

THE RATE OF CONVERGENCE OF THE TWO-STATE LATTICE MODEL FOR PRICING VANILLA OPTIONS

MARK JOSHI AND CHUN FUNG KWOK

ABSTRACT. Variations of the binomial tree model are reviewed and extensions to the two most efficient trees studied in a recent literature are proposed. Tian's modified tree is extended to a more general class of tree, and the third order tree is extended to the seventh order tree. Analysis of the error of American put pricing in the binomial tree model is presented and new trees that have superior performance in pricing in-the-money American puts are developed. To further improve numerical results, a scheme that incorporates different trees is suggested.

1. INTRODUCTION

The Binomial Tree Method is a popular numerical method for pricing derivative products. It is well known for its simplicity and its capability to deal with derivatives which have early exercise features. Since 1979 in which Cox-Ross-Rubinstein, and Rendleman-Bartter independently introduced the binomial tree model and laid down the main framework, many modifications have been proposed. The motivation for the changes is that while the price computed under the original tree model does converge to the correct theoretical price, the convergence is rather slow and unstable. Therefore, new tree models were introduced to attempt to improve the convergence.

Roughly speaking, the developments of the tree method can be categorized into three groups. The first group is "Modifications of conditions on parameters". We start with one major modification that is adopted in most papers on tree method subsequent to CRR and RB's seminal work. It is to match moments exactly rather than asymptotically. Both Cox-Ross-Rubinstein(1979) and Rendleman-Bartter(1979) in their works specified the moment matching conditions in the way that the moments of the binomial tree tend toward those of the continuous process when the number of steps used for the discretization, N , goes to infinity. Instead of matching the moments asymptotically, it was proposed to match them exactly. In other words, the moment matching conditions are specified so that the moments of the tree model match with those of the continuous process for all N . Now we proceed to review some other modifications and highlight the characteristics of these trees in the first group. In general, as it is summarized in Joshi(2011, p.425), the constructions of the new trees differ mainly in the choices of which process to discretize, how many and how the moments are matched, and whether or not to impose symmetry on the tree. In particular, Trigeorgis(1991) matched the first two raw moments exactly in log space, and imposed symmetry on the movement sizes in log space. Jarrow-Rudd(1993) discretized the log risk-neutral stock process and imposed symmetry on the movement probabilities. Tian(1993) suggested to match the first three raw moments exactly in spot space. Chriss(1996)

discretized the real-world stock process and imposed symmetry on the probabilities of the movements. In addition, the first raw moment is matched in spot space exactly, and the second central moment is matched in log space exactly. The tree is grown along the forward. Wilmott(1998) derived a model with two variations. The base model matched the first two raw moments exactly in spot space. The first variation is to impose symmetry on the movement sizes in log space, and the second variation is to impose symmetry on the movement probabilities. Jabbour, Kramin and Young (2001) introduced three models, the extended RB, the ABMC and the ABMD tree. For the extended RB tree, the first two moments are matched exactly in log space and the tree is grown along the forward. For the ABMC tree, the first two moments are matched exactly in spot space. Again, the tree is grown along the forward. For the ABMD tree, it assumes the underlying follows a discrete geometric Brownian motion process instead of a continuous one, then the first two moments are matched exactly in spot space. There are three variations to it. One, impose symmetry on the movement size in the log space. Two, grow tree along the forward. Three, impose symmetry on the movement probabilities. Joshi(2004) discretized the real stock process and constructed a tree centered on the strike in log space.

The second group is “Design of trees with special features”. Tian(1999) incorporated a tilt parameter into the model so that a node in the tree coincides with the strike price at the maturity of the option. This gives a smooth convergence so that acceleration technique (Richardson extrapolation) can be applied. Chang and Palmer(2006) proposed a similar idea, they constructed the tree so that the strike is at the geometric average of two nodes, i.e. the midpoint of two nodes in log space. Andricopoulos, Duck, Newton and Widdicks(2002) attacked the tree problem differently. They introduced a way to choose N such that their tree has a monotonic convergence, then they apply Richardson extrapolation. Gaudenzi and Pressacco(2003) developed a tree based on interpolations over a range of modified strikes. Moreover, they described a technique how options with different strikes can be priced on the same tree. Leisen and Reimer(1996) constructed the tree in a non-traditional way. The movement probabilities in the bond measure and the stock measure are first specified, then from there movement sizes are determined. In addition, the tree was constructed such that the strike is at the center of the tree. This tree is the first of its kind that has a proven second order convergence for pricing European options. Joshi(2010) developed this idea further and showed that trees of arbitrarily high order exist. Explicit construction of a third order tree and guidelines to construction of higher order trees were given. While the tree only worked with odd number of steps, the extension to even number of steps was done by Xiao(2010). Multinomial Trees, trees that have more than two states in each time step, were introduced and developed(Boyle 1988, Boyle, Evnine and Gibbs 1989, Kamrad and Ritchken 1991, Omberg 1988, Parkinson 1987, Tian 1993). In particular, trinomial tree received a lot of attention as it is the natural extension to the binomial tree. Yet, Chan, Joshi, Tang and Yang(2008) conducted a comparison of eight trinomial trees and one binomial tree with and without acceleration techniques, and found out that the binomial tree with acceleration techniques applied performed the best.

The third group is the “Introduction of acceleration techniques”. Hull and White(1988) applied the ‘Control Variate’ technique to binomial trees. The idea is to use European options as a control to American options, so the American option price is adjusted by the error we get when pricing European options. Broadie and Detemple(1996) introduced the ‘Smoothing’ technique. It is to replace the continuation value one time step before the maturity with the Black-Scholes price. The motivation is that the Black-Scholes price has a continuous derivative, while the continuation value does not. With such replacement, the tree price converges with much less oscillations and more smoothly. In the same paper, Broadie and Detemple(1996) also introduced another acceleration technique, the ‘Richardson Extrapolation’, to enhance their numerical results further. By doing an appropriate extrapolation, part of the error can be removed. Figlewski and Gao(1999) developed the Adaptive Mesh Model. This model reduces nonlinearity error by using finer mesh at some selected stages. It gives a reasonably better accuracy at a cost of a little computation time. Andricopoulos, Duck, Newton and Widdicks(2004) proposed the ‘Truncation’ technique to speed up the computation. The technique truncates the computation range according to the number of standard derivations from the expected value. The justification is that it is very unlikely for the underlying to reach the extreme values, so the contributions of those nodes are so small that can be ignored. Staunton(2005) suggested to use six standard derivations from the expected value for the computation range. Based on the truncation idea, Chen and Joshi(2010) invented a new method, the tolerance method, for deciding the truncation range. The range is chosen based on comparison between the time value and the tolerance level at each node. This can give a 50% better performance than the original method.

We have focused on the construction aspect of the tree development, now we discuss the performance aspect. One way to evaluate the efficiency of a tree is to use the notion of “order of convergence”. A tree is of i -th order or has a i -th order of convergence if the price error in the N -step tree is $O(N^{-i})$, where O is the big-O notation. In the context of American puts pricing, only a few trees have their order of convergence proven theoretically. In particular, Lamberton(1998) proved the Cox-Ross-Rubinstein(1979) tree converges with order three-quarters. Leisen(1998) improved the result and showed that the Cox-Ross-Rubinstein(1979) tree converges with order one. It has also been proven that the Jarrow-Rudd(1983) tree and the Tian(1993) tree converge with order between half and one. Yet, numerically speaking, with the introduction of acceleration techniques, the tree method in general has an order of convergence between first and second order. For more details, one can consult Chance(2008) and Joshi(2009), where comparisons on the convergence of more than 20 trees were conducted. Here we simply state the conclusions of the two papers. Chance(2008) examined 11 trees and concluded there is no one tree that performs consistently better than the other ten trees. On the other hand, Joshi(2009) examined 11 trees (with very little overlap with Chance(2008)) with and without acceleration techniques and concluded that the Tian(1993) third moment matching tree with smoothing and extrapolation, and the Joshi(2010) higher order tree with extrapolation perform the best.

We discuss the structure of this paper. In Section 2, we define the binomial tree model framework. In Section 3, we investigate two existing trees and give extensions to them. In Section 4, we study the convergence of American put options that helps us understand the nature of the oscillation in American puts pricing. In Section 5, we develop new trees based on our analysis. In Section 6, we present numerical results of the trees. In Section 7, we answer the question, “which tree does best in which region?” and use it to enhance our numerical results. We conclude in Section 8.

We adopt the following notation for the rest of this paper.

S_t : Stock price at time t

K : Strike price of the option

T : Time to maturity of the option(in years)

r : Risk-free interest rate

σ : Volatility of the stock

N : Number of steps in the tree

Δt : The time length of a period

Tree parameters that need to be specified:

p : The probability of an up movement

u : The stock price multiplier for an up movement

d : The stock price multiplier for a down movement.

2. THE BINOMIAL TREE MODEL FRAMEWORK

Here we define the framework we are going to work under. To develop the binomial tree model, we need to first make an assumption on the process the stock follows. As the binomial tree model is developed based on the Black-Scholes model, we use an assumption that is in line with the Black-Scholes model. Under the Black-Scholes model, the stock price in the risk-neutral measure is assumed to follow the stochastic process $dS_t = rS_t dt + \sigma S_t dW_t$, where W_t is the standard Brownian motion. Applying Ito’s formula, it can be shown that

$$d \log S_t = \left(r - \frac{\sigma^2}{2} \right) dt + \sigma dW_t$$

Then solving the stochastic differential equation gives

$$S_t = S_0 \exp \left(\left(r - \frac{\sigma^2}{2} \right) t + \sigma W_t \right)$$

The basic idea of the binomial tree model is to discretize this continuous process. There are two parameters we need to discretize, the time and the states. In a N -step binomial tree model, we divides T into N periods, and we have $\Delta t = T/N$. During the i -th period (for $i = 1, 2, \dots, N$), there are $i+1$ possible states (in contrast to the continuous case where price takes values from a continuous spectrum). More concretely, we have the following discretization of the process:

$$S_i = S_0 \prod_{k=1}^i X_k$$

where X_k 's are independent and identical random variables taking value u, d with probabilities $p, (1 - p)$ respectively. If we rewrite it as

$$S_i = S_0 \exp \left(\sum_{k=1}^i \log \left(\frac{u}{d} \right) Y_k + N \log d \right)$$

where Y_k 's are Bernoulli distributions, then we see that essentially, the discretization is just about approximating the normal distribution with a binomial distribution.

To ensure that the price computed under the binomial tree model does indeed converge to the right price when we let N goes to infinity, we require to match the first two moments of the binomial tree model to those of the continuous process. As we suggested in the introduction section, there are many ways to achieve that. Here we give an example. Suppose we want to match moments exactly in spot space for each time step, then we would need:

$$\begin{aligned} pu + (1 - p)d &= e^{r\Delta t} \\ pu^2 + (1 - p)d^2 &= e^{(2r+\sigma^2)\Delta t} \end{aligned}$$

From here, for the European case, it follows from Lyapunov's variant of the central limit theorem that the binomial distribution converges to the normal distribution, hence the price under the binomial tree model does converge to the right price. For a more technical proof, the interested reader can consult Hsia(1983). For the American case, the first two moments matching condition makes sure our discretization converge weakly to the continuous process. It then follows from Amin and Khanna's(1994) rigorous proof that the price under the binomial tree model does converge to the right price.

For the actual implementation of the binomial tree model, we would need to explicitly compute u, d and p . While we have three parameters, the first two moments conditions only give us two equations. Hence there is some flexibility where we can choose an appropriate third condition to improve the convergence. This is the central idea of improving the convergence of the binomial tree model.

3. STUDIES ON EXISTING TREES

3.1. Tian's modified binomial model. Tian(1993) proposed a modified binomial model that matches the first three raw moments exactly in spot space for each step. The conditions are:

$$\begin{aligned} (1) \quad & pu + (1 - p)d = M_1 \\ (2) \quad & pu^2 + (1 - p)d^2 = M_2 \\ (3) \quad & pu^3 + (1 - p)d^3 = M_3 \end{aligned}$$

where $M_1 = e^{r\Delta t}$, $M_2 = e^{(2r+\sigma^2)\Delta t}$, $M_3 = e^{(3r+3\sigma^2)\Delta t}$.

Solving the system of equations, we have tree parameters:

$$\begin{aligned} u &= \frac{e^{(r+\sigma^2)\Delta t}}{2} \left[e^{\sigma^2\Delta t} + 1 + \sqrt{e^{2\sigma^2\Delta t} + 2e^{\sigma^2\Delta t} - 3} \right], \\ d &= \frac{e^{(r+\sigma^2)\Delta t}}{2} \left[e^{\sigma^2\Delta t} + 1 - \sqrt{e^{2\sigma^2\Delta t} + 2e^{\sigma^2\Delta t} - 3} \right], \\ p &= \frac{e^{r\Delta t} - d}{u - d}. \end{aligned}$$

3.2. Extension to Tian's modified binomial model. We will show that Tian's modified binomial tree is a special case of a more general class of tree. We consider a tree model that matches the first two raw moments exactly in spot space for each step and it satisfies the condition $ud = W$. That is to say, we have the following conditions for the tree parameters u, d and p :

$$\begin{aligned} pu + (1-p)d &= M_1 \\ pu^2 + (1-p)d^2 &= M_2 \\ ud &= W \end{aligned}$$

If we let $W = (M_2/M_1)^2$ and solve the system, we get exactly the same solution set (u, d, p) as in Tian's model. Therefore, the Tian tree is indeed a special case of our tree above. We call this general class of tree the 'First two moments matching and Center node specifying' class. The reason is that if we restrict N to be even, write $N = 2k$, then it follows that $S_0 u^k d^k = S_0 W^k$ is the position of the center node. Hence, the condition $ud = W$ is in fact a direct specification of the center node position.

Note that the motivation of the Tian's modified binomial tree is to match the first three moments to the continuous process. In our general tree model, if we choose a different center node position, the tree would no longer be first-three-moment matching. Yet, our point here is that while one particular choice of center node position has additional meaning, the Tian tree actually belongs to a larger class of tree. And we will see later there is actually a better choice of the center node position.

3.3. Higher order tree. Joshi(2010) proposed a new class of trees that has a high order of convergence in pricing European options. The construction is given by matching the first moment exactly in spot space for each time step and specifying the probabilities such that the prices for the digital option and the asset-or-nothing option are correct to high order. This tree has two special features. First, it places the center node of the tree at the strike asymptotically. Second, it has smooth error asymptotics in pricing European options.

Here is the specification of the parameters.

$$p = \frac{1}{2} + \frac{d_2}{2\sqrt{2}} \frac{1}{\sqrt{k}} + \sum_{m=1}^{\infty} \frac{\alpha_m}{\sqrt{k}^m}, \quad u = e^{r\Delta t} \frac{p'}{p}, \quad d = \frac{e^{r\Delta t} - p^2 u}{1 - p^2}.$$

where

$$p' = \frac{1}{2} + \frac{d_1}{2\sqrt{2}} \frac{1}{\sqrt{k}} + \sum_{m=1}^{\infty} \frac{\beta_m}{\sqrt{k^m}},$$

$$d_1 = \frac{(r + \frac{1}{2}\sigma^2)T + \log(S/K)}{\sigma\sqrt{T}},$$

$$d_2 = \frac{(r - \frac{1}{2}\sigma^2)T + \log(S/K)}{\sigma\sqrt{T}}.$$

From the construction, it may not be immediately clear why this tree converges in pricing American puts. Here we give an explanation. Consider the second moment of the model:

$$(pu^2 + (1-p)d^2)^N = \left(e^{r\Delta t} \frac{p'^2}{p} + \frac{(1-p')^2}{1-p} \right)^{2k+1}$$

Expanding it in an asymptotic series, we have:

$$\exp(2rT) \cdot \exp(d_2 - d_1)^2 + O\left(\frac{1}{k}\right)$$

Substituting d_1 and d_2 , then as k goes to infinity, we have $e^{(2r+\sigma^2)T}$. Therefore, the tree actually has the first two moments matched to the continuous process. The first moment is matched in spot space exactly as mentioned before, and the second moment is matched in spot space asymptotically as we have just shown. It then follows from the last part of the ‘framework’ section that the tree does converge to the correct price for American options.

3.4. Extension to the higher order tree. Joshi(2010) gave explicit constructions of a third order tree and an almost fourth order tree. We here extend the result to seventh order. Note that before Joshi(2010), the highest order of convergence that had ever been achieved was second order. And our seventh order tree is N^5 times computationally more efficient than the second order. Take $N = 10$ as an example, our tree would have a pricing error of around 0.0000001, while a second order tree would have a pricing error of around 0.01. An illustration of a performance comparison with other trees is given in figure 1. The derivation of our seventh order tree is similar to that of third order tree in Joshi(2010), however, the additional high order terms have to be carefully chosen in each step of the proof so that the correct coefficients can be found at the end. Here we simply state the final expression.

Denote the standard normal cumulative distribution function by $N(\cdot)$, and the incomplete binomial sum

$$\sum_{j=k}^N \binom{N}{j} p^j (1-p)^{N-j}$$

by $\Phi(k; N; p)$.

Theorem 3.1. If $p = \frac{1}{2} + \frac{1}{\sqrt{k}} \sum_{m=0}^{\infty} \alpha_m k^{-m}$, then

$$\Phi(2k+1; k+1; p) = N(2\sqrt{2}\alpha_0) + \frac{1}{k}e_1 + \frac{1}{k^2}e_2 + \frac{1}{k^3}e_3 + \frac{1}{k^4}e_4 + \frac{1}{k^5}e_5 + \frac{1}{k^6}e_6 + \dots$$

where

$$\begin{aligned} e_1 &= \frac{2}{\sqrt{\pi}} e^{-4\alpha_0^2} \left(\alpha_0^3 + \frac{3}{8}\alpha_0 + \alpha_1 \right), \\ e_2 &= \frac{2}{\sqrt{\pi}} e^{-4\alpha_0^2} \left(\alpha_2 + \frac{3}{8}\alpha_1 - \frac{7}{128}\alpha_0 - 4\alpha_0\alpha_1^2 - \frac{7}{48}\alpha_0^3 - 8\alpha_0^4\alpha_1 - \frac{5}{6}\alpha_0^5 - 4\alpha_0^7 \right), \\ e_3 &= \frac{2}{\sqrt{\pi}} e^{-4\alpha_0^2} \left(\alpha_3 + \frac{3}{8}\alpha_2 - \frac{7}{128}\alpha_1 - \frac{4}{3}\alpha_1^3 + \frac{9}{1024}\alpha_0 - 8\alpha_0\alpha_1\alpha_2 - \frac{3}{2}\alpha_0\alpha_1^2 + \frac{32}{3}\alpha_0^2\alpha_1^3 + \frac{3}{128}\alpha_0^3 \right. \\ &\quad \left. - 16\alpha_0^3\alpha_1^2 - 8\alpha_0^4\alpha_2 - 3\alpha_0^4\alpha_1 + \frac{\alpha_0^5}{8} + 32\alpha_0^5\alpha_1^2 - \frac{64}{3}\alpha_0^6\alpha_1 - \alpha_0^7 + 32\alpha_0^8\alpha_1 - \frac{20}{3}\alpha_0^9 + \frac{32}{3}\alpha_0^{11} \right), \\ e_4 &= \frac{2}{\sqrt{\pi}} e^{-4\alpha_0^2} \left(\alpha_4 + \frac{3}{8}\alpha_3 - \frac{7}{128}\alpha_2 + \frac{9}{1024}\alpha_1 - 4\alpha_1^2\alpha_2 - \frac{\alpha_1^3}{2} + \frac{59}{32768}\alpha_0 - 4\alpha_0\alpha_2^2 \right. \\ &\quad \left. - 8\alpha_0\alpha_1\alpha_3 - 3\alpha_0\alpha_1\alpha_2 + \frac{7}{32}\alpha_0\alpha_1^2 + 8\alpha_0\alpha_1^4 + 32\alpha_0^2\alpha_1^2\alpha_2 - 12\alpha_0^2\alpha_1^3 + \frac{59}{12288}\alpha_0^3 - 32\alpha_0^3\alpha_1\alpha_2 \right. \\ &\quad \left. - 6\alpha_0^3\alpha_1^2 - \frac{64}{3}\alpha_0^3\alpha_1^4 - 8\alpha_0^4\alpha_3 - 3\alpha_0^4\alpha_2 + \frac{7}{16}\alpha_0^4\alpha_1 + 96\alpha_0^4\alpha_1^3 - \frac{49}{7680}\alpha_0^5 + 64\alpha_0^5\alpha_1\alpha_2 - 52\alpha_0^5\alpha_1^2 \right. \\ &\quad \left. - \frac{64}{3}\alpha_0^6\alpha_2 - 8\alpha_0^6\alpha_1 - \frac{256}{3}\alpha_0^6\alpha_1^3 + \frac{51}{320}\alpha_0^7 + \frac{640}{3}\alpha_0^7\alpha_1^2 + 32\alpha_0^8\alpha_2 - 52\alpha_0^8\alpha_1 - \frac{979}{360}\alpha_0^9 - 128\alpha_0^9\alpha_1^2 \right. \\ &\quad \left. + \frac{512}{3}\alpha_0^{10}\alpha_1 - \frac{133}{9}\alpha_0^{11} - \frac{256}{3}\alpha_0^{12}\alpha_1 + \frac{136}{3}\alpha_0^{13} - \frac{64}{3}\alpha_0^{15} \right), \\ e_5 &= \frac{2}{\sqrt{\pi}} e^{-4\alpha_0^2} \left(-\frac{483}{262144}\alpha_0 - \frac{11}{1024}\alpha_0^5 + \frac{9}{1024}\alpha_2 + \frac{59}{32768}\alpha_1 - \frac{5}{128}\alpha_0^7 - \frac{161}{32768}\alpha_0^3 \right. \\ &\quad \left. - \frac{9}{256}\alpha_0\alpha_1^2 - \frac{9}{128}\alpha_0^4\alpha_1 - \frac{7}{128}\alpha_3 + \frac{7}{96}\alpha_1^3 + \frac{37}{96}\alpha_0^9 - \frac{151}{20}\alpha_0^{11} - 12\alpha_0^3\alpha_1\alpha_2 - 3\alpha_0\alpha_1\alpha_3 \right. \\ &\quad \left. - 36\alpha_0^2\alpha_1^2\alpha_2 + \frac{3}{8}\alpha_4 - 104\alpha_0^5\alpha_1\alpha_2 - \frac{704}{5}\alpha_0^{10}\alpha_1 - 176\alpha_0^7\alpha_1^2 - 8\alpha_0\alpha_1\alpha_4 - \frac{532}{15}\alpha_0^{13} - \frac{3}{2}\alpha_0\alpha_2^2 \right. \\ &\quad \left. + \frac{7}{16}\alpha_0^4\alpha_2 + \frac{7}{8}\alpha_0^3\alpha_1^2 + \frac{7}{6}\alpha_0^6\alpha_1 - \frac{79}{12}\alpha_0^2\alpha_1^3 - \frac{103}{4}\alpha_0^5\alpha_1^2 - \frac{103}{4}\alpha_0^8\alpha_1 + 32\alpha_0\alpha_1^3\alpha_2 + 120\alpha_0^3\alpha_1^4 \right. \\ &\quad \left. + \frac{5728}{9}\alpha_0^6\alpha_1^3 + \frac{3184}{3}\alpha_0^9\alpha_1^2 - 52\alpha_0^8\alpha_2 - \frac{212}{3}\alpha_0^4\alpha_1^3 - 5\alpha_0\alpha_1^4 - 3\alpha_0^4\alpha_3 + 32\alpha_0^2\alpha_1\alpha_2^2 + 32\alpha_0^2\alpha_1^2\alpha_3 \right. \\ &\quad \left. - 8\alpha_0\alpha_2\alpha_3 + \frac{6368}{9}\alpha_0^{12}\alpha_1 + \frac{7}{16}\alpha_0\alpha_1\alpha_2 - \frac{3}{2}\alpha_1^2\alpha_2 - 8\alpha_0^6\alpha_2 - 4\alpha_1^2\alpha_3 - \frac{128}{5}\alpha_0^2\alpha_1^5 \right. \\ &\quad \left. - 16\alpha_0^3\alpha_2^2 - 8\alpha_0^4\alpha_4 + \frac{512}{15}\alpha_0^4\alpha_1^5 + 32\alpha_0^5\alpha_2^2 - 320\alpha_0^5\alpha_1^4 - \frac{64}{3}\alpha_0^6\alpha_3 + \frac{512}{3}\alpha_0^7\alpha_1^4 + 32\alpha_0^8\alpha_3 \right. \\ &\quad \left. - \frac{8576}{9}\alpha_0^8\alpha_1^3 + \frac{512}{3}\alpha_0^{10}\alpha_2 + \frac{1024}{3}\alpha_0^{10}\alpha_1^3 - \frac{3584}{3}\alpha_0^{11}\alpha_1^2 - \frac{256}{3}\alpha_0^{12}\alpha_2 + \frac{1024}{3}\alpha_0^{13}\alpha_1^2 - \frac{2048}{3}\alpha_0^{14}\alpha_1 \right. \\ &\quad \left. + \frac{512}{3}\alpha_0^{16}\alpha_1 - 4\alpha_1\alpha_2^2 + \alpha_5 + \frac{8}{5}\alpha_1^5 - \frac{6592}{45}\alpha_0^{17} + \frac{512}{15}\alpha_0^{19} - 32\alpha_0^3\alpha_1\alpha_3 - \frac{256}{3}\alpha_0^3\alpha_1^3\alpha_2 + 288\alpha_0^4\alpha_1^2\alpha_2 \right. \\ &\quad \left. + 64\alpha_0^5\alpha_1\alpha_3 - 256\alpha_0^6\alpha_1^2\alpha_2 + \frac{1280}{3}\alpha_0^7\alpha_1\alpha_2 - 256\alpha_0^9\alpha_1\alpha_2 + \frac{2464}{15}\alpha_0^{15} \right), \end{aligned}$$

$$\begin{aligned}
e_6 = & \frac{2}{\sqrt{\pi}} e^{-4\alpha_0^2} \left(-\frac{2323}{4194304} \alpha_0 + \frac{115}{196608} \alpha_0^5 + \frac{59}{32768} \alpha_2 - \frac{483}{262144} \alpha_1 - \frac{829}{172032} \alpha_0^7 \right. \\
& - \frac{2323}{1572864} \alpha_0^3 - \frac{59}{8192} \alpha_0 \alpha_1^2 - \frac{59}{4096} \alpha_0^4 \alpha_1 + \alpha_6 + \frac{9}{1024} \alpha_3 - \frac{3}{256} \alpha_1^3 - \frac{5843}{96768} \alpha_0^9 + \frac{67171}{60480} \alpha_0^{11} \\
& + \frac{7}{4} \alpha_0^3 \alpha_1 \alpha_2 + \frac{7}{16} \alpha_0 \alpha_1 \alpha_3 - \frac{79}{4} \alpha_0^2 \alpha_1^2 \alpha_2 - \frac{7}{128} \alpha_4 - \frac{103}{2} \alpha_0^5 \alpha_1 \alpha_2 - \frac{1292}{15} \alpha_0^{10} \alpha_1 - \frac{323}{3} \alpha_0^7 \alpha_1^2 \\
& - 3 \alpha_0 \alpha_1 \alpha_4 - \frac{167663}{7560} \alpha_0^{13} + \frac{7}{32} \alpha_0 \alpha_2^2 - \frac{9}{128} \alpha_0^4 \alpha_2 - \frac{9}{64} \alpha_0^3 \alpha_1^2 - \frac{3}{16} \alpha_0^6 \alpha_1 + \frac{31}{32} \alpha_0^2 \alpha_1^3 \\
& + \frac{121}{32} \alpha_0^5 \alpha_1^2 + \frac{121}{32} \alpha_0^8 \alpha_1 - 20 \alpha_0 \alpha_1^3 \alpha_2 - \frac{115}{2} \alpha_0^3 \alpha_1^4 - 342 \alpha_0^6 \alpha_1^3 - 601 \alpha_0^9 \alpha_1^2 - \frac{103}{4} \alpha_0^8 \alpha_2 \\
& - \frac{181}{4} \alpha_0^4 \alpha_1^3 - \frac{55}{16} \alpha_0 \alpha_1^4 + 64 \alpha_0^2 \alpha_1 \alpha_2 \alpha_3 - 4 \alpha_1^2 \alpha_4 + 8 \alpha_1^4 \alpha_2 - 4 \alpha_0 \alpha_3^2 - \frac{32}{3} \alpha_0 \alpha_1^6 + \frac{32}{3} \alpha_0^2 \alpha_1^3 \\
& + \frac{7}{16} \alpha_0^4 \alpha_3 - 36 \alpha_0^2 \alpha_1 \alpha_2^2 - 36 \alpha_0^2 \alpha_1^2 \alpha_3 - 3 \alpha_0 \alpha_2 \alpha_3 + \frac{512}{9} \alpha_0^3 \alpha_1^6 - 8 \alpha_0^4 \alpha_5 - \frac{2048}{45} \alpha_0^5 \alpha_1^6 - \frac{64}{3} \alpha_0^6 \alpha_4 \\
& + \frac{11264}{15} \alpha_0^6 \alpha_1^5 + \frac{640}{3} \alpha_0^7 \alpha_2^2 + 32 \alpha_0^8 \alpha_4 - \frac{4096}{15} \alpha_0^8 \alpha_1^5 - 128 \alpha_0^9 \alpha_2^2 + \frac{24832}{9} \alpha_0^9 \alpha_1^4 + \frac{512}{3} \alpha_0^{10} \alpha_3 \\
& - \frac{2048}{3} \alpha_0^{11} \alpha_1^4 - \frac{256}{3} \alpha_0^{12} \alpha_3 + \frac{41984}{9} \alpha_0^{12} \alpha_1^3 - \frac{2048}{3} \alpha_0^{14} \alpha_2 - \frac{8192}{9} \alpha_0^{14} \alpha_1^3 + 4096 \alpha_0^{15} \alpha_1^2 \\
& + \frac{512}{3} \alpha_0^{16} \alpha_2 - \frac{2048}{3} \alpha_0^{17} \alpha_1^2 + \frac{16384}{9} \alpha_0^{18} \alpha_1 - \frac{4096}{15} \alpha_0^{20} \alpha_1 + \frac{4864}{15} \alpha_0^{21} - \frac{2048}{45} \alpha_0^{23} - \frac{1202}{3} \alpha_0^{12} \alpha_1 \\
& - \frac{9}{128} \alpha_0 \alpha_1 \alpha_2 + \frac{7}{32} \alpha_1^2 \alpha_2 + 7/6 \alpha_0^6 \alpha_2 - 3/2 \alpha_1^2 \alpha_3 + 80 \alpha_0^2 \alpha_1^5 - 6 \alpha_0^3 \alpha_2^2 - 3 \alpha_0^4 \alpha_4 - 512 \alpha_0^4 \alpha_1^5 \\
& - 52 \alpha_0^5 \alpha_2^2 + 1096 \alpha_0^5 \alpha_1^4 - 8 \alpha_0^6 \alpha_3 - \frac{9536}{3} \alpha_0^7 \alpha_1^4 - 52 \alpha_0^8 \alpha_3 + \frac{10960}{3} \alpha_0^8 \alpha_1^3 - \frac{704}{5} \alpha_0^{10} \alpha_2 \\
& - \frac{21632}{3} \alpha_0^{10} \alpha_1^3 + \frac{72128}{15} \alpha_0^{11} \alpha_1^2 + \frac{6368}{9} \alpha_0^{12} \alpha_2 - \frac{68480}{9} \alpha_0^{13} \alpha_1^2 + \frac{41216}{15} \alpha_0^{14} \alpha_1 - \frac{34240}{9} \alpha_0^{16} \alpha_1 \\
& - \frac{3}{2} \alpha_1 \alpha_2^2 + \frac{3}{8} \alpha_5 - \alpha_1^5 + \frac{230996}{405} \alpha_0^{17} - \frac{6560}{9} \alpha_0^{19} - 12 \alpha_0^3 \alpha_1 \alpha_3 + 480 \alpha_0^3 \alpha_1^3 \alpha_2 - 212 \alpha_0^4 \alpha_1^2 \alpha_2 \\
& - 104 \alpha_0^5 \alpha_1 \alpha_3 + \frac{5728}{3} \alpha_0^6 \alpha_1^2 \alpha_2 - 352 \alpha_0^7 \alpha_1 \alpha_2 + \frac{6368}{3} \alpha_0^9 \alpha_1 \alpha_2 - \frac{4}{3} \alpha_2^3 - 8 \alpha_1 \alpha_2 \alpha_3 \\
& - 8 \alpha_0 \alpha_2 \alpha_4 - 8 \alpha_0 \alpha_1 \alpha_5 + 48 \alpha_0 \alpha_1^2 \alpha_2^2 + 32 \alpha_0 \alpha_1^3 \alpha_3 + 32 \alpha_0^2 \alpha_1^2 \alpha_4 - 128 \alpha_0^2 \alpha_1^4 \alpha_2 - 32 \alpha_0^3 \alpha_2 \alpha_3 \\
& - 32 \alpha_0^3 \alpha_1 \alpha_4 - 128 \alpha_0^3 \alpha_1^2 \alpha_2^2 - \frac{256}{3} \alpha_0^3 \alpha_1^3 \alpha_3 + 288 \alpha_0^4 \alpha_1 \alpha_2^2 + 288 \alpha_0^4 \alpha_1^2 \alpha_3 + \frac{512}{3} \alpha_0^4 \alpha_1^4 \alpha_2 + 64 \alpha_0^5 \alpha_2 \alpha_3 \\
& + 64 \alpha_0^5 \alpha_1 \alpha_4 - 1280 \alpha_0^5 \alpha_1^3 \alpha_2 - 256 \alpha_0^6 \alpha_1 \alpha_2^2 - 256 \alpha_0^6 \alpha_1^2 \alpha_3 + \frac{1280}{3} \alpha_0^7 \alpha_1 \alpha_3 + \frac{2048}{3} \alpha_0^7 \alpha_1^3 \alpha_2 \\
& \left. - \frac{8576}{3} \alpha_0^8 \alpha_1^2 \alpha_2 - 256 \alpha_0^9 \alpha_1 \alpha_3 + 1024 \alpha_0^{10} \alpha_1^2 \alpha_2 - \frac{7168}{3} \alpha_0^{11} \alpha_1 \alpha_2 + \frac{2048}{3} \alpha_0^{13} \alpha_1 \alpha_2 - \frac{7313}{81} \alpha_0^{15} \right).
\end{aligned}$$

Corollary 3.2. *If $p = \frac{1}{2} + \frac{1}{\sqrt{k}} \sum_{m=0}^6 \alpha_m k^{-m}$, where*

$$\alpha_0 = \frac{d_i}{2\sqrt{2}}, \quad \alpha_1 = -\alpha_0^3 - \frac{3}{8}\alpha_0, \quad \alpha_2 = \frac{25}{128}\alpha_0 + \frac{13}{12}\alpha_0^3 + \frac{5}{6}\alpha_0^5,$$

$$\alpha_3 = -\frac{105}{1024}\alpha_0 - \frac{119}{128}\alpha_0^3 - \frac{23}{16}\alpha_0^5 - \frac{1}{2}\alpha_0^7,$$

$$\alpha_4 = \frac{1659}{32768}\alpha_0 + \frac{361}{512}\alpha_0^3 + \frac{6407}{3840}\alpha_0^5 + \frac{103}{90}\alpha_0^7 + \frac{79}{360}\alpha_0^9,$$

$$\alpha_5 = -\frac{6237}{262144}\alpha_0 - \frac{16071}{32768}\alpha_0^3 - \frac{16363}{10240}\alpha_0^5 - \frac{6263}{3840}\alpha_0^7 - \frac{587}{960}\alpha_0^9 - \frac{3}{40}\alpha_0^{11},$$

$$\alpha_6 = \frac{50765}{4194304}\alpha_0 + \frac{42377}{131072}\alpha_0^3 + \frac{1336991}{983040}\alpha_0^5 + \frac{148651}{80640}\alpha_0^7 + \frac{2959421}{2903040}\alpha_0^9 + \frac{22573}{92720}\alpha_0^{11} + \frac{71}{3024}\alpha_0^{13}.$$

then $\Phi(2k+1; k+1; p) = N(d_i) + O\left(\frac{1}{N^7}\right)$.

Thus, if we define the tree parameters p as above, $u = e^{r\Delta t} \frac{p'}{p}$ and $d = \frac{e^{r\Delta t} - p^2 u}{1 - p^2}$ as described in Joshi(2010), where p' is obtained by replacing the α_m 's from above by β_m 's, then a seventh order tree can be constructed.

3.5. Combining the Tian tree and the higher order tree. Joshi(2009) investigated 11 trees with and without acceleration techniques for pricing American put options and found that the most effective trees among the 11 trees were the Tian modified binomial tree with truncation, smoothing and extrapolation, and the higher order tree with truncation and extrapolation. We would like to construct a tree which has features of both trees, we shall refer to this tree as the Higher order Tian tree and denote it by 'HT'. One way to proceed is to start with the higher order tree's construction and make adjustment so that the tree adapts the features of Tian's tree. In particular, we are interested in matching the first two moments exactly. We will first show that such a construction is impossible, then we discuss another way to construct the tree we want.

Lemma 3.3. *It is not possible to construct a tree using Joshi's(2010) tree scheme with second order convergence if the first two moments are matched exactly in spot space.*

Proof outline: We will show if we use the specifications from the higher order tree scheme and match the first two moments exactly in spot space, then the first order error term would not be zero, hence the tree is not of second order.

Proof. First, the higher order tree construction requires $p = \frac{1}{2} + \frac{1}{\sqrt{k}} \sum_{m=0}^{\infty} \alpha_m k^{-m}$.

Substituting into the first two moments matching condition,

$$\begin{aligned} pu + (1-p)d &= \exp\left(\frac{rT}{2k+1}\right), \\ pu^2 + (1-p)d^2 &= \exp\left(\frac{(2r + \sigma^2)T}{2k+1}\right). \end{aligned}$$

and solving for u gives:

$$u = \frac{pM_1 + \sqrt{p(1-p)(M_2 - M_1^2)}}{p}$$

, where

$$M_1 = \exp\left(\frac{rT}{2k+1}\right), M_2 = \exp\left(\frac{(2r + \sigma^2)T}{2k+1}\right).$$

Next, the higher order tree also requires $p' = \frac{pu}{r_{2k+1}}$, where $r_{2k+1} = \exp\left(\frac{rT}{2k+1}\right)$.

So we have

$$p' = p + \frac{\sqrt{p(1-p)(M_2 - M_1^2)}}{M_1} = p + \sqrt{p(1-p) \left(\exp\left(\frac{\sigma^2 T}{2k+1}\right) - 1 \right)}.$$

Substitute p and expand p' in an asymptotic series, we get

$$p' = \frac{1}{2} + \frac{1}{\sqrt{k}} \sum_{m=0}^{\infty} \beta_m k^{-m}$$

where the first few coefficients are:

$$\begin{aligned} \beta_0 &= \alpha_0 + \frac{\sigma\sqrt{T}}{2\sqrt{2}}, & \beta_1 &= \alpha_1 - \frac{\sigma\sqrt{T}}{8\sqrt{2}}(8\alpha_0^2 + 1) + \frac{(\sigma\sqrt{T})^3}{16\sqrt{2}}, \\ \beta_2 &= \alpha_2 - \frac{\sigma\sqrt{T}}{64\sqrt{2}}(-3 + 128\alpha_0\alpha_1 + 64\alpha_0^4 - 16\alpha_0^2) - \frac{(\sigma\sqrt{T})^3}{64\sqrt{2}}(8\alpha_0^2 + 3) + \frac{5(\sigma\sqrt{T})^5}{768\sqrt{2}}. \end{aligned}$$

Now apply theorem 3.1, we have

$$\begin{aligned} \Phi(2k+1; k+1; p) &= N(2\sqrt{2}\alpha_0) + \frac{1}{k}e_1 + \frac{1}{k^2}e_2 + \dots \\ \Phi(2k+1; k+1; p') &= N(2\sqrt{2}\beta_0) + \frac{1}{k}e'_1 + \frac{1}{k^2}e'_2 + \dots \end{aligned}$$

where

$$e_1 = \frac{2}{\sqrt{\pi}}e^{-4\alpha_0^2} \left(\alpha_0^3 + \frac{3}{8}\alpha_0 + \alpha_1 \right), \quad e'_1 = \frac{2}{\sqrt{\pi}}e^{-4\beta_0^2} \left(\beta_0^3 + \frac{3}{8}\beta_0 + \beta_1 \right)$$

As it follows from put-call parity that calls and puts have the same order of convergence, here we'll just assume we are dealing with the call options. Then, to have the tree converge to the right price, we need to set $\alpha_0 = d_2/2\sqrt{2}$. It follows from above that $\beta_0 = d_1/2\sqrt{2}$.

Now, by corollary 3.2, if we need $\phi(2k+1; k+1; p)$ to have second order

convergence, we require $e_1 = 0$, i.e. $\alpha_1 = -\alpha_0^3 - \frac{3}{8}\alpha_0$.

Do substitutions and eliminate all α_m 's and β_m 's except α_0 and β_0 , we get

$$e'_1 = \frac{2}{\sqrt{\pi}} e^{-4\beta_0^2} \left(\frac{8\alpha_0^2 + 1}{16\sqrt{2}} (\sigma\sqrt{T}) + \frac{3}{8}\alpha_0 (\sigma\sqrt{T})^2 + \frac{1}{8\sqrt{2}} (\sigma\sqrt{T})^3 \right)$$

This is in general not equal to zero because $\frac{2}{\sqrt{\pi}} e^{-4\beta_0^2} \neq 0$ and

$$\begin{aligned} & \left(\frac{8\alpha_0^2 + 1}{16\sqrt{2}} (\sigma\sqrt{T}) + \frac{3}{8}\alpha_0 (\sigma\sqrt{T})^2 + \frac{1}{8\sqrt{2}} (\sigma\sqrt{T})^3 \right) \\ &= \frac{\sigma\sqrt{T}}{16\sqrt{2}} \left[d_2^2 + 1 + 3d_2(\sigma\sqrt{T}) + 2(\sigma\sqrt{T})^2 \right] \\ &= \frac{\sigma\sqrt{T}}{16\sqrt{2}} \left[\left(d_2 + \frac{3}{2}(\sigma\sqrt{T}) \right)^2 - \frac{(\sigma\sqrt{T})^2}{4} + 1 \right] \end{aligned}$$

is zero only for a particular choice of d_2 .

We have just shown $\Phi(2k+1; k+1; p)$ and $\Phi(2k+1; k+1; p')$ cannot both have second order of convergence, but we have not shown that when put together to price a call option, they cannot have second order of convergence. The proof is as follows.

Consider the first order error term of the expression

$$\begin{aligned} & S\Phi(2k+1; k+1; p') - Ke^{rT}\Phi(2k+1; k+1; p) \\ &= S N(2\sqrt{2}\beta_0) - Ke^{rT} N(2\sqrt{2}\alpha_0) + (Se_1 - Ke^{rT}e'_1) \frac{1}{k} + \dots \end{aligned}$$

By noting $Se^{4\alpha_0^2 - 4\beta_0^2} = Ke^{-rT}$, we have:

$$Se_1 - Ke^{rT}e'_1 = \frac{2}{\sqrt{\pi}} e^{-4\alpha_0^2} Ke^{-rT} \left(\frac{8\alpha_0^2 + 1}{16\sqrt{2}} \sigma\sqrt{T} + \frac{3}{8}\alpha_0 (\sigma\sqrt{T})^2 + \frac{1}{8\sqrt{2}} (\sigma\sqrt{T})^3 \right)$$

This is again in general not equal to zero using similar reasoning as before. Overall, we have shown it is impossible to construct a second order tree with first two moments matched exactly. \square

Recall that we are attempting to construct a tree that combines features of the Tian tree and the higher order tree. We showed that if we construct a tree beginning with the higher order tree's construction, it is impossible to capture the features of Tian's tree. Now we work with the other direction, we start with the Tian tree's construction and adapt to features of the higher order tree. Note that Tian's tree has all three conditions completely specified already, so there is no flexibility we can use. Hence, we start with our generalization of Tian's tree, where the first two moments are matched exactly in spot space and the center node is yet to be specified. We only work with the case where N is even, write $N = 2k$. We are interested in the feature of the higher order tree that places the center node at the strike, so we require

$Su^k d^k = K$. Rearranging gives $ud = (K/S)^{\frac{1}{k}}$. Combining with the moments matching condition and solving the whole system gives:

$$u = \frac{A + \delta}{2}, \quad d = \frac{A - \delta}{2}, \quad p = \frac{e^{r\Delta t} - d}{u - d}.$$

where $W = \left(\frac{K}{S}\right)^{\frac{1}{k}}$, $A = e^{(r+\sigma^2)\Delta t} + W e^{-r\Delta t}$, $\delta = \sqrt{A^2 - 4W}$.

Remark: Note that if we replace K by any function independent of N , there would be little effect on the tree. This is because W is determined by taking N -th root of the prespecified constant, as N increases, the change in W is so small that it does not really affect the tree. This implicitly suggests that it is not exactly the relative position of the center node to the strike matters, it is the dynamic of the change of the relative position (of the center node to the strike) across time steps that matters.

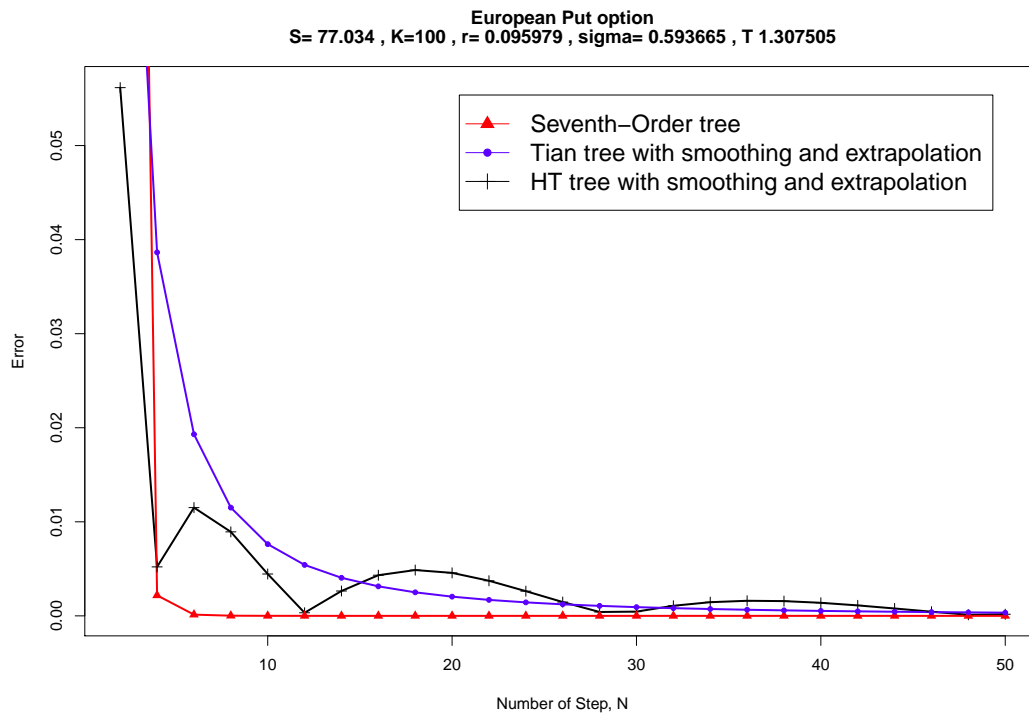


FIGURE 1. Illustration of trees performance: N from 2 to 50

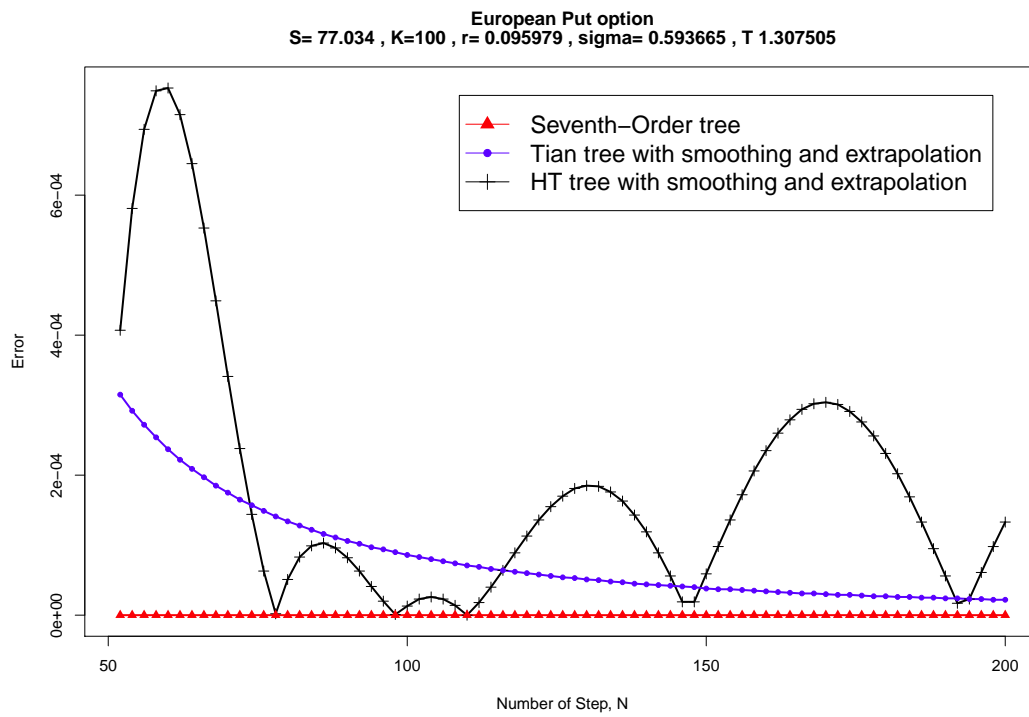


FIGURE 2. Illustration of trees performance: N from 50 to 200

4. ANALYSIS OF THE PRICING ERROR FOR THE AMERICAN PUT OPTIONS

In this section, we analyze the oscillatory nature of the binomial tree model for pricing American put options. To begin with, we consider a general class of tree. It has the first moment matched exactly in spot space for each time step and its parameters have the following forms:

$$p = \frac{1}{2} + \sum_{m=1}^{\infty} c_m N^{-\frac{m}{2}}, \quad u = 1 + \sum_{m=1}^{\infty} d_m N^{-\frac{m}{2}}.$$

We examine closely a situation where the early exercise boundary lies in-between two layers of nodes in a tree. Figure 3 illustrates this. Note that this situation almost always occurs when N is small.

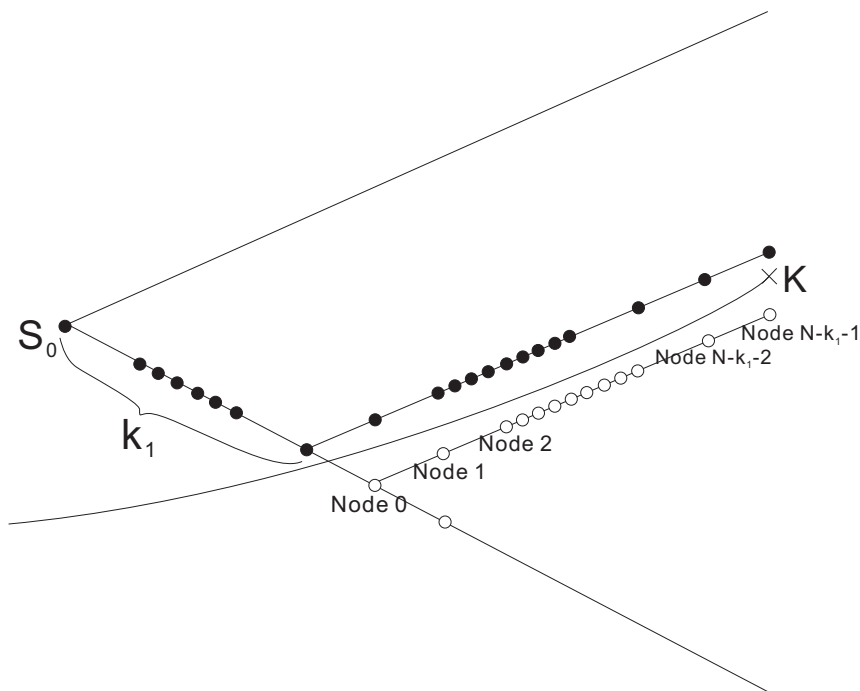


FIGURE 3. Illustration of a situation where the early exercise boundary is in-between two layers of nodes

In the figure, k_1 is the integer such that $k_1 \Delta t < \tau < (k_1 + 1) \Delta t$, where τ is the solution to $Sd^t = B(t)$ and $B(t)$ is the early exercise boundary. It expresses the position of the node before the intersection point at which the early exercise boundary cuts the lower part of the tree. Note that the tree grows in size of order N , so intuitively speaking, if k_1 has a series expansion in N , the highest order would at most be N . While k_1 is unknown, we assume k_1 to be of the form $b_0 N + b_1 \sqrt{N} + O(1)$. We are now done with the setup.

Proposition 4.1. *In a general tree model, if the early exercise boundary is in-between two layer of nodes, and the position of the node before the point where the boundary crosses the lower part of the tree has form $b_0 N + b_1 \sqrt{N} + O(1)$, then an asymptotic expansion in N for the American puts price can be found, i.e. the pricing error is smooth.*

Proof. We present the case when $k_1 = N/2 + b_1\sqrt{N} + b_2$, the proof for the general case $k_1 = b_0N + b_1\sqrt{N} + O(1)$ is similar. To simplify expressions, we write $k_1 = N/2 + k_2$, where $k_2 = b_1\sqrt{N} + b_2$ of course. Let $q = 1 - p$. The American puts price is given by the expectation in the risk-neutral measure. Since at maturity all nodes above K have zero values, so all the contributions to the price come from the layer of nodes right below the early exercise boundary. Using notation in figure 3, we have:

$$\begin{aligned} \text{Price} &= \sum_{k=0}^{N-k_1-1} \mathbb{P}(\text{Nodes } k) \cdot \text{Values}(\text{Nodes } k) \\ &= \sum_{k=0}^{N-k_1-1} \binom{k_1+k}{k} (pe^{-r\Delta t})^k (qe^{-r\Delta t})^{k_1+1} (K - Su^k d^{k_1+1}) \end{aligned}$$

Expanding the last bracket, we get:

$$K \sum_{k=0}^{N-k_1-1} \binom{k_1+k}{k} (pe^{-r\Delta t})^k (qe^{-r\Delta t})^{k_1+1} - S \sum_{k=0}^{N-k_1-1} \binom{k_1+k}{k} (pue^{-r\Delta t})^k (qde^{-r\Delta t})^{k_1+1}$$

Rewrite as:

$$\begin{aligned} &\frac{K (qe^{-r\Delta t})^{k_1+1}}{(1 - pe^{-r\Delta t})^{k_1+1}} \cdot \sum_{k=0}^{N-k_1-1} \binom{k_1+k}{k} (pe^{-r\Delta t})^k (1 - pe^{-r\Delta t})^{k_1+1} \\ &\quad - \frac{S (qde^{-r\Delta t})^{k_1+1}}{(1 - pue^{-r\Delta t})^{k_1+1}} \sum_{k=0}^{N-k_1-1} \binom{k_1+k}{k} (pue^{-r\Delta t})^k (1 - pue^{-r\Delta t})^{k_1+1} \end{aligned}$$

Now note that the summations are in the form of a cumulative distribution function of the negative binomial distribution, using the identity in Patil (1960), it is equal to:

$$\begin{aligned} K \left(\frac{q}{e^{r\Delta t} - p} \right)^{k_1+1} [1 - I_{pe^{-r\Delta t}}(N - k_1, k_1 + 1)] \\ - S \left(\frac{qd}{e^{r\Delta t} - pu} \right)^{k_1+1} [1 - I_{pue^{-r\Delta t}}(N - k_1, k_1 + 1)] \end{aligned}$$

where $I_{\hat{p}}(a, b) = \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} \int_0^{\hat{p}} y^{a-1} (1-y)^{b-1} dy$ is the regularized incomplete beta function.

We can simplify the expression further by noting that the first moment matching condition $pu + qd = e^{r\Delta t}$ implies $qd = e^{r\Delta t} - pu$. Hence the term right behind S is just one, and we get:

$$(4) \quad K \left(\frac{q}{e^{r\Delta t} - p} \right)^{k_1+1} [1 - I_{pe^{-r\Delta t}}(N - k_1, k_1 + 1)] - S [1 - I_{pue^{-r\Delta t}}(N - k_1, k_1 + 1)]$$

Here we examine $I_{\hat{p}}(N - k_1, k_1 + 1)$ closely and aim to expand it in an asymptotic series. For the integral to have an asymptotic expansion, k_1 has to be

carefully chosen. Our assumption of $k_1 = b_0N + b_1\sqrt{N} + O(1)$ is made based on this motivation. See Diener and Diener(2004) for more technical details.

1. First we consider the term $\frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)}$. We use the asymptotic approximation of the Gamma function (Copson 1965, p.57):

$$\Gamma(z) = z^{z-\frac{1}{2}}e^{-z}\sqrt{2\pi} \left(1 + \frac{1}{12z} + \frac{1}{288z^2} + \dots\right), \quad |\arg(z)| < \pi$$

Let $F(z) = \left(1 + \frac{1}{12z} + \frac{1}{288z^2} + \dots\right)$. We get:

$$\begin{aligned} & \frac{\Gamma(N+1)}{\Gamma(N-k_1)\Gamma(k_1+1)} \\ &= \frac{(N+1)^{N+\frac{1}{2}}}{\sqrt{2\pi} \left(\frac{N}{2} - k_2\right)^{\frac{N}{2}-k_2-\frac{1}{2}} \left(\frac{N}{2} + k_2 + 1\right)^{\frac{N}{2}+k_2+\frac{1}{2}}} \frac{F(N+1)}{F\left(\frac{N}{2} - k_2\right) F\left(\frac{N}{2} + k_2 + 1\right)} \end{aligned}$$

Dividing the numerator and denominator by $N^{N+\frac{1}{2}}$,

$$= \left(\frac{2^N\sqrt{N}}{\sqrt{2\pi}}\right) \left[\frac{\left(1 + \frac{1}{N}\right)^{N+\frac{1}{2}}}{\left(1 - \frac{2k_2}{N}\right)^{\frac{N}{2}-k_2-\frac{1}{2}} \left(1 + \frac{2(k_2+1)}{N}\right)^{\frac{N}{2}+k_2+\frac{1}{2}}} \right] \left[\frac{F(N+1)}{F\left(\frac{N}{2} - k_2\right) F\left(\frac{N}{2} + k_2 + 1\right)} \right]$$

Now it is clear the terms inside the middle and the last square brackets have asymptotic expansions. We will deal with the term in the round bracket, denoted by (*), later.

2. Next we consider the term $\int_0^{\hat{p}} y^{a-1}(1-y)^{b-1}dy$. We apply similar techniques to those employed in Diener and Diener(2004):

$$\begin{aligned} \int_0^{\hat{p}} y^{N-k_1-1}(1-y)^{k_1}dy &= \int_0^{\hat{p}} y^{\frac{N}{2}-k_2-1}(1-y)^{\frac{N}{2}+k_2}dy \\ &= \int_0^{\hat{p}} (y(1-y))^{\frac{N}{2}} \left(\frac{1-y}{y}\right)^{k_2} \frac{1}{y} dy \\ &= \int_0^{\hat{p}} e^{Nh(y)}g(y)dy \end{aligned}$$

where $h(y) = \log\left(\sqrt{y(1-y)}\right)$, $g(y) = \left(\frac{1-y}{y}\right)^{k_2} \frac{1}{y}$

Now we let $z = \left(y - \frac{1}{2}\right)\sqrt{N}$. The reason for this change of variables is we note that $h(y)$ has an unique global maximum over the integration domain at $y = \frac{1}{2}$ and we prepare to apply the Laplace's method to approximate the

integral. We have:

$$= \int_{-\frac{\sqrt{N}}{2}}^{(\hat{p}-\frac{1}{2})\sqrt{N}} e^{N h(\frac{1}{2} + \frac{z}{\sqrt{N}})} g\left(\frac{1}{2} + \frac{z}{\sqrt{N}}\right) dz \frac{1}{\sqrt{N}}$$

We then do an asymptotic expansion in N for $h\left(\frac{1}{2} + \frac{z}{\sqrt{N}}\right)$ and series expansion in z for $g\left(\frac{1}{2} + \frac{z}{\sqrt{N}}\right)$:

$$h\left(\frac{1}{2} + \frac{z}{\sqrt{N}}\right) = -\log(2) - \frac{2z^2}{N} - \frac{4z^4}{N^2} - \frac{32}{3} \frac{z^6}{N^3} + O\left(\frac{1}{N^4}\right)$$

$$\begin{aligned} g\left(\frac{1}{2} + \frac{z}{\sqrt{N}}\right) &= 2 - \left[8b_1 + \frac{4(2b_2 + 1)}{\sqrt{N}}\right] z + \left[16b_1^2 + \frac{16b_1(2b_2 + 1)}{\sqrt{N}} + \frac{8(2b_2^2 + 2b_2 + 1)}{N}\right] z^2 \\ &- \left[\frac{64}{3}b_1^3 + \frac{32(2b_2 + 1)}{\sqrt{N}}b_1^2 + \left(\frac{2}{3} + b_2 + b_2^2\right) \frac{64}{N}b_1 + \left(16 + \frac{128}{3}b_2 + 32b_2^2 + \frac{64}{3}b_2^3\right) \frac{1}{N^{3/2}}\right] z^3 \\ &+ O(z^4) \end{aligned}$$

Substituting back into the integral, we have an approximation of:

$$\begin{aligned} &= \frac{2^{-N}}{\sqrt{N}} \int_{-\frac{\sqrt{N}}{2}}^{(\hat{p}-\frac{1}{2})\sqrt{N}} e^{-2z^2} \\ &\left\{ 2 - \left[8b_1 + \frac{4(2b_2 + 1)}{\sqrt{N}}\right] z + \left[16b_1^2 + \frac{16b_1(2b_2 + 1)}{\sqrt{N}} + \frac{8(2b_2^2 + 2b_2 + 1)}{N}\right] z^2 \right. \\ &\left. - \left[\frac{64}{3}b_1^3 + \frac{32(2b_2 + 1)}{\sqrt{N}}b_1^2 + \left(\frac{2}{3} + b_2 + b_2^2\right) \frac{64}{N}b_1 + \left(16 + \frac{128}{3}b_2 + 32b_2^2 + \frac{64}{3}b_2^3\right) \frac{1}{N^{3/2}}\right] z^3 \right\} dz \end{aligned}$$

Note that the coefficient before the integral cancels out nicely the N dependence in (*). And the integrand is just a sum of Gaussian functions, so the integral can be evaluated. We are done with $I_{\hat{p}}(N - k_1, k_1 + 1)$.

Now we deal with the term $\left(\frac{q}{e^{r\Delta t} - p}\right)^{k_1+1}$. Recall we have $p = \frac{1}{2} + \sum_{m=1}^{\infty} c_m k^{-\frac{m}{2}}$,

so all of q , $e^{r\Delta t}$ and p have asymptotic expansions in N . Hence, the asymptotic expansion of the quotient inside the bracket can be found easily, it is equal to:

$$1 - 2 \frac{rT}{N} - 4rTc_1 \left(\frac{1}{N^{3/2}}\right) + \frac{4rT(rT - c_2) - r^2T^2 - 8rTc_1^2}{N^2} + \dots$$

Then we take it to the power $N/2 + b_1\sqrt{N} + b_2 + 1$. We use identity $a^n = e^{n \log a}$, asymptotic expansions of the exponential function and the logarithmic function to get:

$$e^{-rT} \left[1 - \frac{2rT(b_1 + c_1)}{\sqrt{N}} + \dots \right]$$

(Here we only give the first few terms of the expansion because terms at the back get very cumbersome).

We have by now shown all the terms in equation (4) admit asymptotic expansions. Thus, overall the American puts price has an asymptotic expansion in N . \square

Remark: Recall that our proposition assumes the early exercise boundary lies in-between two layers of nodes. So ideally, if the boundary is known, then we can grow a tree along the boundary so that the above situation occurs for all N , it then follows an arbitrarily high order tree can be constructed for pricing American puts. But, of course, in reality, without performing heavy computations, the early exercise boundary is not known. Yet, we can always use an approximation. One could expect that the better the early exercise boundary is approximated, the better accuracy we could achieve for the price. This motivates us to construct new trees by adapting trees to the early exercise boundary in section 5.

We now generalize proposition 4.1 by going one step further and considering the case where exactly one layer of nodes crosses the early exercise boundary. Here is an illustration.

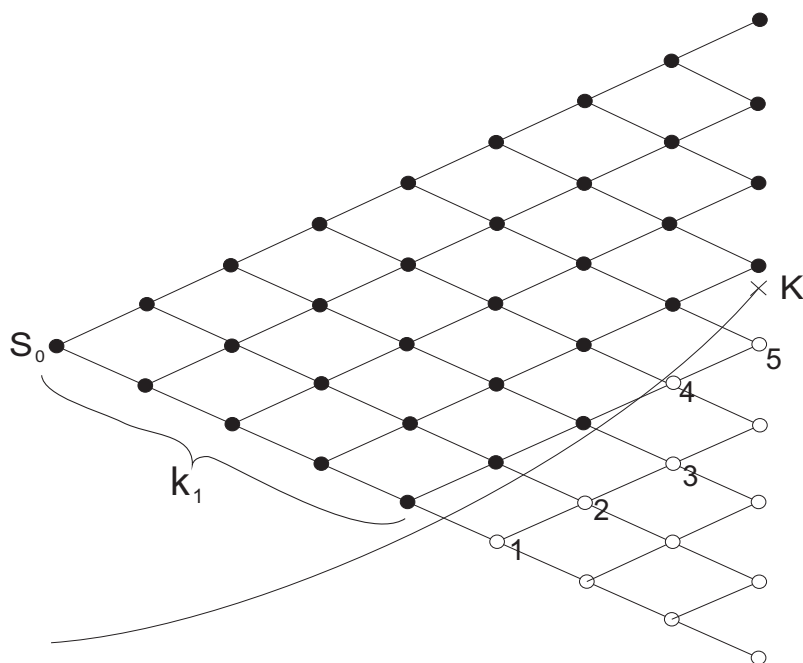


FIGURE 4. Illustration of a situation where exactly one layer of node crosses the early exercise boundary

In this situation, an analysis similar to what we presented can be applied. Using the case in figure 4 as an example, we have:

$$\begin{aligned} \text{Price} &= \sum_{k=1}^5 \mathbb{P}(\text{Nodes } k) \cdot \text{Values}(\text{Nodes } k) \\ &= \sum_{k=0}^2 \binom{k_1 + k}{k} (pe^{-r\Delta t})^k (qe^{-r\Delta t})^{k_1+1} (K - Su^k d^{k_1+1}) \\ &\quad + \sum_{k=3}^4 \binom{k_1 - 1 + k}{k} (pe^{-r\Delta t})^k (qe^{-r\Delta t})^{k_1} (K - Su^k d^{k_1}) \end{aligned}$$

Clearly, the first summation can be expanded in an asymptotic series by going through the same procedure we presented in proposition 4.1. The second summation can be written as the difference of two summation, $\sum_{k=0}^4 - \sum_{k=0}^2$, where each of them has an asymptotic expansion by again applying proposition 4.1. The general case where more than one layer of nodes crosses the early exercise boundary can be done inductively. The main idea is that the tool we developed allows us to compute asymptotic expansions for an expression which has contributions coming from the nodes in the same layer.

Now we apply our analysis to explain the oscillatory behaviour of the price error of American put options. While this analysis is based on the assumptions we made which some other tree models may not have satisfied, the result should have indicative values as the tree models in essence do not differ by a lot. Figure 5 demonstrates the oscillatory nature of the price error for an American put option. From the figure, we observe cycles in the price error. Each cycle contains two phases. In phase 1, as N increases, new nodes enter the layers of the tree while the old nodes stay in the same layer as they were before the new nodes enter. Figure 6 illustrates this. This is simply the outcome of our previous analysis: the price has an asymptotic expansion in N , so we see a monotonic decreasing error in phase 1. Phase 2 is the adjustment phase. At the end of phase 1, a new layer crosses the boundary causing a movement of contributing nodes from one layer to another layer. Figure 7 illustrates this.

As N increases and the layer cuts deeper into the boundary, more contributing nodes (the white nodes) move from the upper layer(layer 2) to the lower layer(layer 3). Since the nodes in the lower layer have higher values, so there is an increase in the price, and hence an increase in the price error. When the layer cuts sufficiently deep into the boundary, the situation stabilizes and no more movement of nodes (across layers) occurs. The phase 2 ends and another cycle begins.

Here we remark that the little increments from time to time actually should be thought as the ‘‘corrections’’ to the price. This is because a tree in general does not grow exactly but approximately along the boundary, so the model

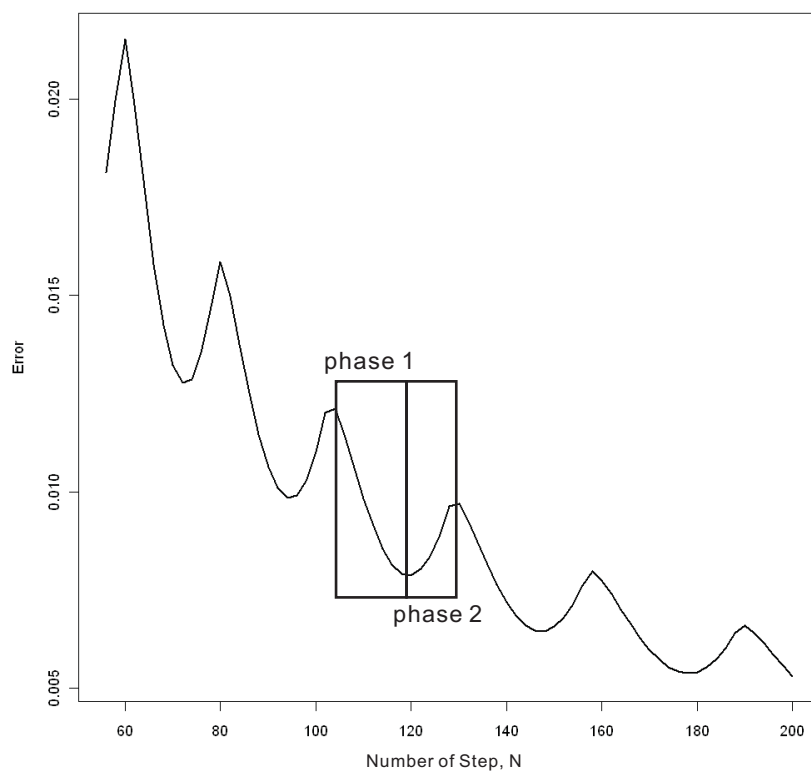


FIGURE 5. Illustration of the oscillatory nature of the price error for an American put option.

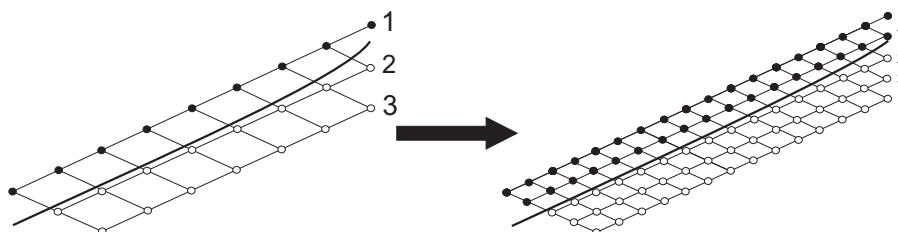


FIGURE 6. Illustration of the change of nodes near the boundary.

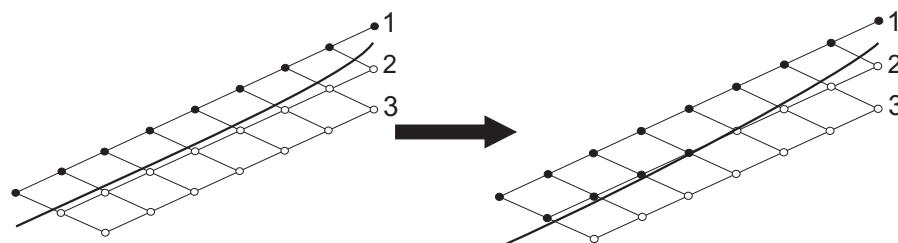


FIGURE 7. Illustration of the change of nodes near the boundary.

does not converge to the right price at the first place. Only when the boundary is crossed by more layers of nodes, extra information about the boundary arrives, adjustment (the increments) is performed to the price and then the model gradually converges to the right price. The concepts of the monotonically decreasing error (phase 1) and consecutive corrections (phase 2) are the center blocks/underlying intuitions of our American puts pricing error analysis. Another remark is that as N increases, the increments will get increasingly smaller because the contribution from each node diminishes as N increases. This makes sense in the context of convergence because if the increments behave otherwise, the price would then fail to converge. This completes our analysis, and now we proceed to the next section where we will construct new trees to improve convergence.

5. NEW TREE CONSTRUCTION

From our analysis, we see that just like other methods used to price American put options, the binomial tree method can also be viewed as a problem of approximating the early exercise boundary (with the center layer of nodes). (Note that it actually does not matter which layer of nodes we use, here we use the center layer just for convenience.)

We revisit the Tian's modified binomial tree. Suppose $N = 2k$, from the model we have $ud = e^{(2r+2\sigma^2)\Delta t}$. This implies that the center layer of nodes is given by $(ud)^i = e^{(2r+2\sigma^2)i\Delta t}$, $i = 1, 2, \dots, k$. So the Tian modified tree can be interpreted as approximating the boundary $B(t)$ by the exponential function $e^{(r+\sigma^2)t}$. This is similar to the idea in Omberg(1987) where the pricing of American put options was studied under the analytical framework using a general exponential function as an approximation to the early exercise boundary. As Ju(1998) extended the idea and used a piecewise exponential function as approximate for the early exercise boundary, we can also apply similar idea to develop a 'split tree'. While Joshi(2009) has also suggested a split tree approach, they are vastly different in the way the tree is grown. The two drifts used in Joshi(2009) were

$$\mu_1 = \frac{\log K - \log S}{\Delta t \lfloor N/2 \rfloor}, \quad \mu_2 = 0$$

while our drifts are chosen such that the tree adapts to the approximative early exercise boundary.

We first discuss two ways to get an approximate of the early exercise boundary, then we specify two ways to adapt the tree to the boundary. As many researches on early exercise boundary have been done, there are numerous approximations available. We can simply pick one of them. However, we have to be careful that some approximations only work with a short timeframe and are completely inaccurate when used to price options with long maturity. For our simulation, we will use the approximative boundary (3.8) in Zhu and

He(2007) and we choose k to be $\frac{(r + \sigma^2)2\sigma^2T}{(r + 2\sigma^2)}$, i.e.

$$(5) \quad B(t) = K \left[\frac{2r}{2r + \sigma^2} + \frac{\sigma^2}{2r + \sigma^2} e^{-(r+\sigma^2)T\sqrt{T-t}} \right]$$

The other way to go is to use a tree to find out the early exercise nodes, then fit an exponential function to them to get an approximative boundary, a possible form is: $Ke^{A\sqrt{T-t}+B(T-t)}$.

Now we give two ways of adapting the tree to the boundary. Assume $N = 2k$ and assume that we already have a boundary approximate $B(i)$, where $i = 0, \Delta t, 2\Delta t, \dots, (N-1)\Delta t, N\Delta t = T$.

5.1. Split Tree. For the first k steps, we use (u_1, d_1, p_1) as the tree parameters, and for the remaining k steps, we use (u_2, d_2, p_2) as the tree parameters, where

$$u_i d_i = \begin{cases} \left(\frac{B(T/2)}{B(0)} \right)^{\frac{1}{k}}, & i = 1 \\ \left(\frac{B(T)}{B(T/2)} \right)^{\frac{1}{k}}, & i = 2 \end{cases} \quad \frac{u_i}{d_i} = e^{2\sigma\sqrt{\Delta t}}, i = 1, 2. \quad p_k = \frac{e^{r\Delta t} - d_k}{u_k - d_k}, i = 1, 2.$$

Solving them gives:

$$u_1 = \sqrt{e^{2\sigma\sqrt{\Delta t}} \left(\frac{B(T/2)}{B(0)} \right)^{\frac{1}{k}}}, \quad d_1 = \sqrt{e^{-2\sigma\sqrt{\Delta t}} \left(\frac{B(T/2)}{B(0)} \right)^{\frac{1}{k}}}, \quad p_1 = \frac{e^{r\Delta t} - d_1}{u_1 - d_1}.$$

$$u_2 = \sqrt{e^{2\sigma\sqrt{\Delta t}} \left(\frac{B(T)}{B(T/2)} \right)^{\frac{1}{k}}}, \quad d_2 = \sqrt{e^{-2\sigma\sqrt{\Delta t}} \left(\frac{B(T)}{B(T/2)} \right)^{\frac{1}{k}}}, \quad p_2 = \frac{e^{r\Delta t} - d_2}{u_2 - d_2}.$$

5.2. Boundary matching tree. A natural generalization to the split tree would be a tree with more divisions. The most we can do is N . So for the i -th steps, we use (u_i, d_i, p_i) as the tree parameters, where

$$u_i d_i = \frac{B((i+1)\Delta t)}{B(i\Delta t)}, \quad \frac{u_i}{d_i} = e^{2\sigma\sqrt{\Delta t}}, \quad p_i = \frac{e^{r\Delta t} - d_i}{u_i - d_i}.$$

Solving them gives:

$$u_i = \sqrt{e^{2\sigma\sqrt{\Delta t}} \left(\frac{B((i+1)\Delta t)}{B(i\Delta t)} \right)}, \quad d_i = \sqrt{e^{-2\sigma\sqrt{\Delta t}} \left(\frac{B((i+1)\Delta t)}{B(i\Delta t)} \right)}, \quad p_i = \frac{e^{r\Delta t} - d_i}{u_i - d_i}.$$

6. NUMERICAL RESULTS.

Here we present numerical results. To compare the pricing error across different trees, we need a measure of error. There are a few common choices, the mean absolute error, the maximum absolute error and the root-mean-squared error. Broadie and Detemple(1996) suggested to use the root-mean-squared

error, because a large error is undesirable and we want to penalize it with more weight. So, the mean absolute error is not appropriate. On the other hand, maximum absolute error over-penalizes large error and reflects very little about the smaller errors. Hence, the root-mean-squared error is used. In addition, we are more interested in relative error than in absolute error, as clearly a error of 0.01 is much more significant for a price of 1 than for a price of 100. So in short, we will use the root-mean-squared relative error. To avoid distortion in results, Broadie and Detemple(1996) removed from the simulations all the cases where the price is less than 0.5. In order to capture these extreme cases(the deeply out-of-the-money cases) while retaining meaningful results, we adopt a modification on the measure of relative error proposed in Joshi(2009). It is as follows:

$$\text{Modified Relative Error} = \frac{\text{TreePrice}-\text{TruePrice}}{0.5+\text{TruePrice}-\text{IntrinsicValue}}$$

For the selection of parameters for the simulations, we follow Broadie and Detemple(1996)'s scheme with one exception, we particularly are interested in the in-the-money American puts, so we take S to be uniformly distributed between 70 and 100. Other parameters are specified as follows:

- K is fixed at 100;
- σ is uniformly distributed between 0.1 and 0.6;
- T is uniformly distributed between 0.1 and 1 with probability 0.75, and it is uniformly distributed between 1 and 5 with probability 0.25;
- r is uniformly distributed between 0 and 0.1 with probability 0.8, and it is equal to 0 with probability 0.2.

We compare the following trees for step numbers of 100, 300 and 500.

Tian: The Tian modified tree in section 3.1 with smoothing and extrapolation.

H7: The seventh-order tree in section 3.3 with extrapolation.

These two trees are chosen because they are the most efficient trees reviewed in Joshi(2009). Now we also include trees developed in this paper.

HT: The tree of section 3.5 which combines features of the higher order tree and the Tian tree.

ST: The Split Tree of section 5.1 using the approximative boundary function specified by equation (5).

BMT: The Boundary Matching Tree of section 5.2 using the approximate boundary function specified by equation (5).

We will also use the accelerated versions of them. In particular, we will use Smoothing, Richardson extrapolation and both. The benchmark price we use is Tian with smoothing and extrapolation using $N=5000$. For the number of simulations, Broadie and Detemple(1996) ran 2500 simulations for each tree, here we will do 10000 simulations.

N=100. #Simulation:10000

	Original	With Smoothing	With Extrapolation	With Smoothing and Extrapolation
HT	0.004495	0.004007	0.003667	0.003629
ST	0.005994	0.005680	0.008192	0.005611
BMT	0.005003	0.004182	0.008172	0.005556
H7	-	-	0.003653	-
Tian	-	-	-	0.003385

TABLE 1. Table of root-mean-squared relative error. Tian with smoothing and extrapolation performs the best. HT with smoothing, HT with smoothing and extrapolation, and H7 place second. HT with smoothing and BMT with smoothing place third.

N=300. #Simulation:10000

	Original	With Smoothing	With Extrapolation	With Smoothing and Extrapolation
HT	0.001691	0.001560	0.001701	0.001694
ST	0.002142	0.002047	0.003136	0.002453
BMT	0.001870	0.001618	0.003132	0.002447
H7	-	-	0.001745	-
Tian	-	-	-	0.001422

TABLE 2. Table of root-mean-squared relative error. Tian with smoothing and extrapolation performs the best. HT with smoothing, BMT with smoothing place second. HT original, HT with smoothing, HT with smoothing and extrapolation place third.

N=500. #Simulation:10000

	Original	With Smoothing	With Extrapolation	With Smoothing and Extrapolation
HT	0.0008482	0.0007598	0.0006996	0.0007006
ST	0.001109	0.001040	0.001585	0.001073
BMT	0.0009179	0.0007159	0.001587	0.001076
H7	-	-	0.0007046	-
Tian	-	-	-	0.0006258

TABLE 3. Table of root-mean-squared relative error. Tian with smoothing and extrapolation performs the best. HT with extrapolation, HT with smoothing and extrapolation, H7 with extrapolation place second. BMT with smoothing place third.

Overall, Tian's tree with smoothing and extrapolation performs the best pricing in-the-money American puts, yet, we remark that while the performance of Tian's tree is about 10-20% better than our trees. The time Tian's tree takes to compute the price is significantly more than our trees. The following table shows the time taken for the trees to compute the price.

#Simulations:10000

	Tian	SmthEplTian	HT	Smth HT	BMT	SmthBMT
N=100	0.001364	0.006829	0.001337	0.001355	0.000934	0.000900
N=300	0.011887	0.059913	0.012008	0.012032	0.019946	0.019881
N=500	0.033359	0.166844	0.033404	0.033362	0.088871	0.088365

TABLE 4. Average time (in sec(s)) for the trees to compute one option price

We observe that the base cases for Tian's tree and HT take roughly the same time to compute one option price. However, when corresponding acceleration techniques are applied, Tian's tree takes roughly 5 times more time than HT to compute the price. Hence, it should be practically more appropriate to price using HT. On the other hand, we also see a superior performance in speed of BMT with smoothing over Tian's tree with smoothing and extrapolation. It is 7.5 times to 2 times faster when N ranges from 100 to 500. One should note that the time BMT takes to compute a price increases cubically, so it is

not suitable for benchmarking, yet when $N \leq 1000$, the tree is almost always more efficient than Tian's tree with smoothing and extrapolation.

7. WHICH TREE DOES BEST IN WHICH REGION?

While the overall performance of some trees are quite close, they actually have different performance locally. To study the local performance, we divide the whole parameters space into twelve regions characterized by 'Long/Short maturity', 'High/Medium/Low interest rate' and 'In-the-money/Out-of-the-money'. We define maturity of 0.1-1.5 years to be short maturity, and maturity of 1.5-5 years to be long maturity. We define an interest rate of 0.07-0.1 to be high interest rate, 0.04-0.07 to be medium interest rate and 0-0.04 to be low interest rate. As we have K fixed at 100, so if S ranges from 70-100, then the option would be in-the-money and if S ranges from 100-130, then the option would be out-of-the-money. Based on 2000 simulations in each region running each tree for $N=200$, we identify which tree performs best in which region and summarize the results in the following table.

S	r	T	Choice
70-100	0-0.04	0.1-1.5	Smth-Epl-HT
70-100	0.04-0.07	0.1-1.5	Smth-BMT
70-100	0.07-0.1	0.1-1.5	Smth-BMT
70-100	0-0.04	1.5-5	Epl-HT
70-100	0.04-0.07	1.5-5	Smth-Epl-Tian
70-100	0.07-0.1	1.5-5	Smth-HT
100-130	0-0.04	0.1-1.5	Epl-H7
100-130	0.04-0.07	0.1-1.5	Epl-H7
100-130	0.07-0.1	0.1-1.5	Epl-H7
100-130	0-0.04	1.5-5	Epl-HT
100-130	0.04-0.07	1.5-5	Epl-HT
100-130	0.07-0.1	1.5-5	Smth-Epl-Tian

TABLE 5. Partition of the parameters space

It is suggested to use the above table and let different trees handle the regions in which they perform the best. For example, if it is out-of-the-money, and the maturity is long, then we will use the higher order tree with extrapolation to price the American put option.

We compare the pricing error using Tian's tree with smoothing and extrapolation alone with the pricing error using our scheme and we get the following results. Using a benchmark of $N=1500$, over 2500 simulations, for $N=100$,

our scheme gives a relative root-mean-squared error of 0.002387, while Tian's tree with smoothing and extrapolation gives a relative root-mean-squared error of 0.002610. For $N=200$, the result is 0.000783 against 0.000894, and for $N=300$, the result is 0.001086 against 0.001194. These demonstrate a consistent improvement of about 10% in the pricing accuracy.

8. CONCLUSION

In this paper, we gave extensions to the two most efficient trees studied in Joshi(2009), the Tian(1993) third-moment-matching tree and the Joshi(2010) higher order tree. We generalized the Tian tree and extended the higher order tree from the third order to the seventh order. Moreover, we constructed a tree, referred as the Higher order-Tian tree, that captures features of both trees. We then presented asymptotic analysis of the pricing error of American put options under the binomial tree scheme and used it to tackle the puzzle, 'the oscillatory behavior of the error of American puts pricing'. Based on that, we constructed another new tree, referred as the Boundary Matching Tree, that adapts to the early exercise boundary. Taking into account the balance between time and accuracy, our trees HT with smoothing and BMT with smoothing gave a better performance than Tian's tree with smoothing and extrapolation, which was known as the most efficient tree for pricing American put options. At the end, we investigated the trees' local performance by dividing the parameters space into different regions and suggested a scheme to incorporate different trees by letting the trees handle the regions they are best in. Compared with using Tian's tree with smoothing and extrapolation alone, our scheme demonstrated a clear improvement in the accuracy of pricing American put options.

REFERENCES

- Amin, K., Khanna, A., 1994. Convergence of American Option Values from discrete- to continuous-time financial models. *Mathematical Finance*, 4, 289-304.
- Andricopoulos, A.D., Widdicks, M., Duck, P.W., Newton, D.P., 2002. On the enhanced convergence of standard lattice methods for option pricing. *Journal of Futures*, 22, 315-338.
- Andricopoulos, A.D., Widdicks, M., Duck, P.W., Newton, D.P., 2004. Curtailing the range for lattice and grid methods. *The Journal of Derivatives*, 11, 5561.
- Bartter, B.J., Rendleman, R.J., Jr., 1979. Two-state option pricing. *Journal of Finance*, XXXIV, 1093-1110.
- Boyle, P.P., 1988. A lattice framework for option pricing with two state variables. *Journal of Financial and Quantitative Analysis*, 23, 1-12.
- Boyle, P.P., Evnine, J., Gibbs, S., 1989. Numerical evaluation of multivariate contingent claims. *Review of Financial Studies*, 2, 241-250.
- Broadie, M., Detemple, J., 1996. American option valuation: new bounds, approximations, and a comparison of existing methods. *Review of Financial Studies*, 9, 1211.
- Chan, J.H., Joshi, M., Tang, R., Yang, C., 2009. Trinomial or binomial: Accelerating American put option price on trees. *Journal of Futures Markets*, 29, 826-839.
- Chance, D., 2008. A Synthesis of Binomial Option Pricing Models for Lognormally Distributed Assets. *Journal of Applied Finance*, 18, 38-56.
- Chang, L., Palmer, K., 2006. Smooth convergence in the binomial model. *Finance and Stochastics*, 11, 91-105.
- Chen, T., Joshi, M., 2011. Truncation and acceleration of the Tian tree for the pricing of American put options. *Quantitative Finance*, 12, 11, 1695-1708.
- Chriss, N.A., 1996. *Black-Scholes and Beyond: Option Pricing Models*. McGraw-Hill, New York.
- Copson, E.T., 1965. *Asymptotic Expansions*. Cambridge University Press.
- Cox, J.C., Ross, S.A., Rubinstein, M., 1979. Option pricing: A simplified approach. *Journal of Financial Economics*, 7, 229-263.
- Diener, F., Diener, M., 2004. Asymptotics of the price oscillations of a European call option in a tree model. *Mathematical finance*, 14, 271-293.
- Figlewski, S., Gao, B., 1999. The adaptive mesh model: a new approach to efficient option pricing. *Journal of Financial Economics*, 53, 313-351.
- Gaudenzi, M., Pressacco, F., 2003. An efficient binomial method for pricing the American options. *Decisions in Economics and Finance*, 26, 1-17.
- Hsia, C., 1983. On binomial option pricing. *The Journal of Financial Research*, VI, 41-46.
- Hull, J., White, A., 1988. The use of the control variate technique in option pricing. *Journal of Financial and Quantitative Analysis*, 23, 237-251.
- Jabbour, G.M., Kramin, M.V., Young, S.D., 2001. Two-State Option Pricing: Binomial Models Revisited. *Journal of Futures Markets*, 21, 987-1001.
- Jarrow, R.A., Rudd, A., 1983. *Option Pricing*. Irwin Professional Publishing.
- Joshi, M.S., 2004. *The Concepts and Practice of Mathematical Finance*, 1st ed. Cambridge University Press.
- Joshi, M.S., 2009. The convergence of binomial trees for pricing the American put. *Journal of Risk*, 11, 87-108.
- Joshi, M.S., 2010. Achieving higher order convergence for the prices of European options in binomial trees. *Mathematical Finance*, 20, 891-903.
- Joshi, M.S., 2011. *More mathematical finance*. Pilot Whale Press.

- Ju, N., 1998. Pricing by American option by approximating its early exercise boundary as a multipiece exponential function. *Review of Financial Studies*, 11, 27.
- Kamrad, B., Ritchken, P., 1991. Multinomial Approximating Models for Options with k State Variables. *Management Science*, 37, 1640-1652.
- Lamberton, D., 1998. Error estimates for the binomial approximation of American put options. *Annals of Applied Probability*, 8, 206233.
- Leisen, D.P.J., 1998. Pricing the American put option: A detailed convergence analysis for binomial models. *Journal of Economic Dynamics and Control*, 22, 1419-1444.
- Leisen, D.P.J., Reimer, M., 1996. Binomial models for option valuation-examining and improving convergence. *Applied Mathematical Finance*, 3, 319346.
- Omberg, E., 1987. The valuation of American put options with exponential exercise policies. *Advances in Futures and Options Research*, 2, 117-142.
- Omberg, E., 1988. Efficient Discrete Time Jump Process Models in Option Pricing. *Journal of Financial and Quantitative Analysis*, 23, 161-174.
- Parkinson, M., 1977. Option Pricing: The American Put. *Journal of Business*, 50, 2136.
- Patil, G.P., 1960. On the Evaluation of the Negative Binomial Distribution with Examples. *Technometrics*, 2, 501.
- Staunton, M., 2005. Efficient estimates for valuing American options, in: Wilmott, P. (Ed.), *The Best of Wilmott 2*. Wiley, England, pp. 91-97.
- Tian, Y., 1993. A modified lattice approach to option pricing. *Journal of Futures Markets*, 13, 563577.
- Tian, Y., 1999. A flexible binomial option pricing model. *Journal of Futures Markets*, 19, 817843.
- Trigeorgis, L., 1991. A log-transformed binomial numerical analysis method for valuing complex multi-option investments. *Journal of Financial and Quantitative Analysis*, 26, 309326.
- Wilmott, P., 1998. *Derivatives: The Theory and Practice of Financial Engineering*. John Wiley & Sons.
- Xiao, X., 2010. Improving speed of convergence for the prices of European options in binomial trees with even numbers of steps. *Applied Mathematics and Computation*, 216, 2659-2670.
- Zhu, S.P., He, Z.W., 2007. Calculating the early exercise boundary of American put options with an approximation formula. *International Journal of Theoretical and Applied Finance*, 10, 1203.