Complexity of the Ritchken-Trevor-Cakici-Topyan GARCH Option Pricing Algorithm

Yuh-Dauh Lyuu
Computer Science & Information Engineering
National Taiwan University
No. 1, Sec. 4, Roosevelt Rd.
Taipei, Taiwan

Chi-Ning Wu
Department of Finance
National Taiwan University
No. 1, Sec. 4, Roosevelt Rd.
Taipei, Taiwan

Abstract

The trinomial-tree GARCH option pricing algorithm of Ritchken and Trevor (1999) is claimed to be efficient. That algorithm is subsequently modified by Cakici and Topyan (2000). However, this paper proves that both algorithms explode exponentially when the number of partitions per day, n, exceeds a typically small number determined by the GARCH parameters. Worse, when explosion happens, the tree cannot grow beyond a certain maturity date and the usual tradeoff between accuracy and efficiency ceases to exist. Hence the algorithms must be limited to using small n's, which may have accuracy problems. Numerical data confirm the theoretical results. The problem of designing efficient tree-based GARCH option pricing algorithms therefore remains open.

Keywords: GARCH, trinomial tree, path dependency, option

1 Introduction

Efficient numerical algorithms are paramount to derivatives pricing. Because the exponential function grows so fast, exponential-time algorithms present the difficult choice between accuracy and reasonable running time much earlier than polynomial-time algorithms. Exponential-time algorithms are therefore said to suffer from combinatorial explosion.

In pricing derivatives numerically, the diffusion process of the asset price can be discretized to yield a tree structure. Derivatives can then be priced on the tree. 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 variations, are that it recombines and that an m-period tree contains only $O(m^2)$ nodes. As a consequence, vanilla options can be efficiently priced. Efficient as the tree may be, a pricing algorithm based on it may still explode if the derivative itself is complex. The Asian option fits this characterization because of the vast amount of extra states added by its path-dependent feature. To tackle this problem, approximations are necessary as surveyed in Lyuu (2002).

A qualitatively more serious problem emerges when the explosion arises from the model itself. If the model generates unrecombining trees, pricing can be expensive even for simple derivatives like the vanilla option. 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 trees that do not recombine: An m-period binomial tree now contains $2^{m+1}-1$ nodes. The situation may be rectified by a technique of Nelson and Ramaswamy (1990) to transform the diffusion process into one with a constant volatility. But the methodology, even where applicable, does not guarantee to reduce the tree size to subexponential. The complexity issue is particularly relevant when the diffusion process is bivariate. Two bivariate models are the interest rate model of Ritchken and Sankarasubramanian (1995) and the GARCH option pricing model of Duan (1995), the focus of this paper.

Bollerslev (1986) and Taylor (1986) independently propose the GARCH process popular in modeling the stochastic volatility of asset returns. Duan (1995) later applies the model to options pricing. Because of the massive path dependency of the model, simulation has been the algorithm of choice. The situation changes with the appearance of the tree algorithm of Ritchken and Trevor (1999). Their algorithm is claimed to be efficient; furthermore, it is general enough to work beyond GARCH models.

This paper investigates the performance of the Ritchken-Trevor algorithm and its modified version by Cakici and Topyan (2000). The results are discouraging, both theoretically and numerically. It will be shown that the Ritchken-Trevor-Cakici-Topyan (RTCT) algorithm creates exponential-sized trees. The condition for this combinatorial explosion mirrors that for GARCH to be nonstationary. It is satisfied when the number of partitions per day, n, exceeds a typically small number. The algorithm is hence not efficient unless the tree is restricted to small n's.

Now suppose one is willing to trade efficiency for better accuracy with a large n. Such a tradeoff usually exists for numerical algorithms. But this sensible practice turns out to be impossible: When explosion occurs, the RTCT tree cannot grow beyond a certain maturity. This extremely negative result obliterates the tradeoff between accuracy and efficiency taken for granted in the literature. It also throws into question some of the calculated prices in Ritchken and Trevor (1999). All the theoretical results are backed up by numerical data. The existence of efficient tree-based GARCH option pricing algorithms therefore remains open.

The paper is organized as follows. The GARCH model is presented in Section 2. Section 3 reviews the essence of the RTCT tree. In Section 4, conditions for the tree to explode exponentially are proved. The nonexistence of the tradeoff between efficiency and accuracy is proved in Section 5. Section 6 provides numerical data to back up the theoretical results and throw into question some of the calculated prices in Ritchken and Trevor (1999). Section 7 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 $(t+1)$st day $[t, t+1]$. Here, "one day" is 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):

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

where

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

It is postulated that $\beta_0, \beta_1, \beta_2 \geq 0$ to make the conditional variance h_t^2 positive. 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. Process (1) for conditional variance, due to Engle and Ng (1993), is called the nonlinear asymmetric GARCH.

3 Building the Tree

The RTCT trinomial tree approximates the continuous-state GARCH process with discrete states. Partition a day into n periods. Three successor states will 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. We next pick the jump size and the branching probabilities to match the distribution of y_{t+1}. Let $\gamma = h_0$ and $\gamma_n = \gamma/\sqrt{n}$. (Our results will be seen to be independent of how γ is picked.) The jump size will be some integer multiple η of γ_n. See Fig. 1 for illustration. Note that the middle branch does not change the underlying asset's price. The probabilities for the up, middle, and down branches are

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

$$p_m = 1 - \frac{h_t^2}{\eta^2\gamma^2}, \qquad (3)$$

$$p_d = \frac{h_t^2}{2\eta^2\gamma^2} - \frac{r - (h_t^2/2)}{2\eta\gamma\sqrt{n}}. \qquad (4)$$

The intermediate nodes between dates are dispensed with to create a $(2n+1)$-nomial tree as in Fig. 2 to reduce the node count. The resulting model is multinomial with $2n+1$ branches from any state (y_t, h_t^2). From Eqs. (2)–(4), valid branching probabilities exist (i.e., $0 \leq p_u, p_m, p_d \leq 1$) if and only if

$$\frac{|\, r - (h_t^2/2)\,|}{2\eta\gamma\sqrt{n}} \leq \frac{h_t^2}{2\eta^2\gamma^2} \leq \min(1 - \frac{|\, r - (h_t^2/2)\,|}{2\eta\gamma\sqrt{n}}, \frac{1}{2}). \tag{5}$$

The updating rule (1) must be modified to account for the adoption of the discrete-state model. For $-n \leq \ell \leq n$, state (y_t, h_t^2) at date t is followed by state $(y_t + \ell\eta\gamma_n, h_{t+1}^2)$ at date $t+1$, where

$$\begin{aligned} h_{t+1}^2 &= \beta_0 + \beta_1 h_t^2 + \beta_2 h_t^2 (\epsilon'_{t+1} - c)^2, \quad (6) \\ \epsilon'_{t+1} &= \frac{\ell\eta\gamma_n - (r - h_t^2/2)}{h_t}. \end{aligned}$$

This transition happens with probability

$$\sum_{j_u, j_m, j_d} \frac{n!}{j_u!\, j_m!\, j_d!}\, p_u^{j_u} p_m^{j_m} p_d^{j_d},$$

where $j_u, j_m, j_d \geq 0$, $n = j_u + j_m + j_d$, and $\ell = j_u - j_d$.

As volatility h_t changes through time, we may have to pick different η's for different states so that p_u, p_m, and p_d lie between 0 and 1. This implies varying jump sizes. As the necessary requirement $p_m \geq 0$ implies $\eta \geq h_t/\gamma$, we go through

$$\eta = \lceil\, h_t/\gamma\, \rceil, \lceil\, h_t/\gamma\, \rceil + 1, \lceil\, h_t/\gamma\, \rceil + 2, \ldots \tag{7}$$

until valid probabilities are obtained or until their nonexistence is confirmed by inequalities (5). Obviously, the magnitude of η grows with h_t. Backward induction starts after the tree has been built.

Figure 3 depicts a 3-day tree with $n = 1$. Nodes A and B pick $\eta = 2$. Observe that different states may pick different η's. The number of possible values of h_t at a node can be exponential as each path leading to the node carries a different h_t. To handle this complexity, only the maximum and minimum h_t are recorded at each node. For example, the maximum and minimum h_t at nodes C and D pick different jump sizes. Ritchken and Trevor (1999) add extra volatilities between the maximum and minimum h_t^2. Because these interpolated volatilities can only increase the range of future volatilities, our analysis will stand without considering them. Hollow nodes in Fig. 3 are not occupied because they are unreachable. As will be shown later, their count is minuscule.

4 Sufficient Conditions for Explosion

One typically increases n for better accuracy. Unfortunately, the maximum value of h_t grows exponentially in t if n is large enough. When this happens, the tree explodes because it must pick an η that expands exponentially by virtue of Eq. (7). Hence the RTCT tree must be restricted to small n's to have any hope of being efficient. We next provide the argument for the claimed exponential growth of h_t.

Assume $r = 0$ and $c = 0$ first. The updating rule (6) is now

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

where $\ell = 0, \pm 1, \pm 2, \ldots, \pm n$. To make h_{t+1}^2 as large as possible, set $\ell = n$. The updating rule becomes

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

By induction,

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

The above expression grows exponentially if $\beta_1 + \beta_2 n > 1$. 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 maximum value of h_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.

5 The Shallowness of an Exploding Tree

Can a large n be chosen to improve accuracy if we are willing to accept long running times? Unfortunately, the RTCT tree does not admit such a tradeoff. The reason is that there is a ceiling on volatility h_t for valid branching probabilities to exist. With the maximum value of h_t growing exponentially, this ceiling will quickly be reached at some nodes and the tree can grow no further. The choice of n is thus capped even if infinite resources are available. We next derive the said upper bound.

Inequalities (5) imply

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

Hence

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

which can be simplified to yield

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

Finally, the above quadratic inequality (in h_t^2) is equivalent to

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

We conclude that

$$h_t^2 \leq 2(r+n) + 2\sqrt{2rn+n^2} \tag{8}$$

is necessary for the existence of valid branching probabilities. This condition does not depend on the choice of γ because the identity $\gamma = h_0$ did not enter the analysis.

The impossibility result may sound puzzling at first. Under the Black-Scholes model, valid branching probabilities always exist if n is large enough. Why, one may ask, can't the same property hold here? The answer lies in volatility. 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. (2)–(4) for probabilities, and increasing n will eventually have inequality (8) satisfied for all states. In contrast, the volatility h_t under GARCH fluctuates. So each state (y_t, h_t^2) faces different Eqs. (2)–(4) in solving for probabilities. Increasing n makes inequality (8) harder to satisfy for those states with a large h_t^2, whose existence under GARCH has been confirmed earlier.

6 Numerical Examples

The following parameters from Ritchken and Trevor (1999) and Cakici and Topyan (2000) will be assumed throughout the section: $S_0 = 100$, $y_0 = \ln S_0 = 4.60517$, $r = 0$, $h_0^2 = 0.0001096$, $\gamma = 0.010469$, $\beta_0 = 0.000006575$, $\beta_1 = 0.9$, $\beta_2 = 0.04$, and $c = 0$. As $r = c = 0$, combinatorial explosion occurs when

$$n > \frac{1-\beta_1}{\beta_2} = \frac{1-0.9}{0.04} = 2.5.$$

Figure 4 picks $n = 3, 4, 5$ to demonstrate the exponential growth of the RTCT tree. The rate of growth clearly increases with n. For comparison, the standard trinomial tree contains only $2t+1$ nodes at date t.

The number of nodes is a critical indicator because the running time is proportional to it. We mentioned earlier that there may be nodes which are not reachable (recall Fig. 3). In principle, if such nodes are numerous, the algorithm can potentially run more efficiently by skipping them. Figure 5 shows that the proportion of unreachable nodes among all the nodes is small for $n = 3, 4, 5$. We will see shortly that the same conclusion also holds for larger n's. As the overwhelming majority of nodes between the top and bottom nodes are reachable, no substantive benefits can be obtained by clever programming techniques to skip unreachable nodes.

Now suppose we pick $n = 100$ to seek very high accuracy at the expense of efficiency. The theory predicts a high risk of having the RTCT tree cut short. Indeed, with $r = 0$, inequality (8) imposes the upper bound $h_t^2 \leq 4n = 400$.

This means that a node with $h_t > 20$ cannot have valid branching probabilities and thus cannot grow further. As this ceiling is breached at date 9 because of the exponential growth of the maximum value of h_t, the tree stops growing then if not earlier. See Table 1 for the final dates under various n's beyond the threshold of explosion. Observe that the tree's longest maturity decreases rapidly as n increases. It is therefore important not to pick too large an n, for only trees of very short maturities will be generated otherwise. Table 1 also tabulates the total numbers of nodes and unreachable ones. Again, the overwhelming majority of the nodes are occupied as mentioned earlier.

Some of the calculated option prices in Ritchken and Trevor (1999) use n as large as 25 and maturity dates as far as 200. These choices contradict our analysis and data. For example, Table 1 says that the RTCT tree with $n = 25$ should stop growing at date 18. These prices must therefore be viewed with caution. Cakici and Topyan (2000) use $n = 1$ throughout; thus explosion is avoided.

7 Conclusions

We proved that for n suitably large, the RTCT tree explodes. The Ritchken-Trevor-Cakici-Topyan GARCH option pricing algorithm is hence inefficient. Worse, a global upper bound on the volatility renders the tree shallow when explosion occurs. Cakici and Topyan (2000) claim that their pricing algorithm is empirically accurate enough at $n = 1$ for vanilla options. But accuracy must remain a concern with under-refined trees. The problem of designing efficient tree-based GARCH option pricing algorithms therefore remains open. Our results literally carry over to the BDT-GARCH interest rate model of Bali (1999).

References

[1] Bali, T.G. (1999) An Empirical Comparison of Continuous Time Models of the Short Term Interest Rate. *Journal of Futures Markets*, 19(7) , pp. 777–797.

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

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

[4] 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.

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

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

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

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

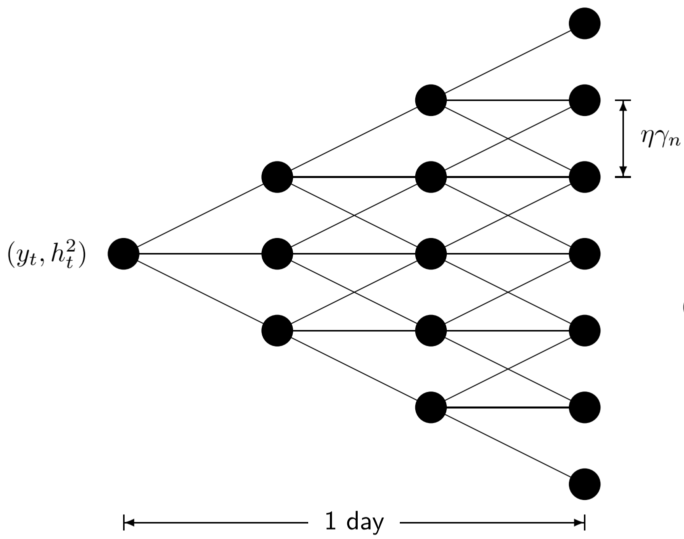
[9] Nelson, D.B., and Ramaswamy, K. (1990) Simple Binomial Processes as Diffusion Approximations in Financial Models. *Review of Financial Studies*, 3(3), pp. 393–430.

[10] 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.

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

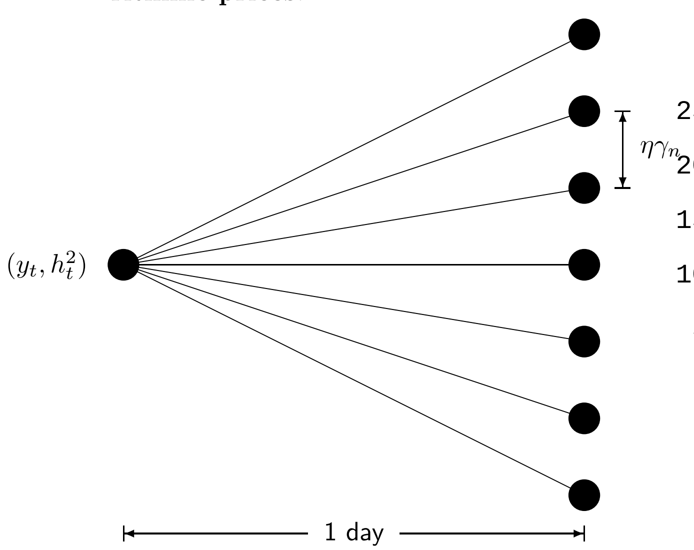
[12] Taylor, S. (1986) *Modelling Financial Time Series*. New York: Wiley.

Figure 1: **Trinomial tree for logarithmic price y_t for the duration of one day.**



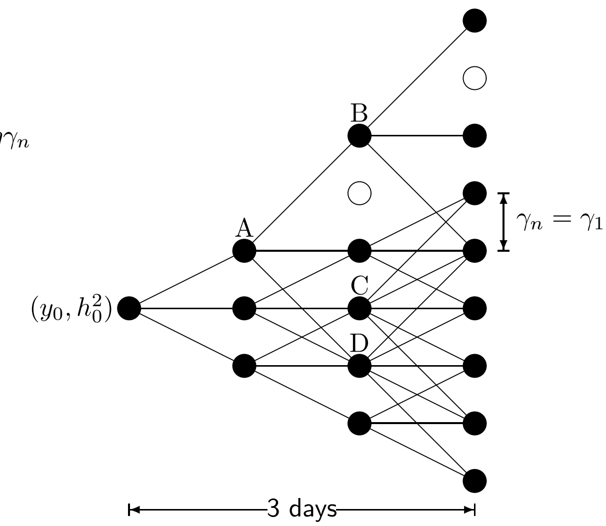
A day is partitioned into $n = 3$ periods, and the jump size is $\eta\gamma_n$. The 7 values on the right should approximate the distribution of y_{t+1} given y_t and h_t^2.

Figure 2: **Multinomial tree for daily logarithmic prices.**



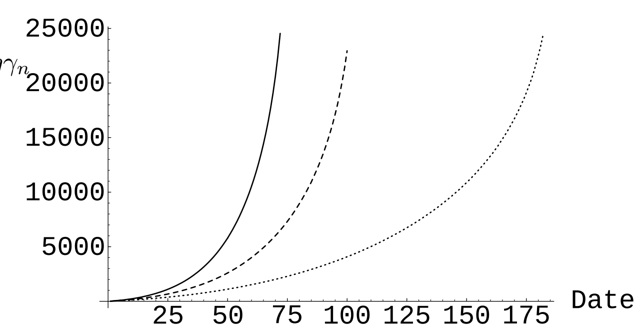
This heptanomial tree is the outcome of the trinomial tree in Fig. 1 after its interior nodes are removed. Recall that $n = 3$.

Figure 3: **Possible geometry of a 3-day RTCT tree.**



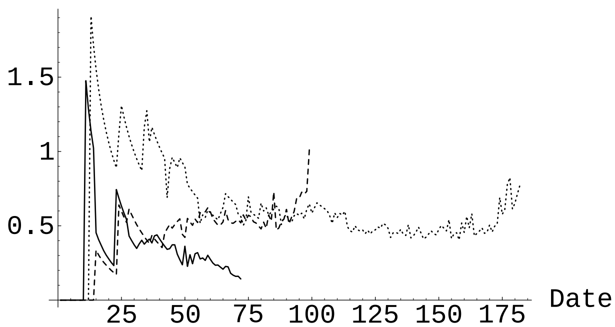
A day is partitioned into $n = 1$ period. Nodes A and B have a jump size of $2\gamma_1$. Nodes C and D have two jump sizes: γ_1 and $2\gamma_1$. All other nodes have a jump size of γ_1. Two nodes between the top and bottom nodes are not reachable. They are shown as hollow nodes.

Figure 4: **Exponential growth of the RTCT tree.**



The parameters are $S_0 = 100$, $y_0 = \ln S_0 = 4.60517$, $r = 0$, $h_0^2 = 0.0001096$, $\gamma = 0.010469$, $\beta_0 = 0.000006575$, $\beta_1 = 0.9$, $\beta_2 = 0.04$, and $c = 0$. The dotted line is based on $n = 3$, the dashed line on $n = 4$, and the solid line on $n = 5$. The standard trinomial tree, in contrast, has only $2t + 1$ nodes at date t.

Figure 5: **The percent of unreachable nodes.**



The parameters are $S_0 = 100$, $y_0 = \ln S_0 = 4.60517$, $r = 0$, $h_0^2 = 0.0001096$, $\gamma = 0.010469$, $\beta_0 = 0.000006575$, $\beta_1 = 0.9$, $\beta_2 = 0.04$, and $c = 0$. The 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.

Table 1: **Final maturity dates and sizes of exploding trees.**

n	Final date (t)	Total number of nodes	Total number of 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

The parameters are $S_0 = 100$, $y_0 = \ln S_0 = 4.60517$, $r = 0$, $h_0^2 = 0.0001096$, $\gamma = 0.010469$, $\beta_0 = 0.000006575$, $\beta_1 = 0.9$, $\beta_2 = 0.04$, and $c = 0$. With $n > 2.5$, all RTCT trees in the table explode. The final maturity date of the tree shortens quickly as n increases. The total number of nodes in each tree far exceeds the $(t+1)^2$ of the standard trinomial tree. The overwhelming majority of nodes are reachable in all trees.