.0369689	0.0360213	110	4.14834	4.14834	4.12174
120	0.429464	0.429464	0.425557	120	6.11205	6.11205	6.10547
130	1.51894	1.51894	1.51725	130	6.87410	6.87410	6.87334
140	3.29127	3.29127	3.27861	140	7.12440	7.12440	7.12317
150	5.43580	5.43580	5.41148	150	7.19484	7.19484	7.19399

BS: analytic price in the BS model.

section, although we can safely compute the price of up-and-out knock-out call and put options and down-and-out knock-out put options by the spectral binomial tree approach, our algorithm is unstable to compute the price of down-and-out call options because of the rounding errors. Instead, we use the put-call-parity given by (3.20).

The computed prices by discrete time models are likewise the same even if put-call-parity is applied. On the other hand, the largest percentage error between our prices and those by Black–Scholes model is 0.17% for call and 3.19% for put.

4.4. Computing time

We have seen the prices of knock-out options and their relatives in the preceding subsections; We now discuss the differences in CPU time between the spectral binomial tree approach and the conventional binomial tree approach.

Tables 7 and 8 present the CPU times of the spectral binomial tree approach and the conventional binomial tree approach in terms of the number of meshes, *i.e.* Δt, and time to maturity respectively. The CPU times shown in these tables are in units of milliseconds if a time is less than one second, and in seconds otherwise. Note that the array size for the spectral binomial tree approach is about twice as long as that for the conventional binomial tree approach.

Table 7
Computing time for the number of meshes. (Computing time is expressed as 'min′ s″'. Prices are in the lower row.)

No. meshes	10^5	10^6	10^7	10^8	10^9
Spectral	0.002″	0.003″	0.007″	0.018″	0.066″
Binomial	4.405″	2′37″	86′51″	NA[a]	NA[a]
Price	0.235060	0.229631	0.229312	0.229225	0.229112
Error[b] (%)	2.77 (%)	0.39 (%)	0.25 (%)	0.21 (%)	0.16 (%)

$S = K = 100,\ r = 0.10,\ \sigma = 0.30,\ T = 1,\ (L, U) = (80,\ 120)$.
Analytic price (Kunitomo and Ikeda [2]): 0.228733.
[a] Not finished in two days.
[b] Error (%) is obtained by $100 \times \frac{\text{Spectral-KI}}{\text{KI}}$.

Table 8
Computing time for time to maturity. (Computing time is expressed as 'min′ s″'.) Common parameters are: $S = K = 100,\ r = 0.10,\ \sigma = 0.30,\ \Delta t = 1/120,000$.

Maturity	1/12	1/4	1	2	3
a. Double knock-out ($(L,\ U) = (80,\ 120)$)					
Spectral	0.002″	0.002″	0.001″	0.001″	0.001″
Binomial	0.496″	1.500″	5.884″	13.337″	23.449″
b. Down-and-out ($L = 80$)					
Spectral	0.321″	1.580″	12.324″	37.141″	NA[a]
Binomial	7.696″	1′12″	18′33″	83′19″	223′41″
c. Up-and-out ($U = 120$)					
Spectral	0.345″	1.577″	12.300″	34.453″	NA[b]
Binomial	8.489″	1′25″	25′45″	117′50″	262′14″

[a] Out of memory.
[b] Out of memory.

For double knock-out call options, the parameters $S = K = 100, r = 0.10, \sigma = 0.30, T = 1, (L, U) = (80, 120)$ and $\Delta t = 1/10^9$ took us less than 0.1 s to compute using the spectral binomial tree approach. On the other hand, it took more than two days to compute the price of double barrier options using the conventional binomial tree (Table 7). Even if for the length of time step with $\Delta t = 1/10^5$, it takes about five seconds to compute the price of double barrier options using the conventional binomial tree approach only.

Table 7 also exhibits the corresponding price. Compared to the numerical results obtained by Kunitomo and Ikeda [2] and Pelsser [4], the figure seems to become closer as the number of meshes increases, *i.e.* $\Delta t \to 0$. In addition, although the derived prices as a function of the number of meshes for the time direction show non-monotonous decrease and slow convergence toward the prices by Kunitomo and Ikeda [2], they do not become lower than those by continuous time models and the percentage difference becomes 0.2% for $\Delta t = 1/10^9$.

Likewise, although the spectral binomial tree requires more CPU time for the barrier options, in particular single-barrier options with longer maturity dates, it is only less than one minute so long as it is calculable (Table 8). On the other hand, although the conventional binomial tree approach has an advantage in that it needs relatively less memory to obtain the prices for three-year single-barrier options, it took about six hours to finish the computation.[3] To this end, since more memory is needed, the proposed approach may fail to calculate the prices of longer maturity options.[4] However, higher computational abilities would calculate the prices of barrier options easily and quickly.

5. Concluding remarks

We propose the spectral binomial tree approach, a new algorithm for pricing barrier options and their relatives. We also derive the prices of double knock-out options and knock-out options with a single boundary and rebates at a hit. Comparison of computing time between the proposed approach and the conventional binomial tree model is also included in this study.

Our computational results are summarized as follows: First, although the proposed approach requires more memory than the conventional binomial tree methods, it requires much less computing time to obtain the results for all cases. Especially, it is worth noting that less than 0.1 s is needed for the double-barrier call option with one billion meshes for time direction. Second, the obtained prices and rebates are always the same as those derived by the conventional binomial tree approach and quite close to those by continuous time models such as Kunitomo and Ikeda [2] and Pelsser [4]. This is much more apparent as the mesh size for time direction becomes smaller and smaller. Likewise, the same is true for the prices of rebates and

3 The barrier options with long maturity are very popular in the foreign exchange market. Moreover, barrier options with long maturity (several years) are also very important to arrange structural bonds.

4 We could not compute the price of one-sided barrier options because of lack of the memory, if maturity date is longer than two and a quarter years.

single barrier options in spite of that it is hard to calculate the prices of single barrier options with huge meshes for time direction. To summarize, the worth of our technique is, although the trees with a large number of meshes do not increase CPU time dramatically, we are able to have a quite good approximate value for options with barrier(s) and their relatives compared to continuous time models.

While the binomial tree model is easy to handle and expandable, it is time consuming and thus unsuited to practical use. However, we believe that our approach will help overcome such difficulties and be especially meaningful for practical use.

References

[1] R. Merton, Theory of rational option pricing, Bell Journal of Economics and Management Science 4 (1973) 141–183.
[2] N. Kunitomo, M. Ikeda, Pricing options with curved boundaries, Mathematical Finance 2 (1992) 275–298.
[3] H. Geman, M. Yor, Pricing and hedging double barrier options: a probabilistic approach, Mathematical Finance 6 (1996) 365–378.
[4] A. Pelsser, Pricing double barrier options using analytical inversion of Laplace transform, Finance and Stochastics 4 (2000) 95–104.
[5] D. Davydov, V. Linetsky, Pricing options on scalar diffusions: eigenfunction expansion approach, Operations Research 51 (2003) 185–209.
[6] J. Cox, Notes on option pricing I: constant elasticity of diffusions, Unpublished note, Graduate School of Business, Stanford University, 1975.
[7] P.P. Boyle, Y. Tian, Pricing lookback and barrier options under the CEV process, The Journal of Financial and Quantitative Analysis (1999) 241–264.
[8] Y.-D. Lyuu, Very fast algorithms for barrier option pricing and the Ballot problem, Journal of Derivatives 5 (1998) 68–79.
[9] T.-S. Dai, L.-M. Liu, Y.-D. Lyuu, Linear-time option pricing algorithms by combinatorics, Computers and Mathematics with Applications 55 (2008) 2142–2157.
[10] J.C. Cox, S.A. Ross, M. Rubinstein, Option pricing: a simplified approach, Journal of Financial Economics 7 (1979) 229–263.
[11] J. Hull, Options, Futures, and Other Derivatives, seveth ed., Prentice Hall, 2008.
[12] C. Meyer, Matrix Analysis and Applied Linear Algebra, SIAM, 2001.
[13] A. Pelsser, T. Vorst, The binomial model and the Greeks, Journal of Derivatives 1 (1994) 45–49.
[14] D. Davydov, V. Linetsky, The valuation and hedging of barrier options and lookback options under the CEV process, Management Science 47 (2001) 949–965.
[15] M. Ikeda, Theory of Option Valuation and Corporate Finance, University of Tokyo Press, 2000 (in Japanese).