On Accurate and Provably Efficient GARCH Option Pricing Algorithms

Yuh-Dauh Lyuu* Chi-Ning Wu†

Abstract

The GARCH model has been very successful in capturing the serial correlation of asset return volatilities. As a result, applying the model to options pricing attracts a lot of attention. However, previous tree-based GARCH option pricing algorithms suffer from exponential running time, a cut-off maturity, inaccuracy, or some combination thereof. Specifically, this paper proves that the popular trinomial-tree option pricing algorithms of Ritchken and Trevor (1999) and Cakici and Topyan (2000) explode exponentially when the number of partitions per day, n, exceeds a threshold determined by the GARCH parameters. Furthermore, when explosion happens, the tree cannot grow beyond a certain maturity date, making it unable to price derivatives with a longer maturity. As a result, the algorithms must be limited to using small n, which may have accuracy problems. The paper presents an alternative trinomial-tree GARCH option pricing algorithm. This algorithm provably does not have the short-maturity problem. Furthermore, the tree-size growth is guaranteed to be quadratic if n is less than a threshold easily determined by the model parameters. This level of efficiency makes the proposed algorithm practical. The surprising finding for the first time places a tree-based GARCH option pricing algorithm in the same complexity class as binomial trees under the Black-Scholes model. Extensive numerical evaluation is conducted to confirm the analytical results and the numerical accuracy of the proposed algorithm. Of independent interest is a simple and efficient technique to calculate the transition probabilities of a multinomial tree using generating functions.

Keywords: GARCH, trinomial tree, path dependency, option pricing, generating function

[*] Corresponding author. Professor, Department of Computer Science & Information Engineering, National Taiwan University, No. 1, Sec. 4, Roosevelt Rd., Taipei, Taiwan. E-mail: lyuu@csie.ntu.edu.tw. Also Professor of Department of Finance, National Taiwan University. The author was supported in part by NSC grant 92-2213-E-002-016.

[†] Department of Finance, National Taiwan University, Taipei, Taiwan.

1 Introduction

Efficient numerical algorithms play a critical role in derivatives pricing because it is often imperative to obtain prices fast, particularly when prices change quickly. In both theory and practice, computational efficiency is measured in terms of running time. Two types of algorithms are usually distinguished: polynomial-time algorithms and exponential-time algorithms (Papadimitriou (1995)). Because the exponential function grows extremely fast, exponential-time algorithms highlight the tradeoff between accuracy and speed much earlier than polynomial-time algorithms. Exponential-time algorithms are therefore said to suffer from combinatorial explosion and should be avoided wherever possible.

In the numerical pricing of derivatives, the continuous diffusion process for the asset price is often discretized to yield a tree first. Derivatives are then priced on the tree by standard backward induction. The lognormal diffusion, for instance, gives rise to the well-known CRR binomial tree of Cox, Ross, and Rubinstein (1979). Two critical features of the CRR tree, as well as its many binomial and trinomial variants, stand out: It recombines and an N-period tree contains only $O(N^2)$ nodes; i.e., it exhibits a quadratic growth (see Fig. 1). As a consequence, simple derivatives such as vanilla options, barrier options, and lookback options can be efficiently priced as surveyed in Lyuu (2002). However, a polynomial-sized tree may still give rise to an exponential-time pricing algorithm if the derivative's payoff is complex. The Asian option with a payoff depending on the arithmetic price average fits this characterization. Specifically, the vast number of extra states (the running averages) needed by the Asian option's path-dependent feature makes pricing on an N-period tree take time exponential in N. Approximations are mandatory for such derivatives to regain efficiency. Of course, approximation algorithms must not sacrifice accuracy. Important numerical techniques include the tree method with interpolation such as Hull and White (1993a), the PDE method such as Forsyth, et al. (2002), and the linear-programming technique such as Dempster and Richards (2000).

A fundamentally more difficult problem emerges when the explosion arises from the model itself. If the model generates trees that do not recombine, pricing is expensive even for the simplest of derivatives like vanilla options. For example, when the volatility is not a constant, such as the interest rate model of Cox, Ingersoll, and Ross (1985), a brute-force discretization leads to exploding binomial trees that do not recombine. The problem may be rectified by the standard technique of Nelson and Ramaswamy (1990) that transforms the diffusion process into one with a constant volatility. But this methodology does not guarantee to do away with combinatorial explosion when the diffusion process is bivariate. For example, Chien (2003) demonstrated that the bivariate interest rate tree of Ritchken and Sankarasubramanian (1995) can explode exponentially even after the said transformation. The focus of

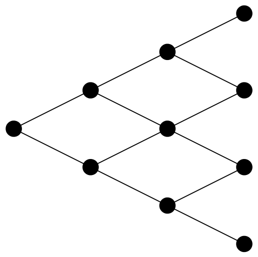

Figure 1: **Binomial tree.** Each node has two successor nodes. The number of nodes at any time t is $t+1$, a linear growth in time. The total number of nodes of an N-period binomial tree is $\sum_{t=0}^{N}(t+1) = (N+2)(N+1)/2 = O(N^2)$, a quadratic growth in maturity N.

this paper, the tremendously influential generalized autoregressive conditional heteroskedastic (GARCH) model, is also bivariate.

Bollerslev (1986) and Taylor (1986) independently proposed the GARCH process for modeling the stochastic volatility of asset returns. Since then, the model has been generalized and used extensively in the finance literature on the modeling of financial time series; see Bollerslev et al. (1992) and Engle and Patton (2001). As the model has received strong empirical support, its application to option pricing draws a lot of attention. Duan (1995) was the first to propose a GARCH option pricing model. The massive path dependency of the pricing model initially favors Monte Carlo simulation over trees. But the Monte Carlo estimate is probabilistic; furthermore, options that can be exercised early, the so-called American options, can be accurately priced only with simulation schemes that employ advanced techniques. The appearance of the trinomial tree of Ritchken and Trevor (1999) addresses these problems and makes a strong case for trees. Their algorithm is simple and claims to be accurate and efficient. It is also general enough to work beyond GARCH models. GARCH option pricing techniques that are not based on trees include the Markov chain approximation of Duan and Simonato (2001), the Edgeworth tree approximation of Duan et al. (2002), and analytical approximations as in Heston and Nandi (2000). Among them, only the Markov chain approximation approach is capable of handling American options.

This paper first investigates the performance of the Ritchken-Trevor algorithm and its modified version by Cakici and Topyan (2000). The findings are mixed, both theoretically and numerically. We prove that the Ritchken-Trevor-Cakici-Topyan (RTCT) algorithm can easily result in exponential-sized trees. This theoretical result is decisively backed up by numerical data. Interestingly, the simple condition for the combinatorial explosion mirrors that for GARCH to be nonstationary and is satisfied when the number of partitions per day, n, exceeds a typically small number. RTCT is hence efficient only if it is restricted to small n. But a small n does not guarantee efficiency; in fact, even the smallest possible choice $n = 1$ can result in explosion under some circumstances. A small n can also yield inaccurate option prices. We note that the Cakici-Topyan algorithm is incompletely specified (see Section 4).

Even though raising n to improve accuracy will quickly incur exponential slowdown, one may be willing to trade efficiency for accuracy. (For example, the slow exponential-time algorithm can be run overnight on spare computers to obtain prices for situations likely to happen on the next trading day.) But this sensible practice turns out to be impossible for RTCT. Specifically, we prove that when explosion occurs, the RTCT tree cannot grow beyond a certain maturity, making it useless for pricing derivatives with a longer maturity. Numerical data again back up this analytical result. The strong impossibility result obliterates the tradeoff between accuracy and efficiency for exploding RTCT trees. The loss of this tradeoff can be devastating if the permissible n's fail to give accurate prices. The result also throws into question some of the calculated prices in Ritchken and Trevor (1999).

This paper then proposes a new trinomial-tree GARCH option pricing algorithm that addresses the above-mentioned shortcomings of RTCT. In a trinomial tree, the three successor nodes of each node must be such that their stock prices match the mean and variance of the model's stock price asymptotically. The three branches must also carry valid branching probabilities. It is obvious that the higher the volatility, the wider the nodes must be spaced to meet the requirements. The RTCT tree, like typical trinomial trees, takes a flat middle branch from each node as in Fig. 2(a). Our new tree departs from that by making the middle branch track the expected stock price as closely as possible as in Fig. 2(b). Therefore, we call it the mean-tracking (MT) algorithm. By tracking the mean, the two flanking branches are expected to fan out less in their attempt to match the mean and variance of the next stock price. This in turn yields more compact trees. Although the mean-tracking idea works for any model, the effects are most striking for models with a stochastic volatility such as GARCH. The above argument is both intuitively appealing and provably true as this paper shows.

MT solves the short-maturity problem of RTCT. As a result, it accepts any n without having to worry about the tree being cut short. The tradeoff between efficiency and accuracy is thus restored. Because an exponential-sized tree consumes so many resources, one may want to be assured that they are not building one before committing the efforts. But is there a bound on n which guarantees a polynomial-sized MT tree? Indeed, such a bound exists and is surprisingly simple. Perhaps even more unexpectedly, the MT tree's size is only quadratic in maturity if n does not exceed that bound. MT is therefore the first tree-based GARCH option pricing algorithm that is provably efficient. Interestingly, this result places MT in the same complexity class as CRR under the much simpler Black-Scholes model. A level of

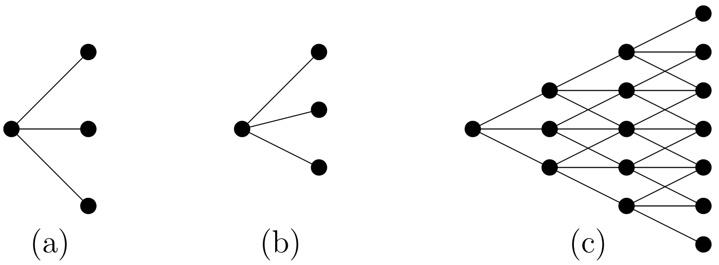

Figure 2: **Trinomial tree.** (a) Three successor nodes follow each node. How widely the two flanking branches fan out around the middle branch depends critically on volatility. (b) The middle branch may maintain a drift to minimize that width. (c) When the vertical node spacing is a constant, the number of nodes at any time t is $2t + 1$, a linear growth. The total number of nodes of an N-period trinomial tree is thus $\sum_{t=0}^{N}(2t + 1) = (N + 1)^2 = O(N^2)$, a quadratic growth in maturity N.

efficiency like this opens up opportunities for the practical use of MT in pricing. The proof of the theoretical results requires only elementary techniques. Numerical experiments demonstrate that a small n gives accurate results. They also definitively confirm the quadratic node count.

In a general sense, the concept of mean tracking is not entirely new to the literature. Numerical pricing techniques that remove the growth trend in the asset price can be said to adopt mean tracking, if only vaguely or implicitly at that. The idea, for instance, is explicit in the algorithms of Hull and White (1993b) and Li et al. (1995). Both works deal with the calibration of no-arbitrage interest rate models. The advantages of mean tracking in reducing complexity, however, have not been analytically explored until recently. For instance, Dai and Lyuu (2004) applied the mean-tracking idea to develop the first exact trinomial-tree Asian option pricing algorithm that breaks the long-standing 3^N time barrier with a provable running time of $3^{O(\sqrt{N})}$, where N is the number of periods of the tree.

Because the numerical performance of the Cakici-Topyan (CT) version of the GARCH option pricing algorithm is slightly superior to that of the Ritchken-Trevor (RT) version, we will compare MT with CT in the numerical valuation of option prices. Our theoretical results on explosion and the shortened maturity of exploding trees apply to both RT and CT. Therefore, we will simply refer to both as RTCT in the analysis.

The paper is organized as follows. The GARCH model is presented in Section 2. Section 3 reviews the RTCT tree from which the MT tree derives. Several differences between the two will be pointed out along the way. A very simple and efficient way to calculate the transitional probabilities using generating functions is also presented there. Section 4 covers backward induction with interpolated volatilities, which are needed to reduce the state space. In Section 5, a condition for the RTCT tree to explode exponentially is stated and proved. The nonexistence of the tradeoff between efficiency and accuracy for the exploding RTCT tree is proved in Section 6. Section 7 provides numerical data to back up the theoretical results and other negative claims regarding RTCT. Section 8 presents and analyzes MT. Section 9 proves that the MT tree grows only quadratically when n does not exceed a simple bound. Section 10 evaluates the various GARCH option pricing algorithms numerically. Section 11 concludes.

2 The GARCH model

Let S_t denote the asset price at date t and h_t the conditional volatility of the return over the $(t+1)$th day $[t, t+1]$. Here, "one day" is just a convenient term for any elapsed time Δt. The following risk-neutral process for the logarithmic price $y_t \equiv \ln S_t$ is due to Duan (1995):

$$y_{t+1} = y_t + r - \frac{h_t^2}{2} + h_t \epsilon_{t+1}, \tag{1}$$

where

$$\begin{aligned} h_{t+1}^2 &= \beta_0 + \beta_1 h_t^2 + \beta_2 h_t^2 (\epsilon_{t+1} - c)^2, \quad (2) \\ \epsilon_{t+1} &\sim N(0,1) \text{ given information at date } t, \\ r &= \text{daily riskless return.} \end{aligned}$$

The model is bivariate as its state is described by (y_t, h_t^2). Updating rule (2) for the squared volatility, due to Engle and Ng (1993), is also called the nonlinear asymmetric GARCH or NGARCH for short. Other GARCH models are surveyed in Duan (1997).

A positive c represents a negative correlation between the shock for the asset return and its conditional volatility. We assume $\beta_0, \beta_1, \beta_2 \geq 0$ to make the squared volatilities h_t^2 positive. We further impose $\beta_1 + \beta_2 < 1$ to make the model stationary. The violation of a version of this inequality will be shown to make the RTCT tree explode. Specifically, explosion occurs when $\beta_1 + \beta_2 n > 1$ given $c = r = 0$ (the general case of nonzero c and/or c is similar, but the explicit formula is complex). In contrast, the MT tree grows only quadratically in maturity when $n \leq (\sqrt{\frac{1-\beta_1}{\beta_2}} - c)^2$. Note that when $c = 0$, this bound becomes $\beta_1 + \beta_2 n \leq 1$, the exact complement of the bound for RTCT to explode. Hence both bounds are optimal in a sense. Throughout the paper, N will denote the maturity of the tree in days, which is also the maturity of the option to be priced by the tree.

3 The RTCT Tree

The RTCT trinomial tree approximates the continuous-state GARCH process with discrete states as follows. Each state is represented as a node. Partition a day into n periods. Three states follow each state (y_t, h_t^2) after a period. As the trinomial tree recombines, $2n+1$ states at date $t+1$ follow each state at date t. Let $\gamma = h_0$ and $\gamma_n = \gamma/\sqrt{n}$. (Our theoretical results will turn out to be independent of how γ is picked. Later, MT will choose a different γ for numerical reasons.) The tree is laid over nodes that are spaced by γ_n in their logarithmic prices as depicted in Fig. 3(a). Consequently, the logarithmic price y_t on each node equals $y_0 + k\gamma_n$ for some integer k.

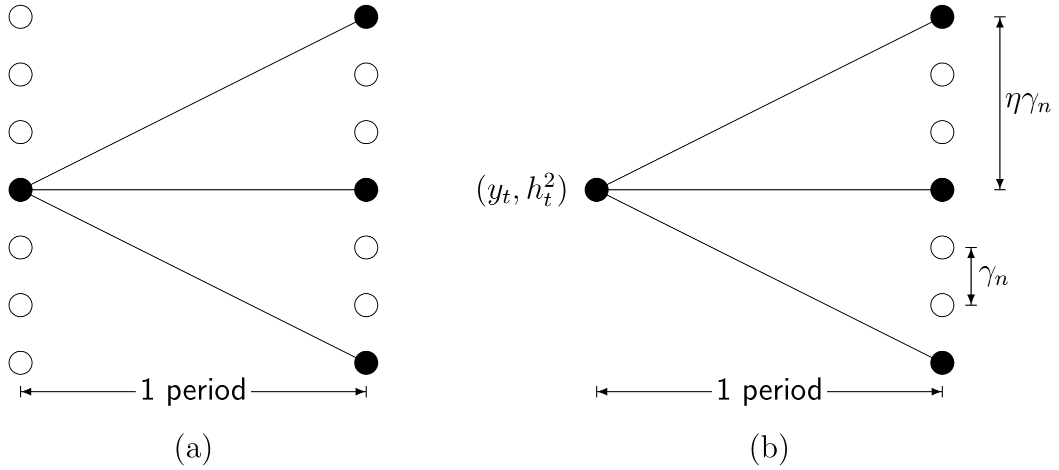

Figure 3: **Jump parameter η and jump size $\eta\gamma_n$.** (a) The tree is laid over a lattice whose vertically adjacent nodes are spaced by γ_n. (b) The two flanking branches fan out around the middle branch to reach the two nodes that are η nodes away from the center. Although $2(\eta - 1)$ hollow nodes are not reached from the node on the left, they may be reached from other nodes. Here $\eta = 3$.

We next pick the jump size for state (y_t, h_t^2). The jump size determines how much the state's three successor states are spaced. As emphasized earlier, the magnitude of the jump size depends on the volatility h_t. By the geometry of the tree, the jump size must be some integer multiple η of γ_n. We call η the jump parameter. The jump parameter measures how much the two flanking branches fan out around the middle branch. It must be large enough for the three branches to match the mean and variance of y_{t+1}. The three nodes one period hence extend over $2\eta + 1$ nodes (inclusively), and inductively the $2n+1$ nodes from (y_t, h_t^2) at date $t+1$ extend over $2n\eta + 1$ nodes (inclusively). See Fig. 3(b) for illustration. The larger the jump parameter η, the larger the tree because it extends over more nodes. The middle branch of the RTCT tree leaves the underlying asset's price unchanged. In contrast, the MT tree lets the middle branch track the mean of y_{t+1}, i.e., $y_t + r - (h_t^2/2)$.

This idea will result in a smaller jump parameter η, thus yielding a more compact tree. Figure 4 illustrates a 1-day trinomial tree with each day partitioned into $n = 3$ periods.

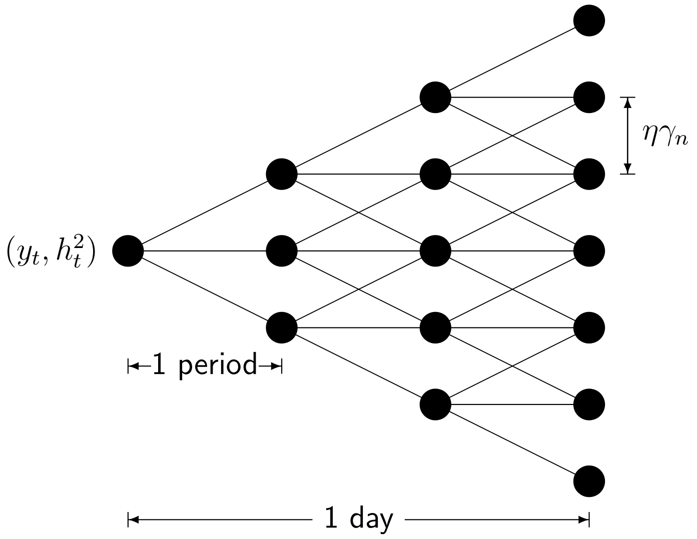

Figure 4: **RTCT trinomial tree for daily logarithmic price y_t for the duration of one day.** A day is partitioned into $n = 3$ periods, and the jump size is $\eta\gamma_n$. The 7 values on the right approximate the distribution of y_{t+1} given (y_t, h_t^2). Recall from Fig. 3 that there are $\eta - 1$ nodes (which are not drawn), between any two vertically adjacent (black) nodes.

The probabilities for the up, middle, and down branches are

$$p_u = \frac{h_t^2}{2\eta^2\gamma^2} + \frac{r - (h_t^2/2)}{2\eta\gamma\sqrt{n}}, \tag{3}$$

$$p_m = 1 - \frac{h_t^2}{\eta^2\gamma^2}, \tag{4}$$

$$p_d = \frac{h_t^2}{2\eta^2\gamma^2} - \frac{r - (h_t^2/2)}{2\eta\gamma\sqrt{n}}. \tag{5}$$

As the branching probabilities are picked to match the mean and variance of y_{t+1} given (y_t, h_t^2) asymptotically, the tree converges to the continuous-state model (1). From Eqs. (3)–(5), it is not hard to verify that valid branching probabilities exist (i.e., $0 \leq p_u, p_m, p_d \leq 1$) if and only if

$$\frac{|\, r - (h_t^2/2)\,|}{2\eta\gamma\sqrt{n}} \leq \frac{h_t^2}{2\eta^2\gamma^2} \leq \min\left(1 - \frac{|\, r - (h_t^2/2)\,|}{2\eta\gamma\sqrt{n}}, \frac{1}{2}\right). \tag{6}$$

The node count can be reduced by a factor of n by getting rid of the intraday nodes. The result is a $(2n+1)$-nomial tree as in Fig. 5; it is multinomial with $2n+1$ branches from every state (y_t, h_t^2). These $2n+1$ successor nodes extend over

$2n\eta+1$ nodes (inclusively). This multinomial tree is the building block of the RTCT tree. Numerical data will demonstrate that although only $2n+1$ of those $2n\eta+1$ nodes are immediately reachable from (y_t, h_t^2), the overwhelming majority of those $2n\eta+1$ nodes are reachable from the root state (y_0, h_0^2). The implication is that the overwhelming majority of the nodes extended over by the RTCT tree are occupied by reachable states.

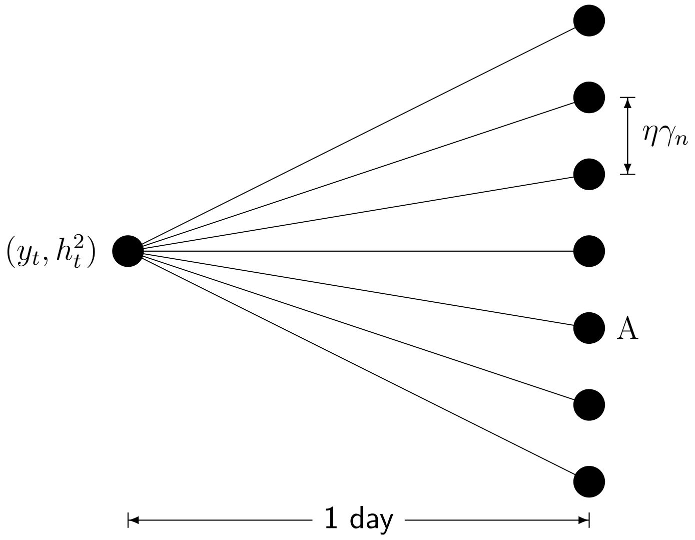

Figure 5: **RTCT multinomial tree for daily logarithmic price y_t for the duration of one day.** This heptanomial tree is the outcome of the trinomial tree in Fig. 4 after removing its intraday nodes. Recall that $n=3$. In general, we infer from Fig. 3 that there are $2n\eta+1$ nodes at date $t+1$ between the top and bottom nodes (inclusive) spaced γ_n apart, but only $2n+1$ of which are reachable from (y_t, h_t^2) and drawn above. The overwhelming majority of those $2n(\eta-1)$ nodes not drawn are reached from the root state as will be demonstrated by numerical data later.

Updating rule (2) must be modified to reflect the adoption of the discrete-state tree model. State (y_t, h_t^2) at date t is now followed by state $(y_t+\ell\eta\gamma_n, h_{t+1}^2)$ at date $t+1$, where

$$\begin{aligned} h_{t+1}^2 &= \beta_0+\beta_1 h_t^2+\beta_2 h_t^2(\epsilon'_{t+1}-c)^2, \\ \epsilon'_{t+1} &= \frac{\ell\eta\gamma_n-(r-h_t^2/2)}{h_t}, \\ \ell &= 0, \pm1, \pm2, \ldots, \pm n. \end{aligned} \tag{7}$$

We will call the resulting tree the *full* RTCT tree. For example, node A in Fig. 5 contains state (y_{t+1}, h_{t+1}^2), where

$$\begin{aligned} y_{t+1} &= y_t+(-1)\eta\gamma_n, \\ h_{t+1}^2 &= \beta_0+\beta_1 h_t^2+\beta_2 h_t^2\left\{\frac{(-1)\eta\gamma_n-[\,r-(h_t^2/2)\,]}{h_t}-c\right\}^2. \end{aligned}$$

As part of a larger RTCT tree, node A may contain other states which are reached from states other than (y_t, h_t^2).

From the underlying trinomial model, the transition from state (y_t, h_t^2) to state $(y_t + \ell\eta\gamma_n, h_{t+1}^2)$ happens with probability

$$P(\ell) \equiv \sum_{j_u, j_m, j_d} \frac{n!}{j_u!\, j_m!\, j_d!}\, p_u^{j_u} p_m^{j_m} p_d^{j_d}, \tag{8}$$

where $j_u, j_m, j_d \geq 0$, $n = j_u + j_m + j_d$, and $\ell = j_u - j_d$. This seemingly complicated formula for probabilities $P(\ell)$ can be calculated very efficiently with the simple generating-function technique in Lyuu (2002) as follows. Note that

$$(p_u x + p_m + p_d x^{-1})^n = \sum_{\ell=-n}^{n} P(\ell)\, x^{\ell}.$$

Therefore, we can multiply out $(p_u x + p_m + p_d x^{-1})^n$ and then retrieve the probabilities $P(\ell)$ by reading off the coefficients of the powers of x. The computation takes $O(n^2)$ steps. Compared with the complex formula (8), the generating-function approach is straightforward and stable.

As volatility h_t changes through time, we may have to pick different jump parameters η for different states so that all branching probabilities p_u, p_m, and p_d lie between 0 and 1. This entails varying jump sizes. As the necessary requirement $p_m \geq 0$ implies

$$\eta \geq h_t/\gamma, \tag{9}$$

RTCT goes through

$$\eta = \lceil h_t/\gamma \rceil, \lceil h_t/\gamma \rceil + 1, \lceil h_t/\gamma \rceil + 2, \ldots \tag{10}$$

until valid probabilities are obtained or until their nonexistence is confirmed by inequalities (6). The latter case means the tree cannot grow further. Later proofs depend critically on noting that the magnitude of η grows at least as fast as h_t by virtue of inequality (9). Hence when h_t grows exponentially, the resulting tree must do likewise.

Every node at date t on the tree carries a different logarithmic price y_t. However, more than one path from the root state (y_0, h_0^2) may lead to the same node, each yielding a different squared volatility h_t^2. The number of possible values of h_t^2 at a node thus equals the number of paths reaching the node. Each h_t^2 picks its own jump parameter η. Figure 6 depicts a 3-day tree with $n = 1$. Nodes A and B each have one h_t^2, whereas node C has two. The h_t^2 at nodes A and B pick the same jump parameter $\eta = 2$, whereas the two h_t^2 at node C pick different jump parameters $\eta = 1, 2$. The overall tree structure is irregular because of the varying jump parameters. Hollow nodes in Fig. 6 are not occupied because they are unreachable from the root state

(y_0, h_0^2). As will be shown later, their count is minuscule. This outcome is expected by observing node C. Although the three branches from node A do not reach node C, node C is nevertheless reached from the two nodes below node A. Another factor contributing to making unreachable nodes few is the fact that most nodes contain more than one squared volatility h_t^2. If these squared volatilities pick different jump parameters such as those at nodes C and D, more than three branches will emanate from the node. More branches make more nodes reachable.

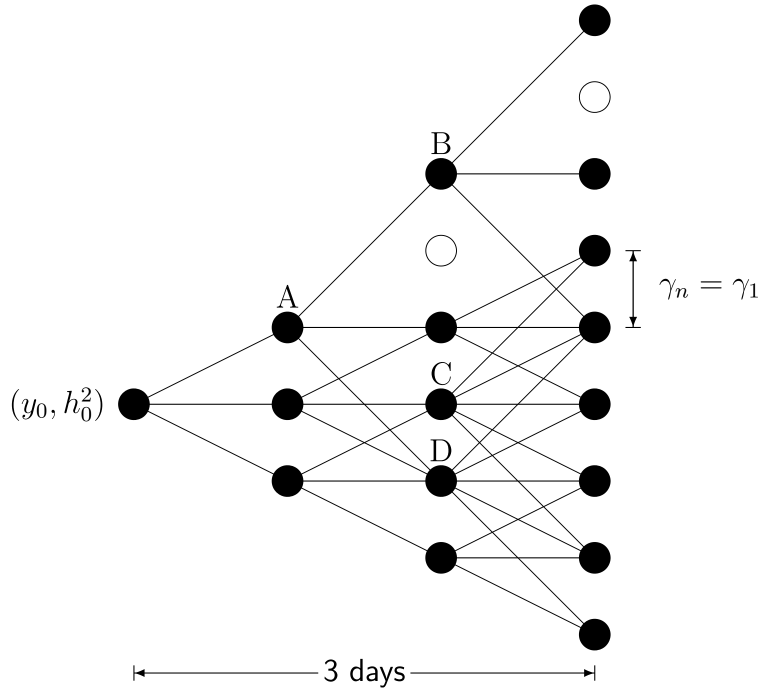

Figure 6: **Geometry of a 3-day RTCT tree.** A day is partitioned into $n = 1$ period. Nodes A and B have a jump parameter of 2. Nodes C with two h_t^2 and D with three h_t^2 have two jump parameters: 1 and 2. All other nodes have a jump parameter of 1. Hollow nodes are not reachable from the root state (y_0, h_0^2).

4 Interpolated Volatilities and Backward Induction

The number of possible volatilities h_t at a node equals the number of paths reaching it. An algorithm that keeps all of these volatilities cannot be efficient as their count is exponential. Therefore, the full RTCT tree (7) must be approximated. The standard approximation methodology by Hull and White (1993a) and Ritchken, Sankarasubramanian, and Vijh (1993) is adopted by both RTCT and, later, MT. The actual RTCT tree now keeps only the maximum volatility $h_{\max}$ and the minimum volatility $h_{\min}$

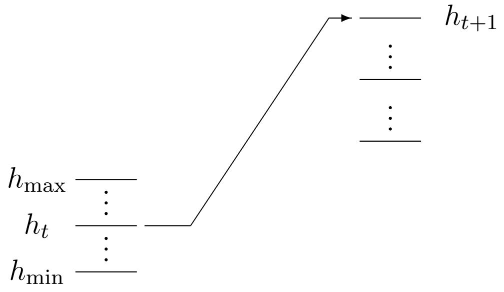

Figure 7: **Case where maximum volatility follows an interpolated volatility.** The maximum volatility h_{t+1} at the node on the right follows interpolated volatility h_t.

at each node. It then creates $K-2$ volatilities between $h_{\min}$ and $h_{\max}$ (what values these volatilities assume will be elaborated soon). Let us call these $K-2$ volatilities interpolated volatilities because they are not the results of actually following updating rule (7). Instead, they are artificial volatilities generated via interpolation. The number of volatilities per node is now a constant K.

CT and RT differ in the way they calculate $h_{\min}$ and $h_{\max}$ at each node. RT calculates them with the interpolated volatilities taken into consideration. This means that every node at date i generates $2n+1$ volatilities for each of its K volatilities via the updating rule. The number of volatilities generated per node is hence $K(2n+1)$. These volatilities together with their associated branches then determine the $h_{\min}$ and $h_{\max}$ at every node at date $i+1$. CT, on the other hand, calculates $h_{\min}$ and $h_{\max}$ in the tree-building process without the interpolated volatilities. This means that every node at date i generates $2n+1$ volatilities for each of its two volatilities $h_{\min}$ and $h_{\max}$ via the updating rule. The number of volatilities generated per node is hence only $2(2n+1)$. These volatilities together with their associated branches then determine the $h_{\min}$ and $h_{\max}$ at every node at date $i+1$. Both CT and RT use interpolated volatilities in backward induction.

The interpolated volatilities between $h_{\min}$ and $h_{\max}$ at a node are artificial because they are not the results of applying updating rule (7) starting from the root state (y_0, h_0^2) at time 0. For the same reason, $h_{\min}$ and $h_{\max}$ in CT are not artificial as they are the results of following chains of updating rule without interpolation. For RT, even $h_{\min}$ and $h_{\max}$ may be artificial if they are the results of applying the updating rule to interpolated volatilities of the previous date (see Fig. 7 for illustration).

In RTCT, the K squared volatilities at a node are equally spaced between $h_{\min}^2$ and $h_{\max}^2$:

$$h_{\min}^2 + j\,\frac{h_{\max}^2 - h_{\min}^2}{K-1}, \quad j = 0, 1, 2, \ldots, K-1.$$

We call it the linear interpolation scheme. A different distribution will be used by MT.

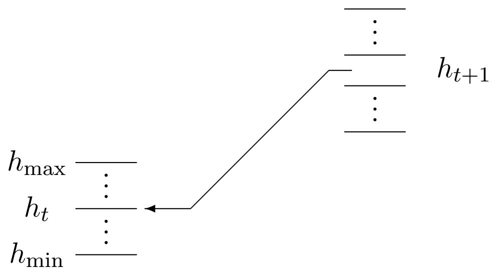

Figure 8: **Backward induction.** Volatility h_{t+1} follows h_t by the updating rule. Because h_{t+1} does not match any interpolated volatility, its corresponding option value is found by interpolating from the two option values whose volatilities bracket it.

To be specific, the K logarithms of squared volatilities are equally spaced between $\ln h_{\min}^2$ and $\ln h_{\max}^2$ in MT; they are

$$\exp\left[\ln h_{\min}^2 + j\,\frac{\ln h_{\max}^2 - \ln h_{\min}^2}{K-1}\right], \quad j = 0, 1, 2, \ldots, K-1.$$

Smaller volatilities are thus sampled more finely than larger volatilities. We call it the log-linear interpolation scheme.

After the tree is built, backward induction commences. For the volatility h_{t+1} following state (y_t, h_t^2) via updating rule (7), RTCT finds the two volatilities that bracket h_{t+1}. The option price corresponding to h_{t+1} is then interpolated linearly from the option prices corresponding to the bracketing volatilities. Figure 8 illustrates this procedure for a branch. After the option prices from all $2n+1$ branches are available, the option price for state (y_t, h_t^2) is calculated as their average discounted value weighted by the branching probabilities.

Before closing this section, we remark that CT is under-specified. CT maintains only $h_{\min}$ and $h_{\max}$ at each node in growing the tree. The interpolated volatilities at each node are not involved in the tree-building process. Only in backward induction are the $K-2$ interpolated volatilities added to approximate the full RTCT tree. It is therefore possible for an interpolated volatility's successor volatility to reach a node that is not reached during the tree-building process and thus has no option prices at all. Take Fig. 8 for example. If h_{t+1} lies in a node not reached during tree building, the node will not have option values at all and backward induction must be terminated. Rare as these situations are, they do arise when n and N are both large. RT, in contrast, has no such problems because the same volatilities (interpolated or otherwise) used in building the tree are used in backward induction. In other words, the interpolated volatilities' branches have all been followed through in the tree-building process. MT follows RT in using all interpolated volatilities, not just $h_{\min}$ and $h_{\max}$, in growing the tree.

5 Sufficient Condition for RTCT To Explode

It is common practice to raise numerical accuracy by increasing n. Unfortunately, this section proves that the largest value of h_t at date t grows exponentially in t if n exceeds a threshold. When this happens, the value of η must also grow exponentially by virtue of inequality (9). Note that the $2n+1$ successor nodes of a state extend over $2n\eta+1$ nodes (recall Fig. 5). So when η grows exponentially, the RTCT tree explodes. We conclude that the RTCT tree must be restricted to small n to have any hope of being efficient. A small n is no guarantee that the tree will not explode, however, because our threshold for explosion is a sufficient condition only. Hence it does not follow that an n not exceeding that threshold escapes explosion. Because RTCT employs a search procedure (10), finding the *necessary* condition for explosion seems difficult.

We now provide the simple argument for the exponential growth of the largest value of h_t at date t for RTCT. Assume $r=0$ and $c=0$. Updating rule (7) is now

$$h_{t+1}^2 = \beta_0 + \beta_1 h_t^2 + \beta_2 \left[\, \ell\eta\gamma_n + (h_t^2/2) \,\right]^2, \quad \ell = 0, \pm 1, \pm 2, \ldots, \pm n.$$

Set $\ell = n$ to make h_{t+1}^2 as large as possible. The updating rule now becomes

$$\begin{aligned} h_{t+1}^2 &= \beta_0 + \beta_1 h_t^2 + \beta_2 \left[\, \sqrt{n}\,\eta\gamma + (h_t^2/2) \,\right]^2 \quad \text{as } \gamma_n = \gamma/\sqrt{n} \\ &\geq \beta_0 + \beta_1 h_t^2 + \beta_2 \left[\, \sqrt{n}\, h_t + (h_t^2/2) \,\right]^2 \quad \text{as } \eta\gamma \geq h_t \\ &\geq \beta_0 + \beta_1 h_t^2 + \beta_2 n h_t^2 \\ &= \beta_0 + (\beta_1 + \beta_2 n)\, h_t^2. \end{aligned}$$

By induction,

$$\begin{aligned} h_{t+1}^2 &\geq \beta_0 \sum_{i=0}^{t} (\beta_1 + \beta_2 n)^i + (\beta_1 + \beta_2 n)^{t+1} h_0^2 \\ &= \frac{\beta_0}{1 - (\beta_1 + \beta_2 n)} + \left[h_0^2 + \frac{\beta_0}{(\beta_1 + \beta_2 n) - 1} \right] (\beta_1 + \beta_2 n)^{t+1}. \end{aligned}$$

The above expression grows exponentially in t if

$$\beta_1 + \beta_2 n > 1. \tag{11}$$

Interestingly, this inequality is reminiscent of the necessary condition $\beta_1 + \beta_2 \geq 1$ for GARCH to be nonstationary.

When $r \neq 0$ or $c \neq 0$, the largest value of h_t at date t still grows exponentially in t as long as n is suitably large. The argument is more tedious but essentially identical. We conclude that the RTCT tree grows exponentially if n is large enough.

6 The Shallowness of the Exploding RTCT Tree

Can a large n be picked assuming we are willing to accept long running times? Unfortunately, as will be shown below, RTCT does not admit such a tradeoff between accuracy and speed because it will be forced to be short-dated when explosion occurs. Consequently, even if one is willing to accept inefficiency, the RTCT tree may not grow to the desired maturity date. An exploding RTCT tree is therefore of very limited use in practice for reasons other than exponential complexity.

The technical reason for shallowness is that there is a global ceiling on volatility h_t for valid branching probabilities to exist: $h_t^2 \leq 2(r+n)+2\sqrt{2rn+n^2}$. This upper bound depends only on r and n. So when the largest value of h_t grows exponentially in t, this ceiling will be quickly reached at a small t and the tree can grow no further. We next derive the promised upper bound that leads to the impossibility result.

Inequalities (6) imply

$$\begin{aligned}\frac{|\,(h_t^2/2)-r\,|}{2\eta\gamma\sqrt{n}} &\leq \frac{h_t^2}{2\eta^2\gamma^2},\\ \frac{h_t^2}{2\eta^2\gamma^2} &\leq \frac{1}{2}.\end{aligned}$$

Hence

$$h_t^2 \leq (\eta\gamma)^2 \leq \left[\frac{h_t^2\sqrt{n}}{|\,(h_t^2/2)-r\,|}\right]^2,$$

which can be simplified to yield

$$\left[\,(h_t^2/2)-r\,\right]^2 \leq nh_t^2.$$

The above quadratic inequality (in h_t^2) is equivalent to

$$2(r+n)-2\sqrt{2rn+n^2} \leq h_t^2 \leq 2(r+n)+2\sqrt{2rn+n^2}\,.$$

We conclude that

$$h_t^2 \leq 2(r+n)+2\sqrt{2rn+n^2} \tag{12}$$

is necessary for the existence of valid branching probabilities for RTCT. As promised earlier, the bound on h_t^2 does not depend on the choice of γ as the equation $\gamma = h_0$ never enters the analysis.

This impossibility finding may sound puzzling at first. After all, under the Black-Scholes model, valid branching probabilities always exist as long as n is suitably large. Why, one may ask, can't the same property hold here? The answer lies in the volatility process. The daily volatility in the Black-Scholes model is a constant, which amounts to setting h_t to some fixed number. So every state solves the same Eqs. (3)–(5) for the probabilities, and increasing n will eventually have inequality (12) satisfied for all states. In contrast, the volatility fluctuates under GARCH. Thus each state (y_t, h_t^2) faces different Eqs. (3)–(5) in solving for the probabilities. Increasing n beyond a certain threshold makes it impossible to satisfy inequality (12) when t is suitably large. The reason is that the largest h_t^2 grows exponentially in t, eventually breaching the upper bound at a suitably large t.

7 Numerical Evaluation of RTCT

The following parameters from Ritchken and Trevor (1999) and Cakici and Topyan (2000) will be assumed throughout the section unless stated otherwise: $S_0 = 100$, $r = 0$, $h_0^2 = 0.0001096$, $\gamma = 0.010469$, $\beta_0 = 0.000006575$, $\beta_1 = 0.9$, $\beta_2 = 0.04$, and $c = 0$. As $r = c = 0$, criterion (11) says combinatorial explosion occurs when

$$n > \frac{1 - 0.9}{0.04} = 2.5.$$

Figure 9 confirms the exponential growth of the RTCT tree with $n = 3, 4, 5$. Note that the total node count increases with n. For comparison, the standard trinomial tree contains only $2t + 1$ nodes at date t.

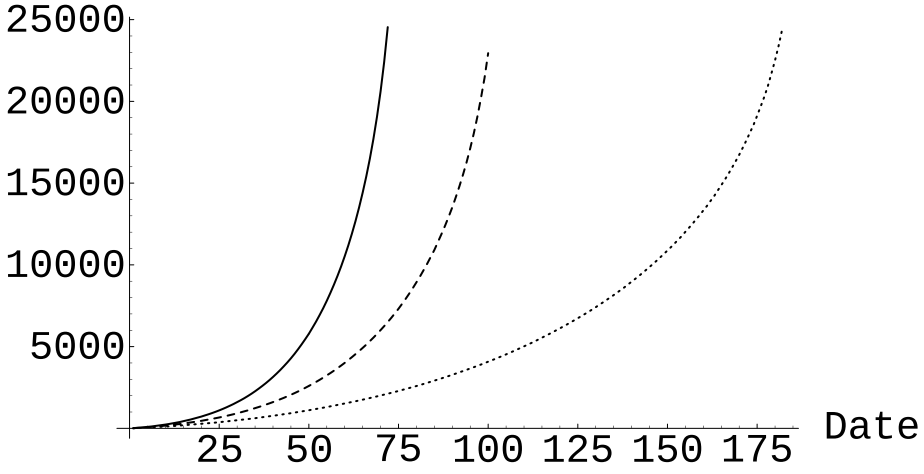

Figure 9: **Exponential growth of the RTCT tree.** Each curve plots the number of nodes at date t of the tree. The dotted line is based on $n = 3$, the dashed line on $n = 4$, and the solid line on $n = 5$. The parameters are $S_0 = 100$, $r = 0$, $h_0^2 = 0.0001096$, $\gamma = 0.010469$, $\beta_0 = 0.000006575$, $\beta_1 = 0.9$, $\beta_2 = 0.04$, and $c = 0$.

The number of nodes (call it M) is critical because the running time is proportional to KM. We mention earlier that there may be nodes extended over by the tree which are not reachable and are depicted as hollow nodes in Fig. 6. In principle, if such nodes are numerous, RTCT can potentially run more efficiently by adopting clever programming techniques to skip them. But Figure 10 shows that the proportion of unreachable nodes is small for $n = 3, 4, 5$. We will see shortly that the same conclusion holds for larger n as well. As the overwhelming majority of nodes are reachable, skipping unreachable ones brings no substantial benefits. Therefore, by tree size we shall mean the number of nodes extended over by the tree regardless of reachability.

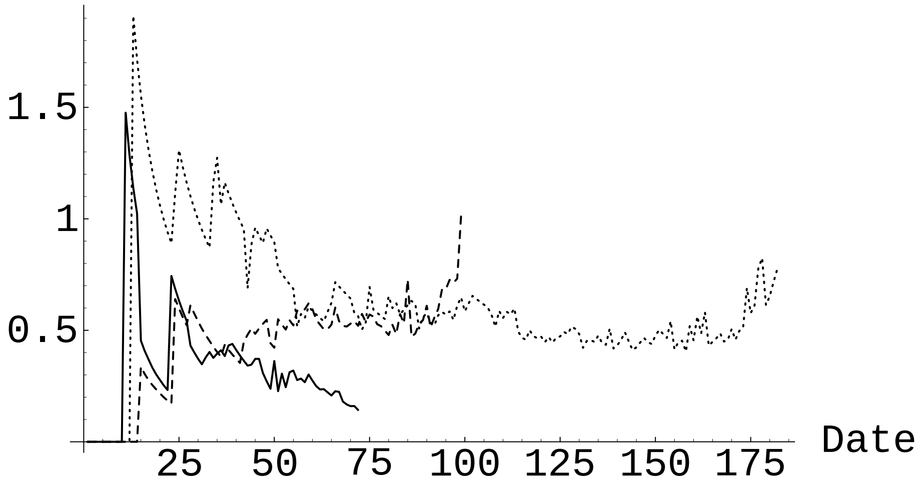

Figure 10: **The percent of unreachable nodes.** The plots show the percent of unreachable nodes among all nodes at each date. The dotted line is based on $n = 3$, the dashed line on $n = 4$, and the solid line on $n = 5$. The number of unreachable nodes is insignificant in all 3 lines. The parameters are $S_0 = 100$, $r = 0$, $h_0^2 = 0.0001096$, $\gamma = 0.010469$, $\beta_0 = 0.000006575$, $\beta_1 = 0.9$, $\beta_2 = 0.04$, and $c = 0$.

Suppose we select $n = 100$ to seek very high accuracy at the expense of efficiency. The theory predicts that the RTCT tree's final maturity will be cut short. Indeed, with $r = 0$, inequality (12) imposes the universal upper bound of $h_t^2 \leq 4n = 400$. This means a node with $h_t > 20$ cannot have valid branching probabilities and thus the tree cannot grow beyond date t. As this ceiling is breached somewhere at date 9 because of the exponential growth of the largest value of h_t, the tree stops growing there. Table 1 lists the final dates under various n. Observe that the tree's final maturity date decreases rapidly as n increases. For example, it is 72 with $n = 5$, 34 with $n = 10$, and 12 with $n = 50$. To be useful, n cannot be so large as to make the tree's final maturity date fall short of the derivative's. This concern sets a firm upper bound on n. Table 1 also tabulates the total numbers of nodes and unreachable ones among them. Again, the overwhelming majority of the nodes are reachable.

Some of the calculated option prices in Ritchken and Trevor (1999) use n as large as 25 and option maturity dates as far as 200. These choices contradict our analysis and data. For example, according to Table 1, the RTCT tree with $n = 25$ stops growing as early as date 18. Therefore, some of their prices should be viewed with caution.

Cakici and Topyan (2000) used $n = 1$ throughout their paper. Explosion is avoided for their choice of parameters, which are also used in this section. But there are two problems with this choice of n. First, CT does not always generate accurate option prices with $n = 1$. For example, suppose we select $\beta_0 = 0.000007$ instead of 0.000006575. The option prices in Table 2 show that CT deviates from the Monte Carlo estimates. Second, it is not true that the choice $n = 1$ always escapes explosion. To support this point, we use the same parameters as Fig. 9 except for $c = 2$ and $n = 1$. The resulting CT tree explodes as plotted in Fig. 11.

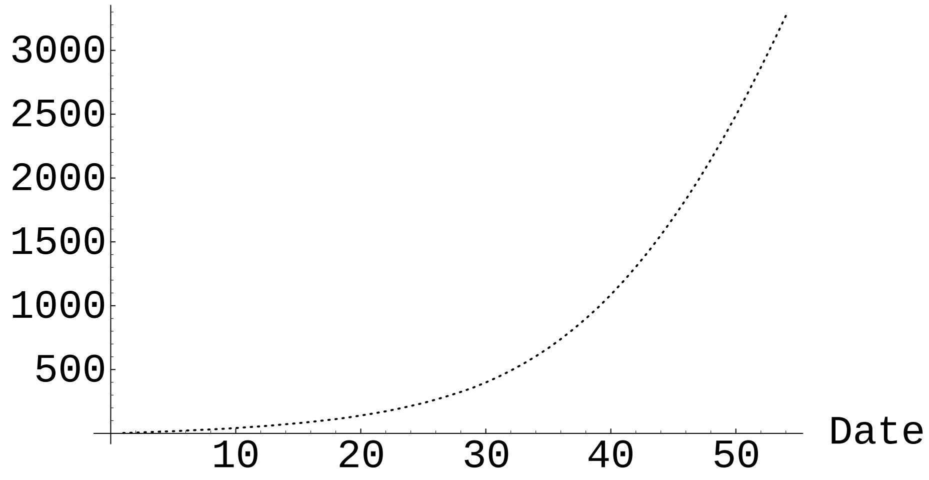

Figure 11: **Exponential growth of the CT tree with $n = 1$.** This CT tree explodes. The final maturity date of the tree is 54. The parameters are $S_0 = 100$, $r = 0$, $h_0^2 = 0.0001096$, $\gamma = 0.010469$, $\beta_0 = 0.000006575$, $\beta_1 = 0.9$, $\beta_2 = 0.04$, $c = 2$, and $n = 1$.

One of the most critical questions to ask of an approximation algorithm such as RTCT is whether it converges with increasing n. Table 3 and Figure 12 show that there is a downward trend in the calculated option prices except for the very short maturity of $N = 2$ days. Moreover, the downward trend accelerates as n increases. To rule out the possibility that the problem originates from the discrete-state full RTCT tree approximation of the continuous-state model, Monte Carlo simulation of the full RTCT tree is carried out. The data in Table 3 demonstrate that the tree model produces Monte Carlo estimates generally consistent with those of the continuous-state model. Hence the problem lies mainly with the way the full RTCT tree is in turn approximated. We will see later that a major reason causing RTCT to deviate numerically from the true option price is its use of the linear volatility interpolation scheme.

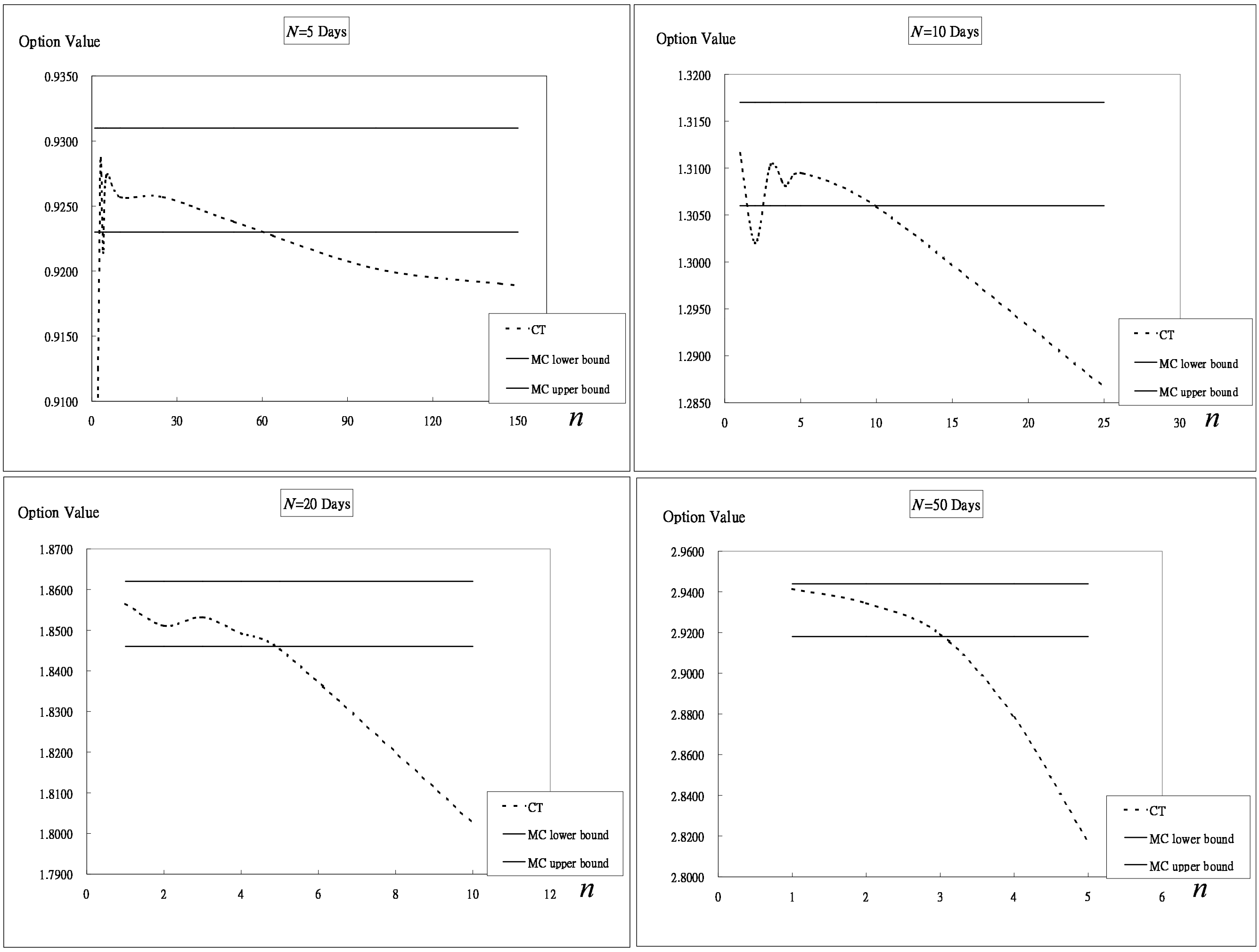

Figure 12: **Select option prices from Table 3**. `MC lower bound` equals ${}^{\infty}L$; `MC upper bound` equals ${}^{\infty}U$.

8 The Mean-Tracking (MT) Tree

To recap, RTCT has at least four weaknesses. First, it explodes exponentially when n exceeds a threshold. Second, it is not known whether there is a simple formula for the threshold n^* so that the RTCT tree escapes explosion as long as $n \leq n^*$. Third, when explosion happens, the tree is cut short, making it unable to price derivatives with a longer maturity date. Fourth, option prices may fail to converge as n increases. We next turn to MT that addresses these problems. MT makes two critical changes to RTCT. The first is to replace the linear interpolation scheme with the log-linear interpolation scheme. This addresses the convergence problem mentioned earlier. The second is to let the middle branch of the multinomial tree track the mean of y_{t+1}. This addresses the explosion problem and its consequence of shortened maturity. It also yields a simple formula for the above-mentioned n^*. Surprisingly, MT's tree size grows only quadratically when $n \leq n^*$. This result makes the MT tree size asymptotically the same as that of the CRR binomial tree. The analysis of MT will turn out to require only elementary techniques.

8.1 Volatility Interpolation Schemes

The distribution of the volatilities reaching a node plays a key role in pricing accuracy. RTCT essentially assumes that the distribution is uniform: Interpolated squared volatilities are equally spaced between the minimum and the maximum squared volatilities. Figure 13, however, shows that the actual distribution is closer to a lognormal distribution than a uniform one. It strongly suggests that there be more interpolated volatilities at the lower values than at the higher values. This is the rationale for MT's adopting the log-linear interpolation scheme, in which the logarithmic volatilities are equally spaced. The log-linear scheme is also used by the Markov chain approximation of Duan and Simonato (2001) for the same numerical considerations. Similar findings in the case of the Asian option exist. For example, Dai (2004) proved that linear interpolation schemes result in overestimates, and Forsyth, et al. (2002) demonstrated that linear interpolation schemes may not converge to the correct price.

8.2 Tree Building

At date t, let node A be the node closest to the mean of y_{t+1} given (y_t, h_t^2), which equals $y_t + r - (h_t^2/2)$. For convenience, we use μ to denote this conditional mean minus the current logarithmic price:

$$\mu \equiv r - \frac{h_t^2}{2}$$

(see Fig. 14). By the geometry of the tree, node A's logarithmic price equals $y_t + a\gamma_n$ for some integer a. The criterion by which node A is chosen ensures that

$$|\ a\gamma_n - \mu\ | \leq \frac{\gamma_n}{2}. \tag{13}$$

To create the desired multinomial tree, make the middle branch of the $(2n+1)$-nomial tree line up with node A as in Fig. 15. Although a node reaches only $2n+1$ nodes after one day, the top and bottom nodes extend over

$$2n\eta + 1 \tag{14}$$

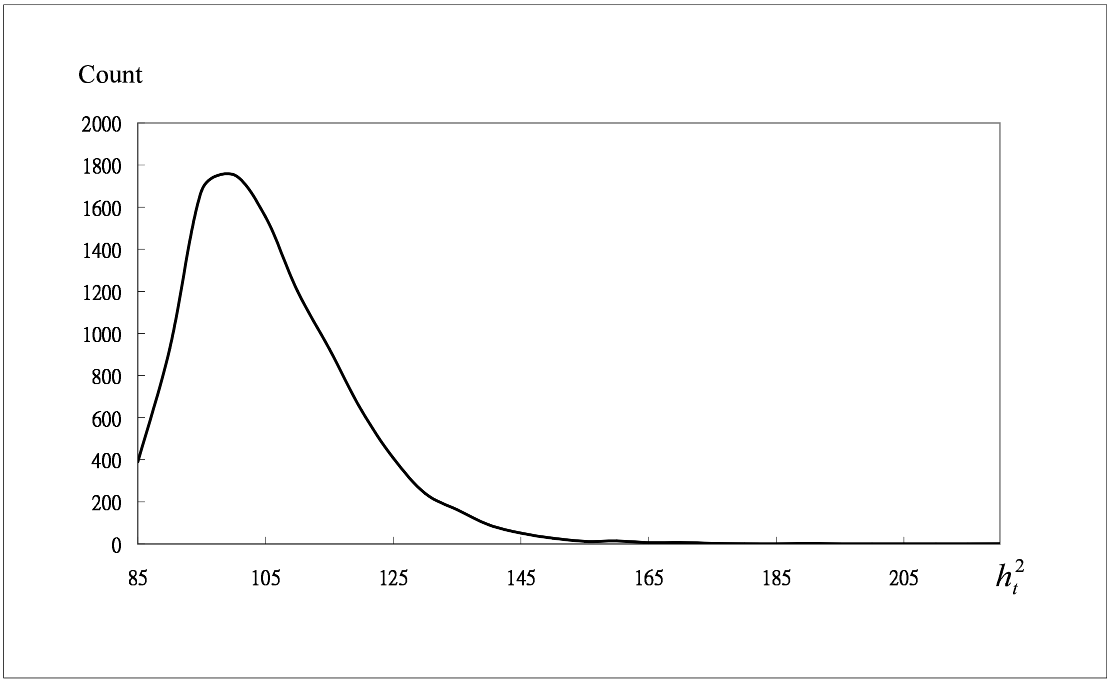

Figure 13: **Histogram of volatility distribution**. All parameters are from Table 3 with $n = 10$ and $N = 10$. Monte Carlo simulation of the full RTCT tree (7) is used to record the volatilities at the middle node at maturity. Out of 500,000 paths, about 10,000 reach the node. The recorded minimum and maximum squared volatilities are 0.000085 and 0.000217, respectively. Squared volatilities h_t^2 are multiplied by 1,000,000 for easier reading.

nodes as in RTCT. Amazingly, as will be demonstrated later, the overall node count is only quadratic in N if n does not exceed a threshold. The probabilities for the upward, middle, and downward branches equal

$$\begin{aligned}
p_u &= \frac{nh_t^2 + (a\gamma_n - \mu)^2}{2n^2\eta^2\gamma_n^2} - \frac{a\gamma_n - \mu}{2n\eta\gamma_n}, \\
p_m &= 1 - \frac{nh_t^2 + (a\gamma_n - \mu)^2}{n^2\eta^2\gamma_n^2}, \\
p_d &= \frac{nh_t^2 + (a\gamma_n - \mu)^2}{2n^2\eta^2\gamma_n^2} + \frac{a\gamma_n - \mu}{2n\eta\gamma_n}.
\end{aligned}$$

As they match the mean and variance of the GARCH process at date $t + 1$ asymptotically, convergence is guaranteed. State (y_t, h_t^2) at date t is followed by state

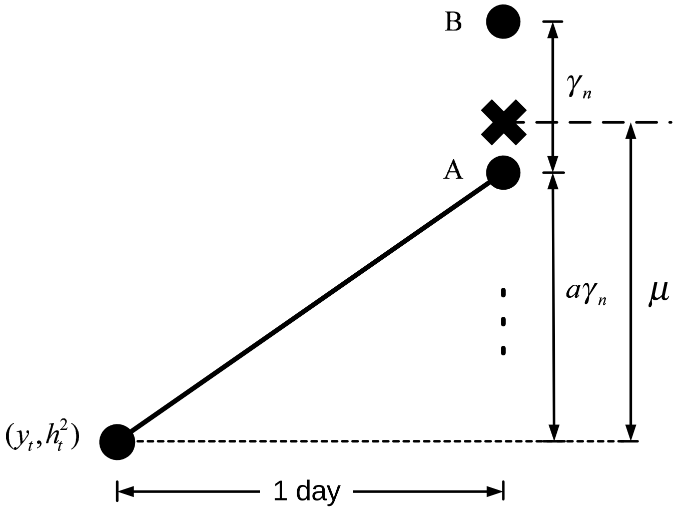

Figure 14: **The next middle node via mean tracking.** The cross identifies the true mean of y_{t+1}. Two nodes, A and B, bracket it. Between them, node A has a logarithmic price closer to the mean. The number $a\gamma_n$ denotes the difference between y_t and node A's logarithmic price.

$(y_t + \ell\eta\gamma_n, h_{t+1}^2)$ at date $t+1$, where

$$\begin{aligned} h_{t+1}^2 &= \beta_0 + \beta_1 h_t^2 + \beta_2 h_t^2 (\epsilon''_{t+1} - c)^2, \\ \epsilon''_{t+1} &= \frac{\ell\eta\gamma_n + a\gamma_n - (r - h_t^2/2)}{h_t}, \\ \ell &= 0, \pm 1, \pm 2, \dots, \pm n. \end{aligned} \tag{15}$$

We will call the resulting tree the *full* MT tree. Formulas for the transition probabilities are stated in equation (8). As the full MT tree contains an exponential number of states, it must be approximated. But unlike in the RTCT case, the approximated MT will be efficient, accurate, and never cut short.

The conditions for the probabilities to lie within 0 and 1, i.e., $0 \leq p_u, p_m, p_d \leq 1$, are

$$\frac{|\, a\gamma_n - \mu \,|}{2n\eta\gamma_n} \leq \frac{nh_t^2 + (a\gamma_n - \mu)^2}{2n^2\eta^2\gamma_n^2}, \tag{16}$$

$$\frac{nh_t^2 + (a\gamma_n - \mu)^2}{2n^2\eta^2\gamma_n^2} \leq \frac{1}{2}, \tag{17}$$

$$\frac{nh_t^2 + (a\gamma_n - \mu)^2}{2n^2\eta^2\gamma_n^2} \leq 1 - \frac{|\, a\gamma_n - \mu \,|}{2n\eta\gamma_n}. \tag{18}$$

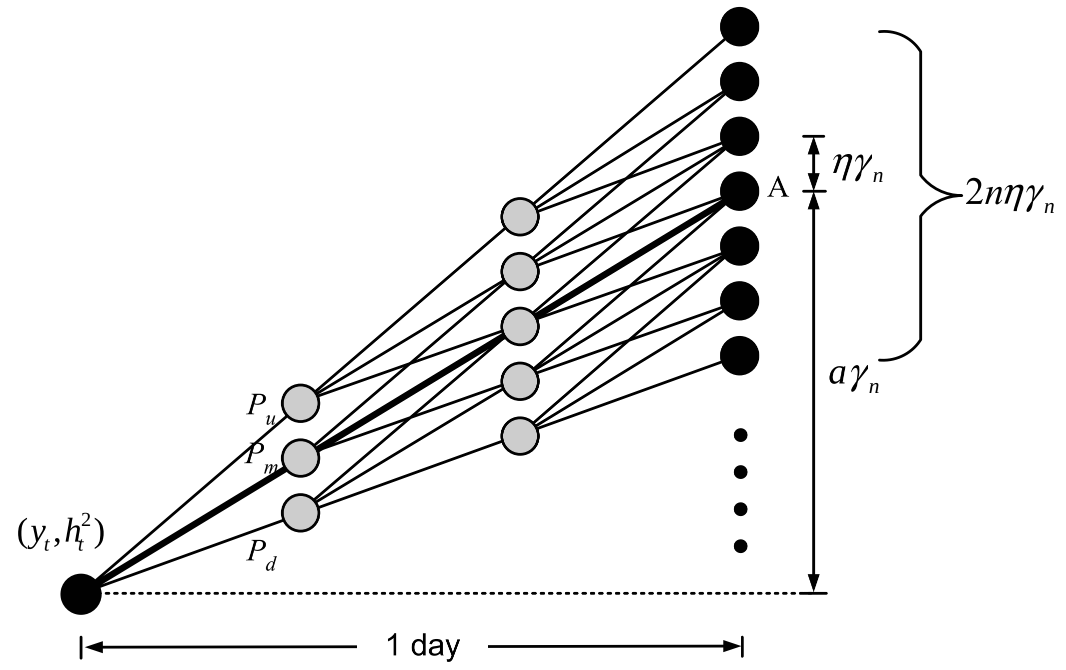

Figure 15: **The MT multinomial tree for daily logarithmic price y_t for a duration of one day.** A day is partitioned into $n = 3$ periods, and the three jump sizes in each period are $(a\gamma_n/n)+\eta\gamma_n$ (upward), $a\gamma_n/n$ (middle), and $(a\gamma_n/n)-\eta\gamma_n$ (downward). The central branch of the tree lines up with node A, the node closest to the mean of y_{t+1} as stated in Fig. 14. The solid nodes are actually used in pricing, but the gray nodes are for illustration only. This heptanomial MT tree should be compared with the RTCT counterpart in Fig. 5.

Inequalities (16)–(17) are equivalent to

$$\frac{\sqrt{nh_t^2 + (a\gamma_n - \mu)^2}}{n\gamma_n} \leq \eta \leq \frac{nh_t^2 + (a\gamma_n - \mu)^2}{n\gamma_n |\, a\gamma_n - \mu \,|}. \tag{19}$$

Inequalities (13) and (17) together imply inequality (18) because

$$\frac{nh_t^2 + (a\gamma_n - \mu)^2}{2n^2\eta^2\gamma_n^2} + \frac{|\, a\gamma_n - \mu \,|}{2n\eta\gamma_n} \;\leq\; \frac{1}{2} + \frac{1}{4n\eta} \;\leq\; 1.$$

Hence the probabilities are valid if and only if the much simpler inequalities (19) hold.

MT can avoid the short-maturity problem of RTCT. Let $H_{\min}^2 \equiv \min(h_0^2, \beta_0/(1-\beta_1))$ to make $H_{\min}^2 \leq h_t^2$ for $t \geq 0$ (see Appendix A for justification). Now, a valid integer η can always be found regardless of the value of n as long as γ_n is less than some constant. More precisely, interval (19) contains positive integers for the jump parameter η to take its value in when

$$\gamma_n^2 \leq H_{\min}^2 \tag{20}$$

as proved in Appendix B. With the existence of η guaranteed, MT will never be cut short, as promised.

Rather than searching for an η to satisfy inequalities (19), MT simply sets

$$\eta = \left\lceil \frac{\sqrt{nh_t^2 + (a\gamma_n - \mu)^2}}{n\gamma_n} \right\rceil . \tag{21}$$

Although other choices are clearly possible, this particular choice is amenable to the analysis on the size of the MT tree later.

We proceed to choose γ, hence γ_n as well because $\gamma_n = \gamma/\sqrt{n}$. If $\gamma \leq H_{\min}$, then γ_n satisfies inequality (20) for all n. A smaller γ generally leads to larger trees, hence longer running times. On the other hand, a smaller γ is expected to result in better accuracy because of the finer grain. To strike an overall balance between accuracy and convergence speed, we set $\gamma = H_{\min}/2$; thus

$$\gamma_n = \frac{H_{\min}}{2\sqrt{n}}. \tag{22}$$

All the parameters of MT have now been specified.

9 A Sufficient Condition for the Nonexplosion of MT

In practice, it is essential to know beforehand that the chosen n will not result in an exploding tree before tree building is even attempted. Without this knowledge, tree building may take a long time if the tree turns out to explode and may even end up with a tree not meeting the required maturity if shortened maturity is an issue. In the case of MT, the criterion for nonexplosion is simple: The MT tree does not explode if

$$n \leq \left(\sqrt{\frac{1-\beta_1}{\beta_2}} - c \right)^2 . \tag{23}$$

Note that the threshold is independent of r. When inequality (23) holds, the tree-size growth is only quadratic in maturity, the same as the CRR tree. (See Appendix C for proof.) The MT tree is thus the first tree-based GARCH option pricing algorithm that is provably efficient. This surprising finding makes MT a very practical algorithm.

Earlier, we show that the RTCT tree explodes if n exceeds some threshold; in fact, the threshold is $(1-\beta_1)/\beta_2$ when $c = r = 0$. But we are silent about the tree size when that threshold is not breached. The positive result (23) for MT fills that void because it says the MT tree is efficient if n does not exceed an explicit threshold. Surprisingly, the sufficient condition for MT's nonexplosion reduces to $n \leq (1-\beta_1)/\beta_2$ when $c = 0$. As the same $(1-\beta_1)/\beta_2$ is the threshold of explosion for RTCT, criterion (23) can be said to be tight.

10 Numerical Evaluation of MT

RTCT can be inaccurate in pricing options. This fact is documented in Table 4. In every combination of n and K, RTCT deviates from the simulation results. In contrast, MT produces prices within the 95% confidence interval for $K \geq 2$ with $n = 1$, $K \geq 10$ with $n = 2$, and $K \geq 50$ with $n = 3, 4$. Hence MT succeeds where RTCT fails. Observe that K needs to be increased for a larger n because the resulting increase in the number of volatilities per node demands more resolution.

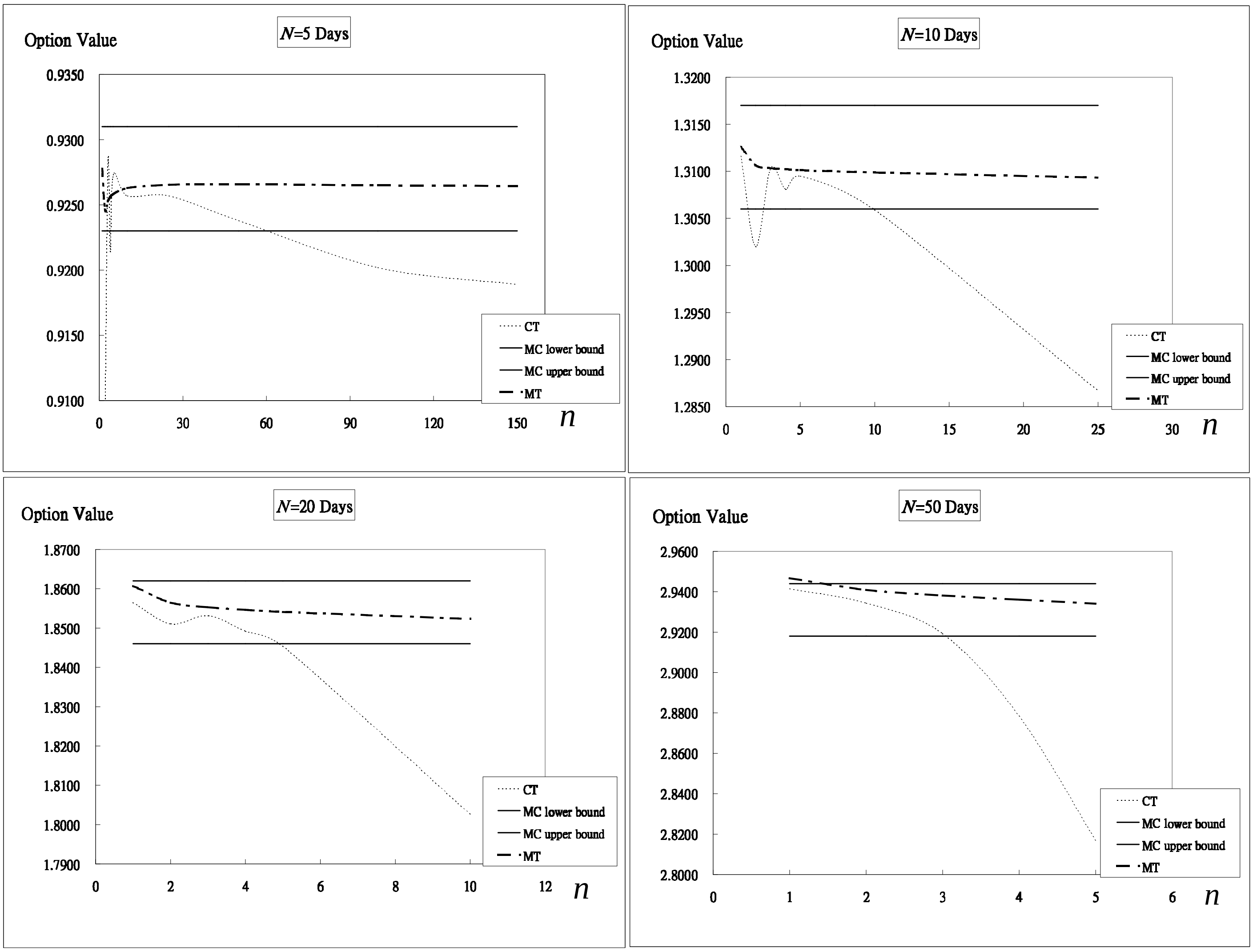

Figure 16: **Select option prices from Table 5**. `MC lower bound` equals ${}^{\infty}L$; `MC upper bound` equals ${}^{\infty}U$. The option prices under the CT algorithm come from Fig. 12.

Next we benchmark MT's performance with increasing n. For this purpose, we duplicate the settings in Table 3 for RTCT to produce Table 5 for MT. Select prices are plotted in Fig. 16 for graphical depiction. Unlike RTCT, all prices generated by MT are within the 95% confidence interval of Monte Carlo simulation of the full MT tree (15). Furthermore, MT provides results very close to the true option price even with a small n. MT therefore outperforms RTCT in terms of overall accuracy and convergence speed.

All the numerical experiments up to now assume $r = c = 0$. In Table 6, we investigate MT's options pricing accuracy with nonzero r and c, specifically, $r = 5\%$ (annual) and $c = 0.5$ from Duan and Simonato (2001). Although a few of the computed option prices are outside the 95% confidence interval, they are nontheless quite close to the Monte Carlo estimates. Furthermore, they are as good as those prices in Table 3 of Duan and Simonato (2001) that are allowed the most computation times.

Criterion (23) is a sufficient condition for MT to be efficient. In this case, the number of tree nodes at date t is the linear $O(t)$; thus the total node count becomes quadratic in maturity. We next use two concrete cases to cross-check the theoretical result. The first setting adopts $\beta_1 = 0.8$, $\beta_2 = 0.1$, and $c = 0$. The tree should not explode because

$$1 = n < \left(\sqrt{\frac{1 - 0.8}{0.1}} - 0 \right)^2 = 2.$$

Indeed, the tree size grows linearly with date as shown in Fig. 17. The total number of nodes is therefore quadratic in maturity, in complete agreement with the theory. Take another setting with $\beta_1 = 0.8$, $\beta_2 = 0.1$, and $c = 0.9$. Criterion (23) is now violated because

$$1 = n > \left(\sqrt{\frac{1 - 0.8}{0.1}} - 0.9 \right)^2 = 0.264416.$$

The tree also turns out to grow exponentially as shown in Fig. 18. Unlike the RTCT tree, however, an exploding MT tree remains useful because it will not be cut short.

Criterion (23) is a sufficient, but not a necessary, condition for the MT tree to escape explosion. Hence its violation does not necessarily imply exponential running time. Take the parameters $\beta_1 = 0.8$, $\beta_2 = 0.1$, and $c = 0.5$ in Table 6 for example. The criterion for nonexplosion is violated for all $n \geq 1$. But in fact, the case of $n = 1$ does not result in a tree of exponential size even though the total node count is more than quadratic. Explosion sets in for $n \geq 2$.

11 Conclusions

GARCH option pricing is difficult because of the GARCH model's bivariate and path-dependent nature. We prove that the Ritchken-Trevor-Cakici-Topyan (RTCT) GARCH option pricing algorithm is inefficient as the RTCT tree explodes for n exceeding a threshold. Worse, a global upper bound on the volatility renders the tree shallow when explosion occurs. Cakici and Topyan (2000) claimed that RTCT is

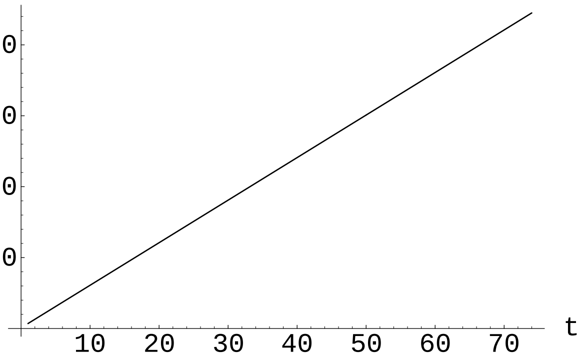

Figure 17: **Linear growth of the MT tree.** The parameters are from Table 6 except for $n = 1$, $c = 0$, $N = 74$, and $K = 20$. Because this set of parameters satisfy criterion (23), the tree should not explode. Indeed, the number of nodes at date t grows linearly with t; the total number of nodes hence grows quadratically in maturity N.

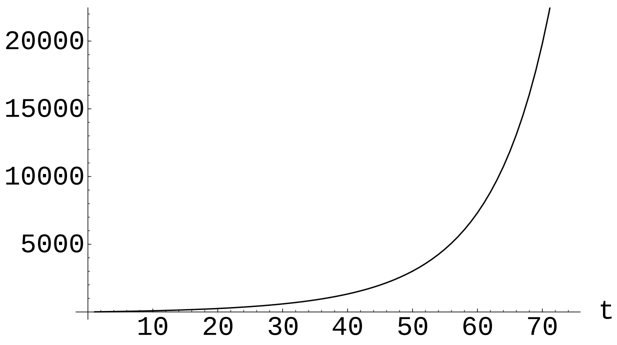

Figure 18: **Exponential growth of the MT tree.** The parameters are identical to those in Fig. 17 except for $c = 0.9$. This set of parameters violate criterion (23). The number of nodes grows exponentially with date t.

empirically accurate with $n = 1$ for vanilla options. Unfortunately, numerical data demonstrate that both inaccuracy and even explosion can result with such under-refined trees. These theoretical results literally carry over to the BDT-GARCH interest rate model of Bali (1999).

The paper modifies RTCT to obtain a simple tree, the mean-tracking (MT) tree. The MT tree is both accurate and provably efficient when n does not exceed a simple threshold. Surprisingly, the tree-size growth is only quadratic in maturity if n does not exceed the threshold. This is the first tree-based GARCH option pricing algorithm that provably does not explode if certain conditions are met. The complexity in fact matches that of the influential and popular CRR binomial tree. Both the threshold and the quadratic size are optimal. Finally, the MT tree does not suffer from the short-maturity problem of the RTCT tree. We conclude that MT is a provably efficient algorithm for derivatives pricing under the GARCH option pricing model. All our theoretical results are proved and backed up by extensive numerical experiments.

Appendix A

Updating rule (15) says $h_t^2 \geq \beta_0 + \beta_1 h_{t-1}^2$. By applying the inequality repeatedly,

$$\begin{aligned} h_t^2 &\geq \beta_0 + \beta_1 h_{t-1}^2 \\ &\geq \cdots \geq \beta_0(1+\beta_1+\beta_1^2+\cdots+\beta_1^{t-1}) + \beta_1^t h_0^2 \\ &= \frac{\beta_0}{1-\beta_1} + \beta_1^t \left(h_0^2 - \frac{\beta_0}{1-\beta_1} \right). \end{aligned} \tag{24}$$

Suppose $h_0^2 \geq \beta_0/(1-\beta_1)$. Then inequality (24) implies $h_t^2 \geq \beta_0/(1-\beta_1)$ for $t \geq 0$. On the other hand, suppose $h_0^2 < \beta_0/(1-\beta_1)$. Let $h_0^2 = \beta_0/(1-\beta_1) - \epsilon$ for some $\epsilon > 0$. By inequality (24) again,

$$h_t^2 \geq \frac{\beta_0}{1-\beta_1} - \beta_1^t \epsilon \geq \frac{\beta_0}{1-\beta_1} - \epsilon = h_0^2$$

because $\beta_1 < 1$. So $h_t^2 \geq h_0^2$ for $t \geq 0$. Combine the two cases to yield

$$h_t^2 \geq \min(h_0^2, \beta_0/(1-\beta_1)), \quad t \geq 0.$$

Appendix B

To confirm that interval (19) contains an integer, it suffices to establish

$$\frac{\sqrt{nh_t^2 + (a\gamma_n - \mu)^2}}{n\gamma_n} \leq \frac{nh_t^2 + (a\gamma_n - \mu)^2}{n\gamma_n |\, a\gamma_n - \mu\,|} - 1. \tag{25}$$

First,

$$\begin{aligned} \frac{nh_t^2 + (a\gamma_n - \mu)^2}{n\gamma_n |\, a\gamma_n - \mu\,|} - 1 &= \frac{h_t^2}{\gamma_n |\, a\gamma_n - \mu\,|} + \frac{|\, a\gamma_n - \mu\,|}{n\gamma_n} - 1 \\ &\geq \frac{h_t^2}{\gamma_n} \frac{1}{(\gamma_n/2)} + \frac{|\, a\gamma_n - \mu\,|}{n\gamma_n} - 1 \quad \text{by inequality (13)} \\ &\geq \left(\frac{2h_t^2}{H_{\min}^2} - 1 \right) + \frac{|\, a\gamma_n - \mu\,|}{n\gamma_n} \quad \text{by inequality (20)} \\ &> 0. \end{aligned}$$

Therefore, it suffices to show that

$$\left[\frac{nh_t^2 + (a\gamma_n - \mu)^2}{n\gamma_n |\, a\gamma_n - \mu\,|} - 1 \right]^2 - \left[\frac{\sqrt{nh_t^2 + (a\gamma_n - \mu)^2}}{n\gamma_n} \right]^2 > 0.$$

Indeed,

$$\begin{aligned}
& \left[\frac{nh_t^2 + (a\gamma_n - \mu)^2}{n\gamma_n |\, a\gamma_n - \mu\,|} - 1 \right]^2 - \left[\frac{\sqrt{nh_t^2 + (a\gamma_n - \mu)^2}}{n\gamma_n} \right]^2 \\
= \;& \frac{1}{n^2\gamma_n^2} \left[\left(\frac{nh_t^2}{|\, a\gamma_n - \mu\,|} + |\, a\gamma_n - \mu\,| - n\gamma_n \right)^2 - nh_t^2 - (a\gamma_n - \mu)^2 \right] \\
= \;& \frac{1}{n^2\gamma_n^2} \left[\frac{n^2h_t^4 - 2n^2h_t^2\gamma_n |\, a\gamma_n - \mu\,|}{(a\gamma_n - \mu)^2} + n^2\gamma_n^2 - 2n\gamma_n |\, a\gamma_n - \mu\,| + nh_t^2 \right] \\
\geq \;& \frac{1}{n^2\gamma_n^2} \left[\frac{n^2h_t^2}{(a\gamma_n - \mu)^2} (h_t^2 - \gamma_n^2) + (n^2 - n)\,\gamma_n^2 + nh_t^2 \right] \quad \text{by inequality (13)} \\
> \;& 0,
\end{aligned}$$

as desired.

Appendix C

We shall prove that the size of the MT tree is only quadratic in maturity N if criterion (23) holds. From Eq. (21),

$$\begin{aligned}
\eta \;&\leq\; \left\lceil \frac{\sqrt{nh_t^2 + (\gamma_n^2/4)}}{n\gamma_n} \right\rceil \quad \text{by inequality (13)} \\
&=\; \left\lceil \frac{2\sqrt{nh_t^2 + [\, H_{\min}^2/(16n)\,]}}{\sqrt{n}\, H_{\min}} \right\rceil \quad \text{by equality (22)} \\
&=\; \left\lceil \sqrt{\frac{4h_t^2}{H_{\min}^2} + \frac{1}{4n^2}} \right\rceil \\
&\leq\; \left\lceil \frac{2h_t}{H_{\min}} \right\rceil + 1 \\
&\leq\; \frac{2h_t}{H_{\min}} + 2. \qquad (26)
\end{aligned}$$

Define $\mathcal{C} \equiv H_{\min}[\, \sqrt{n} + 1/(4\sqrt{n})\,]$ and $\mathcal{D} \equiv \sqrt{\beta_1 + \beta_2(\sqrt{n} + c)^2}$. According to updating rule (15),

$$\begin{aligned}
h_{t+1}^2 &\leq \max_{-n\leq\ell\leq n}\left\{\beta_0+\beta_1 h_t^2+\beta_2 h_t^2\left[\frac{\ell\eta\gamma_n+a\gamma_n-(r-h_t^2/2)}{h_t}-c\right]^2\right\}\\
&= \max_{-n\leq\ell\leq n}\left\{\beta_0+\beta_1 h_t^2+\beta_2\left[\,\ell\eta\gamma_n+a\gamma_n-(r-h_t^2/2)-ch_t\,\right]^2\right\}\\
&\leq \max_{-n\leq\ell\leq n}\left\{\beta_0+\beta_1 h_t^2+\beta_2\left[\,|\,\ell\eta\gamma_n\,|+|\,a\gamma_n-(r-h_t^2/2)\,|+|-ch_t\,|\,\right]^2\right\}\\
&\leq \beta_0+\beta_1 h_t^2+\beta_2\left(n\eta\gamma_n+\frac{\gamma_n}{2}+ch_t\right)^2 \quad \text{by inequality (13)}\\
&\leq \beta_0+\beta_1 h_t^2+\beta_2\left[\left(\frac{2nh_t}{H_{\min}}+2n+\frac{1}{2}\right)\gamma_n+ch_t\right]^2 \quad \text{by inequality (26)}\\
&= \beta_0+\beta_1 h_t^2+\beta_2\left[\,(\sqrt{n}+c)\,h_t+\mathcal{C}\,\right]^2 \quad \text{by equality (22)}\\
&= \left[\,\beta_1+\beta_2(\sqrt{n}+c)^2\,\right]h_t^2+2\beta_2\mathcal{C}(\sqrt{n}+c)\,h_t+\beta_0+\beta_2\mathcal{C}^2\\
&= (\mathcal{D}h_t+\mathcal{P})^2+\mathcal{Q},
\end{aligned}$$

where $\mathcal{P}$ and $\mathcal{Q}$ are positive numbers independent of h_t and t. Hence there exists a number $\mathcal{R}>0$ independent of h_t and t such that

$$h_{t+1}^2\leq(\mathcal{D}h_t+\mathcal{P})^2+\mathcal{Q}\leq(\mathcal{D}h_t+\mathcal{R})^2.$$

(For example, one can pick $\mathcal{R}=\mathcal{P}+\sqrt{\mathcal{Q}}$.) This leads to

$$h_{t+1}\leq\mathcal{D}h_t+\mathcal{R}. \tag{27}$$

Let H_t stand for the largest of all the volatilities at date t. Note that $H_0=h_0$. By inequality (27), state (y_t,h_t^2)'s $2n+1$ successor volatilities are all dominated by $\mathcal{D}h_t+\mathcal{R}$. Suppose H_{t+1} at date $t+1$ follows volatility $\tilde{h}_t$ at date t (which may be an interpolated volatility). Then $H_{t+1}\leq\mathcal{D}\tilde{h}_t+\mathcal{R}\leq\mathcal{D}H_t+\mathcal{R}$; hence $H_{t+1}\leq\mathcal{D}H_t+\mathcal{R}$. By induction,

$$H_{t+1}\leq\mathcal{R}\sum_{i=0}^{t}\mathcal{D}^i+\mathcal{D}^{t+1}h_0=\frac{\mathcal{R}}{1-\mathcal{D}}+\left[h_0+\frac{\mathcal{R}}{\mathcal{D}-1}\right]\mathcal{D}^{t+1}.$$

Therefore, H_{t+1} at date $t+1$ does not grow exponentially when $\mathcal{D}\leq 1$, i.e., when $\beta_1+\beta_2(\sqrt{n}+c)^2\leq 1$, which establishes criterion (23).

We proceed to derive the quadratic tree size. Assume $\mathcal{D}\leq 1$. It is important to note that

$$H_{t+1}\leq\frac{\mathcal{R}}{1-\mathcal{D}}+h_0\mathcal{D}^{t+1}\leq\frac{\mathcal{R}}{1-\mathcal{D}}+h_0.$$

By bound (14) and inequality (26), the number of nodes at date $t > 0$ is at most

$$\begin{aligned}
\sum_{i=0}^{t-1} \left[2n \left(\frac{2H_i}{H_{\min}} + 2 \right) + 1 \right] &= (4n+1)\,t + \frac{4n}{H_{\min}} \sum_{i=0}^{t-1} H_i \\
&\leq (4n+1)\,t + \frac{4n}{H_{\min}}\, t \left(\frac{\mathcal{R}}{1-\mathcal{D}} + h_0 \right) \\
&= \left[4n + 1 + \frac{4n}{H_{\min}} \left(\frac{\mathcal{R}}{1-\mathcal{D}} + h_0 \right) \right] t.
\end{aligned}$$

Hence the number of nodes at date t is linear in t as depicted in Fig. 17. The total size of an N-day MT tree is therefore at most

$$1 + \sum_{t=1}^{N} \left[4n + 1 + \frac{4n}{H_{\min}} \left(\frac{\mathcal{R}}{1-\mathcal{D}} + h_0 \right) \right] t = O(N^2),$$

a quadratic growth.

References

[1] Bali, Turan G. "An Empirical Comparison of Continuous Time Models of the Short Term Interest Rate." *The Journal of Futures Markets*, 19, No. 7 (1999), 777–797.

[2] Bollerslev, T. (1986) Generalized Autoregressive Conditional Heteroskedasticity. *Journal of Econometrics*, 31, pp. 307–327.

[3] Bollerslev, T., Chou, R.Y., and Kroner, K. (1992) ARCH Modeling in Finance: a Review of the Theory and Empirical Evidence. *Journal of Econometrics*, 52, pp. 5–59.

[4] Cakici, N., and Topyan, K. (2000) The GARCH Option Pricing Model: a Lattice Approach. *Journal of Computational Finance*, 3(4), pp. 71–85.

[5] Chien, H.-H. (2003) *On the Complexity of the Ritchken-Sankarasubramanian Interest Rate Model.* MBA Thesis. Department of Finance, National Taiwan University, Taiwan.

[6] Cox, J.C., Ingersoll, J.E., and Ross, S.A. (1985) A Theory of the Term Structure of Interest Rates. *Econometrica*, 53(2), pp. 385–407.

[7] Cox, J.C., Ross, S.A., and Rubinstein, M. (1979) Option Pricing: a Simplified Approach. *Journal of Financial Economics*, 7(3), pp. 229–263.

[8] Dai, T.-S. (2004) *Pricing Asian Options with Lattices.* Ph.D. Thesis. Department of Computer Science and Information Engineering, National Taiwan University, Taiwan.

[9] Dai, T.-S., and Lyuu, Y.-D. (2004) An Exact Subexponential-Time Lattice Algorithm for Asian Options. In *Proceedings of ACM-SIAM Symposium on Discrete Algorithms*, pp. 710–717, New Orleans, January 11–13, 2004. Philadelphia: Society for Industrial and Applied Mathematics, 2004.

[10] Dempster, M.A.H., and Richards, D.G. (2000) Pricing American Options Fitting the Smile. *Mathematical Finance*, 10(4), pp. 157–177.

[11] Duan, J.-C. (1995) The GARCH Option Pricing Model. *Mathematical Finance*, 5(1), pp. 13–32.

[12] Duan, J.-C. (1997) Augmented GARCH(p, q) Process and Its Diffusion Limit. *Journal of Econometrics*, 79, pp. 97–127.

[13] Duan, J.-C., Gauthier, G., Sasseville, C., and Simonato, J.-G. (2003) Approximating American Option Prices in the GARCH Framework. *Journal of Futures Markets*, 23(10), 915–928.

[14] Duan, J.-C., and Simonato, J.-G. (2001) American Option Pricing under GARCH by a Markov Chain Approximation. *Journal of Economic Dynamics & Control*, 22, pp. 1689–1718.

[15] Engle, R.F., and Ng, V. (1993) Measuring and Testing of the Impact of News on Volatility. *Journal of Finance*, 48, pp. 1749–1778.

[16] Engle, R.F., and Patton, A.J. (2001) What Good Is a Volatility Model? *Quantitative Finance*, 1(2), pp. 237–245.

[17] Forsyth, P.A., Vetzal, K.R., and Zvan, R. (2002) Convergence of Numerical Methods for Valuing Path-Dependent Options Using Interpolation. *Review of Derivatives Research*, 5, pp. 273–314.

[18] Heston, S.L., and Nandi, S. (2000) A Closed-Form GARCH Option Valuation Model. *Review of Financial Studies*, 13(3), pp. 585–625.

[19] Hull, J.C., and White, A. (1993a) Efficient Procedures for Valuing European and American Path-Dependent Options. *Journal of Derivatives*, 1(1), pp. 21–31.

[20] Hull, J.C., and White, A. (1993b) One-Factor Interest-Rate Models and the Valuation of Interest-Rate Derivative Securities. *Journal of Financial and Quantitative Analysis*, 28(2), pp. 235–254.

[21] Li, A., Ritchken, P., and Sankarasubramanian, L. (1995) Lattice Models for Pricing American Interest Rate Claims. *Journal of Finance*, 50(2), pp. 719–737.

[22] Lyuu, Y.-D. (2002) *Financial Engineering & Computation: Principles, Mathematics, Algorithms.* Cambridge, U.K.: Cambridge University Press.

[23] Nelson, D.B., and Ramaswamy, K. (1990) Simple Binomial Processes as Diffusion Approximations in Financial Models. *Review of Financial Studies*, 3(3), pp. 393–430.

[24] Papadimitriou, C.H. (1995) *Computational Complexity.* Reading, MA: Addison-Wesley.

[25] Ritchken, P., and Sankarasubramanian, L. (1995) Volatility Structures of Forward Rates and the Dynamics of the Term Structure. *Mathematical Finance*, 5(1), pp. 55–72.

[26] Ritchken, P., Sankarasubramanian, L., and Vijh, A.M. (1993) The Valuation of Path Dependent Contracts on the Average. *Management Science*, 39(10), pp. 1202–1213.

[27] Ritchken, P., and Trevor, R. (1999) Pricing Options under Generalized GARCH and Stochastic Volatility Processes. *Journal of Finance*, 54(1), pp. 377–402.

[28] Taylor, S. (1986) *Modelling Financial Time Series.* New York: Wiley.

		Total number of	Total number of
n	Final date (N)	nodes	unreachable nodes
3	182	1,017,327	5,565
4	100	499,205	3,028
5	72	368,523	947
10	34	222,935	42
25	18	286,844	6,925
50	12	305,113	448
100	9	578,710	3,961
150	8	795,309	2,011
200	7	652,808	1,596
250	7	1,747,758	20,291
300	7	2,929,508	11,510
350	6	1,179,157	3,151

Table 1: **Final maturity dates and sizes of exploding RTCT trees.** All RTCT trees in the table explode. The total number of nodes in each tree exceeds the $(N+1)^2$ of the standard trinomial tree. The overwhelming majority of these nodes are reachable in each tree as shown in column four. The final maturity date of the tree shortens quickly as n increases. The parameters are $S_0 = 100$, $r = 0$, $h_0^2 = 0.0001096$, $\gamma = 0.010469$, $\beta_0 = 0.000006575$, $\beta_1 = 0.9$, $\beta_2 = 0.04$, and $c = 0$.

K	2	10	20	50	100	200
Option price	4.2301	4.2365	4.2267	4.2274	4.2265	4.2268
${}^{\infty}L$	4.2714					
${}^{\infty}U$	4.3087					

Table 2: **Case where CT fails.** K denotes the number of volatilities per node. The option is a European call with a strike price of 100 and a maturity of 100 days. ${}^{\infty}L$ and ${}^{\infty}U$ form the 95% confidence interval for the true option price based on Monte Carlo simulation of the continuous-state model (1) with 500,000 paths. The parameters are $S_0 = 100$, $r = 0$, $h_0^2 = 0.0001096$, $\gamma = h_0 = 0.010469$, $\beta_0 = 0.000007$, $\beta_1 = 0.9$, $\beta_2 = 0.04$, $n = 1$, and $c = 0$.

	Maturity of Option (Days)											
	2			5			10			20		
n	CT	$^{\infty}DU$	$^{\infty}DL$	CT	$^{\infty}DU$	$^{\infty}DL$	CT	$^{\infty}DU$	$^{\infty}DL$	CT	$^{\infty}DU$	$^{\infty}DL$
1	0.5888	0.5903	0.5856	0.9093*	0.9023	0.8945	1.3116*	1.3400	1.3291	1.8565*	1.8765	1.8609
2	0.5674	0.5701	0.5652	0.9091	0.9132	0.9054	1.3020	1.3126	1.3016	1.8511	1.8604	1.8447
3	0.5736	0.5746	0.5697	0.9284	0.9319	0.9242	1.3103	1.3190	1.3080	1.8532	1.8604	1.8448
4	0.5742	0.5758	0.5709	0.9214	0.9289	0.9211	1.3081	1.3115	1.3006	1.8492	1.8540	1.8384
5	0.5836	0.5841	0.5793	0.9273	0.9322	0.9244	1.3095	1.3172	1.3062	1.8454	1.8605	1.8449
10	0.5839	0.5843	0.5795	0.9257	0.9298	0.9221	1.3059*	1.3174	1.3064	1.8026*	1.8623	1.8467
25	0.5877	0.5923	0.5875	0.9257	0.9310	0.9233	1.2867*	1.3175	1.3065	stopped		
50	0.5874	0.5915	0.5866	0.9238*	0.9335	0.9258	1.2651*	1.3172	1.3062	stopped		
100	0.5876	0.5886	0.5837	0.9202*	0.9314	0.9237	stopped			stopped		
150	0.5876	0.5899	0.5851	0.9189*	0.9304	0.9226	stopped			stopped		
200	0.5877	0.5897	0.5849	0.9179*	0.9308	0.9231	stopped			stopped		
$^{\infty}L$	0.5870			0.9230			1.3060			1.8460		
$^{\infty}U$	0.5920			0.9310			1.3170			1.8620		

	50			75			100			200		
n	CT	$^{\infty}DU$	$^{\infty}DL$	CT	$^{\infty}DU$	$^{\infty}DL$	CT	$^{\infty}DU$	$^{\infty}DL$	CT	$^{\infty}DU$	$^{\infty}DL$
1	2.9415	2.9626	2.9376	3.6043	3.6331	3.6021	4.1647	4.1964	4.1603	5.8926	5.9190	5.8667
2	2.9345	2.9464	2.9213	3.5976*	3.5947	3.5637	4.1570	4.1739	4.1379	5.8863	5.9096	5.8533
3	2.9193*	2.9536	2.9285	3.5567*	3.6240	3.5928	4.0794*	4.1874	4.1512	stopped		
4	2.8784*	2.9482	2.9231	3.4499*	3.5967	3.5658	3.8945*	4.1789	4.1427	stopped		
5	2.8168*	2.9473	2.9222	stopped			stopped			stopped		
10	stopped			stopped			stopped			stopped		
25	stopped			stopped			stopped			stopped		
$^{\infty}L$	2.9180			3.5730			4.1420			5.8620		
$^{\infty}U$	2.9440			3.6050			4.1790			5.9160		

Table 3: **Cases where CT fails to converge with increasing n.** The option is a European call with a strike price of 100. The CT tree may stop growing before the required maturity when n exceeds a threshold. $^{\infty}DL$ and $^{\infty}DU$ form the 95% confidence interval for the option price based on Monte Carlo simulation of the full RTCT tree (7) with 500,000 paths. Asterics mark option prices that lie outside this interval. $^{\infty}L$ and $^{\infty}U$ form the 95% confidence interval for the true option price based on Monte Carlo simulation of the continuous-state model (1) with 500,000 paths. The parameters are $S_0 = 100$, $r = 0$, $h_0^2 = 0.0001096$, $\gamma = h_0 = 0.010469$, $\beta_0 = 0.000006575$, $\beta_1 = 0.9$, $\beta_2 = 0.04$, $K = 20$, and $c = 0$.

CT

	K					
n	2	10	20	50	100	200
1	4.2301	4.2365	4.2267	4.2274	4.2265	4.2268
2	4.1769	4.2617	4.2677	4.2697	4.2705	4.2704
3	3.8894	4.1326	4.1899	4.2361	4.2543	4.2636
4	3.7162	3.9266	4.0006	4.0877	4.1433	4.1883
${}^{\infty}L$	4.2714					
${}^{\infty}U$	4.3087					

MT

	K					
n	2	10	20	50	100	200
1	4.2741	4.2799	4.2804	4.2805	4.2806	4.2806
2	4.1941	4.2714	4.2746	4.2753	4.2754	4.2754
3	3.9900	4.2619	4.2700	4.2731	4.2737	4.2744
4	3.7989	4.2518	4.2658	4.2714	4.2725	4.2742
${}^{\infty}L$	4.2714					
${}^{\infty}U$	4.3087					

Table 4: **Case where CT fails but MT succeeds.** The option is a European call with a strike price of 100 and a maturity of $N = 100$ days. ${}^{\infty}L$ and ${}^{\infty}U$ form the 95% confidence interval for the true option price based on Monte Carlo simulation of the continuous-state model (1) with 500,000 paths. Although the CT tree explodes with $n \geq 3$, the cut-off maturity happens to exceed N days. Hence the option prices for $n \geq 3$ can still be found albeit with great difficulties. The parameters are $S_0 = 100$, $r = 0$, $h_0^2 = 0.0001096$, $\gamma = h_0 = 0.010469$, $\beta_0 = 0.000007$, $\beta_1 = 0.9$, $\beta_2 = 0.04$, $n = 1$, and $c = 0$.

	Maturity of Option (Days)											
	2			5			10			20		
n	MT	$^\infty DU$	$^\infty DL$	MT	$^\infty DU$	$^\infty DL$	MT	$^\infty DU$	$^\infty DL$	MT	$^\infty DU$	$^\infty DL$
1	0.5626	0.5648	0.5599	0.9278	0.9319	0.9242	1.3126	1.3190	1.3080	1.8608	1.8763	1.8607
2	0.5799	0.5841	0.5792	0.9246	0.9264	0.9187	1.3107	1.3190	1.3080	1.8565	1.8675	1.8519
3	0.5833	0.5854	0.5806	0.9253	0.9308	0.9231	1.3104	1.3149	1.3039	1.8553	1.8674	1.8517
4	0.5845	0.5874	0.5825	0.9256	0.9307	0.9229	1.3103	1.3170	1.3060	1.8547	1.8590	1.8434
5	0.5851	0.5883	0.5835	0.9258	0.9274	0.9197	1.3102	1.3129	1.3020	1.8541	1.8635	1.8479
10	0.5864	0.5868	0.5820	0.9263	0.9279	0.9202	1.3099	1.3132	1.3022	1.8524	1.8646	1.8490
25	0.5872	0.5897	0.5848	0.9265	0.9302	0.9225	1.3093	1.3126	1.3017			
50	0.5874	0.5907	0.5859	0.9266	0.9314	0.9237	1.3090	1.3139	1.3029			
100	0.5876	0.5880	0.5832	0.9265	0.9301	0.9223						
150	0.5876	0.5891	0.5842	0.9265	0.9289	0.9212						
200	0.5876	0.5888	0.5840	0.9265	0.9292	0.9224						
$^\infty L$	0.5870			0.9230			1.3060			1.8460		
$^\infty U$	0.5920			0.9310			1.3170			1.8620		

	50			75			100			200		
n	MT	$^\infty DU$	$^\infty DL$	MT	$^\infty DU$	$^\infty DL$	MT	$^\infty DU$	$^\infty DL$	MT	$^\infty DU$	$^\infty DL$
1	2.9468	2.9547	2.9296	3.6105	3.6335	3.6025	4.1698	4.1919	4.1558	5.8974	5.9346	5.8822
2	2.9410	2.9539	2.9288	3.6045	3.6153	3.5843	4.1640	4.1908	4.1546	5.8926	5.9205	5.8682
3	2.9383	2.9576	2.9324	3.6009	3.6290	3.5980	4.1595	4.1770	4.1409			
4	2.9362	2.9536	2.9285	3.5978	3.6260	3.5949	4.1555	4.1704	4.1344			
5	2.9342	2.9554	2.9302									
10												
25												
$^\infty L$	2.9180			3.5730			4.1420			5.8620		
$^\infty U$	2.9440			3.6050			4.1790			5.9160		

Table 5: **Convergence of MT.** $^\infty DL$ and $^\infty DU$ form the 95% confidence interval for the true option price based on Monte Carlo simulation of the full MT tree (15) with 500,000 paths. None of the option prices lie outside this confidence interval. $^\infty L$ and $^\infty U$ form the 95% confidence interval for the true option price based on Monte Carlo simulation of the continuous-state model (1) with 500,000 paths. All parameters are from Table 3.

	Maturity of Option (Days) = 30											
	X=55				*X=50*				*X=45*			
	K				*K*				*K*			
n	10	20	50	100	10	20	50	100	10	20	50	100
1	4.8278	4.8576	4.8275	4.8275	1.1038	1.1043	1.1045	1.1045	0.0746	0.0745	0.0744	0.0744
2	4.8350	4.8342	4.8340	4.8340	1.0931	1.0950	1.0955	1.0955	0.0767	0.0762	0.0761	0.0760
3	4.8386	4.8366	4.8360	4.8359	1.0864	1.0912	1.0926	1.0927	0.0780	0.0769	0.0765	0.0765
$^{\infty}L$	4.8364				1.0862				0.0764			
$^{\infty}U$	4.8412				1.0898				0.0792			
DS	4.8377				1.0884				0.0715			

	Maturity of Option (Days) = 90											
	X=55				*X=50*				*X=45*			
	K				*K*				*K*			
n	10	20	50	100	10	20	50	100	10	20	50	100
1	4.9513	4.9519	4.9520	4.9520	1.8361	1.8393	1.8398	1.8399	0.4210	0.4215	0.4216	0.4216
$^{\infty}L$	4.9505				1.8162				0.4131			
$^{\infty}U$	4.9587				1.8232				0.4185			
DS	4.9550				1.8197				0.4036			

	Maturity of Option (Days) = 270											
	X=55				*X=50*				*X=45*			
	K				*K*				*K*			
n	10	20	50	100	10	20	50	100	10	20	50	100
1	5.4806	5.4869	5.4880	5.4881	2.8505	2.8565	2.8573	2.8574	1.2016	1.2036	1.2039	1.2039
$^{\infty}L$	5.4708				2.8361				1.1902			
$^{\infty}U$	5.4838				2.8471				1.1988			
DS	5.4899				2.8471				1.1867			

Table 6: **Accuracy of MT with nonzero r and c.** The option is a European put with a strike price of X. $^{\infty}L$ and $^{\infty}U$ form the 95% confidence interval for the true option price based on Monte Carlo data from Duan and Simonato (2001). DS lists the option prices from Duan and Simonato (2001) given the most computational efforts. The table does not compute prices for $n > 1$ when the maturity exceeds 30 days because the tree explodes. All parameters are from Duan and Simonato (2001): $S_0 = 50$, $r = 5\%$ (annual), $h_0^2 = 0.0001096$, $\beta_0 = 0.00001$, $\beta_1 = 0.8$, $\beta_2 = 0.1$, and $c = 0.5$.