TRINOMIAL TREE OPTION PRICING VIA THRESHOLD-GARCH MODEL

Shu-Ing Liu

Department of Finance, Shih Hsin University, #111, Mu-Cha Road, Sec. 1, Taipei 11604, Taiwan, ROC

ABSTRACT

In this paper, trinomial tree option pricing algorithms for Threshold-GARCH model are presented. The Threshold-GARCH pricing structure provides a more sophisticated description for the changing of conditional variances. To apply the Threshold-GARCH model to evaluate various types of options, convenient and efficient computation algorithms are urgently needed. A simple computational method, called the “Median” algorithm is proposed; moreover, extensions of the interpolating methods proposed by Ritchken & Trevor and Cakici & Topyan are discussed. The numerical results show that the proposed “Median” method is not only accurate, but also offers a significant reduction in computing-time.

Keywords: Conditional variance, Median, Threshold-GARCH model, Trinomial tree.

1. INTRODUCTION

Since the autoregressive conditional heteroskedasticity model (ARCH) was first introduced by Engle [6], it has been successfully and extensively applied to financial econometric analysis. The shape of the impact curve defined by Engle et al. [7] indicates that one of the dominant pricing factors is today’s volatility as a function of yesterday’s returns. Duan [2] pioneered the GARCH approach to option pricing; they proposed, through an equilibrium argument, a discrete-time option pricing model for the GARCH volatility process. In order to alleviate mispricing due to volatility misspecification, flexible volatility models are required. To provide a better description for changing the variance, Härdle et al. [9] introduced a threshold property into the GARCH model for option pricing. Numerical results showed the price of the out-of-the-money options to be enormously dependent on the volatility features. In addition, their empirical results indicated that, for the call options on the German stock index, the DAX, the simulated Threshold-GARCH option prices are rather closer to the market prices than either the Black et al. [1] or the GARCH [2] ones.

Duan, et al. [3] extended the original GARCH model to a simple GJR-GARCH model, which included the possibility of changing the regime of the conditional variances. They also analytically discussed approximate option pricing methods. Liu [8] proposed an alternative representation for a Threshold-GARCH option pricing model. Some analytic properties were derived and the numerical illustrations again indicated that for some sets of S&P500 or S&P100 call options data, the forecasts were rather accurate, in particular, the out-of-the-money case. On the other hand, several computational algorithms for constructing trinomial or multinomial tree approaches under GARCH models have been proposed. First Ritchken et al. [5] developed a lattice algorithm for the pricing of both European and American options under a specific discrete time GARCH process, which was motivated by the Duan GARCH model [2]. Later, Cakici et al. [4] modified the Ritchken et al. [5] algorithm and enhanced its computation efficiency.

The purpose of this paper is to formulate option pricing algorithms via the trinomial trees approach within the discussed Threshold-GARCH model. To ensure both the accuracy and efficiency of the option pricing, a simple and robust method, using only the median of the variance space, instead of applying interpolation, is investigated. Comparisons with the Ritchken et al. [5] and Cakici et al. [4] interpolation methods (of both the accuracy and the efficiency) are performed. The remainder of the paper is organized as follows. The fundamental model and some analytical properties of the proposed model are discussed in Section 2. The construction of Threshold-GARCH lattice is described in Section 3. In Section 4 numerical illustrations that demonstrate the accuracy and efficiency of the discussed option pricing algorithm are provided. Finally, Section 5 concludes this paper and a proof is given in the appendix.
2. THE MODEL

Let the discrete time series data \( \{ X_t \} \) denote the daily asset price data at time \( t \); its one-period rate of log-returns is assumed to be conditionally lognormal distributed. Via a local risk-neutral measure, \( Q \), the GARCH option pricing model, as suggested by Duan [2], can be expressed as

\[
\ln(X_t / X_{t-1}) = r_f - h_t / 2 + \epsilon_t,
\]

where \( \epsilon_t \mid I_{t-1} \sim N(0, h_t) \) and \( h_t = \alpha_0 + \sum_{i=1}^{u} \beta_i (\epsilon_{t-i} - \lambda \sqrt{h_{t-i}})^2 + \sum_{j=1}^{v} \alpha_j h_{t-j} \). Here, \( r_f \) is the daily risk-free rate of return, \( \lambda \) can be interpreted as the daily unit risk premium and \( I_t \) denotes information up to time \( t \). For simplicity, in the following study, attention is focused on the case, \( u = v = 1 \).

Härdele et al. [9] extended the aforementioned model to a Threshold-GARCH model defined by

\[
\ln(X_t / X_{t-1}) = r_f + \eta_t,
\]

where \( \eta_t \mid I_{t-1} \sim N(0, h_t) \) and the conditional variance \( h_t \) is determined by

\[
h_t = \begin{cases} 
\omega + \alpha_1 h_{t-1} + \beta_1 (\eta_{t-1} - \lambda \sqrt{h_{t-1}})^2, & \text{if } \eta_{t-1} \geq \lambda \sqrt{h_{t-1}}, \\
\omega + \alpha_1 h_{t-1} + \beta_2 (\eta_{t-1} - \lambda \sqrt{h_{t-1}})^2, & \text{if } \eta_{t-1} < \lambda \sqrt{h_{t-1}}.
\end{cases}
\]

Empirical results indicate that for the call options on the German stock index (the DAX), the simulated Threshold-GARCH option prices are rather closer to the market prices.

Latter, Duan et al. [3] extended the original GARCH model to a simple GJR-GARCH model defined as

\[
\ln(X_t / X_{t-1}) = r_f - h_t / 2 + \epsilon_t,
\]

\( \epsilon_t \mid I_{t-1} \sim N(0, h_t) \); the conditional variance is generated by the equation below:

\[
h_t = \begin{cases} 
\beta_0 + \beta_1 h_{t-1} + \beta_2 (\epsilon_{t-1} - \lambda \sqrt{h_{t-1}})^2, & \text{if } \epsilon_{t-1} \geq \lambda \sqrt{h_{t-1}}, \\
\beta_0 + \beta_1 h_{t-1} + (\beta_2 + \beta_3) (\epsilon_{t-1} - \lambda \sqrt{h_{t-1}})^2, & \text{if } \epsilon_{t-1} < \lambda \sqrt{h_{t-1}}.
\end{cases}
\]

Actually, the conditional variances for the latter two models are identical. When \( \lambda = 0 \), according to the GARCH model development, \( E[\ln(X_t / X_{t-1}) \mid I_{t-1}] = r_f - h_t / 2 \). Intuitively, the volatility of a bad day, where \( \ln(X_t / X_{t-1}) < r_f - h_t / 2 + \lambda \sqrt{h_t} \), may be different from that of a good day, where \( \ln(X_t / X_{t-1}) \geq r_f - h_t / 2 + \lambda \sqrt{h_t} \).

For convenience, in this paper, an alternative Threshold-GARCH model, which is an alternative representation of the GJR-GARCH model [3] is defined as follows:

\[
y_t = y_{t-1} + r_f - h_t / 2 + \sqrt{h_t} \epsilon_t, \quad (1)
\]

where \( y_t = \ln X_t, \epsilon_t \mid I_{t-1} \sim N(0,1) \), and

\[
h_t = \begin{cases} 
\delta_1 + \alpha_1 h_{t-1} + \beta_1 h_{t-1} (\epsilon_{t-1} - \lambda)^2, & \text{if } \epsilon_{t-1} \geq \lambda, \\
\delta_2 + \alpha_1 h_{t-1} + \beta_2 h_{t-1} (\epsilon_{t-1} - \lambda)^2, & \text{if } \epsilon_{t-1} < \lambda.
\end{cases}
\]

Let \( \theta = (r_f, \lambda, \delta_1, \delta_2, \beta_1, \beta_2, \alpha_1, \alpha_2) \) denote the parameter vector of model (1). The constraints on the parameters are defined as follows: \( 0 < r_f < 1, \ 0 < \lambda < 1, \ 0 < \delta_1, \ 0 < \beta_1, \ 0 < \alpha_1, \) and \( 0 < \alpha_1 + \beta_1 < 1 \). The discussed model splits the data into two regimes: one for the good day period with \( y_t \geq y_{t-1} + r_f - h_t / 2 + \lambda \sqrt{h_t} \); the other for the bad day period with \( y_t < y_{t-1} + r_f - h_t / 2 + \lambda \sqrt{h_t} \). Instead of applying a diffuse process to
describe the risk-free return rate \( r_f \), for simplicity, both \( r_f \) and the risk premium \( \lambda \) are regarded as parameters.

### 3. CONSTRUCTION OF TRINOMIAL TREES

In order to conveniently establish computational algorithm for option pricing using model (1), the Ritchken et al. [5] development is adopted. Model (1) is modified as follows:

\[
y_t = y_{t-1} + r_f - \frac{h_t}{2} + \sqrt{h_t} \varepsilon_t,
\]

where \( \varepsilon_t \mid I_{t-1} \sim N(0,1), \) and the conditional variance \( h_{t,i} \) is defined as

\[
h_{t,i} = \begin{cases} 
    h_{t,i-1} + \delta_1 + \alpha_1 h_{t,i-2} + \beta_1 h_{t,i-2} \left( \varepsilon_{t,i-1} - \lambda \right)^2, & \text{if } \varepsilon_{t,i-1} - \lambda \geq 0, \\
    h_{t,i-1} + \delta_2 + \alpha_2 h_{t,i-2} + \beta_2 h_{t,i-2} \left( \varepsilon_{t,i-1} - \lambda \right)^2, & \text{if } \varepsilon_{t,i-1} - \lambda < 0. 
\end{cases}
\]

That is, in the Threshold-GARCH model we have to determine whether the parameter set \( (\delta_1, \alpha_1, \beta_1) \) or the other set \( (\delta_2, \alpha_2, \beta_2) \), depending on the relative value of \( \varepsilon_t \) and \( \lambda \), is appropriate. It can be shown (see the appendix) that the discounted asset price process \( \{ e^{-rt} X_t \} \) under the discussed model, models (2)-(3), is a martingale process. For convenience, the discussed model is denoted as the Threshold-GARCH model hereafter.

A grid of logarithmic prices is established for the lattice algorithm suggested by Ritchken et al. [5]. The gap between adjacent logarithmic prices is determined by \( \gamma \), a spacing parameter. For simplicity, in this work, \( \gamma \) is defined as \( \gamma = \sqrt{h_t} \), where \( h_t \) is an initial conditional variance. In case of large volatility, another important parameter, called the jump parameter, \( \eta \), is introduced and adjusted accordingly. The general procedure is now described in details.

Assume that the current time is \( t \). The jump parameter, \( \eta_t \) is defined as an integer which depends on the level of volatility, say, \( \eta_t - 1 < \sqrt{h_t}/\gamma \leq \eta_t \). The \( \varepsilon_t \) at time \( t+1 \) is calculated by

\[
\varepsilon_{t+1} = \left( j \eta^{-1} - (r_f - h_t/2) \right) / \sqrt{h_t}, \quad \text{where} \quad j = -1, 0, 1.
\]

Given the discussed threshold property, \( h_{t,i} \) will be either \( h_{t,i+} \) or \( h_{t,i-} \) as determined by eq. (3): at node \((t,i)\), \( h_{t,i+} \) denotes the conditional variance, satisfying \( \varepsilon_{t,i+} - \lambda \geq 0 \), recorded at the left part of this node; similarly, \( h_{t,i-} \) denotes the conditional variance, satisfying \( \varepsilon_{t,i-} - \lambda < 0 \), recorded at the right part. Translating to the next period, the logarithmic price is updated as \( y_{t+1} = y_t + j \eta_t \gamma \) and the trinomial probabilities are given by

\[
\begin{align*}
    P_{t}^{(u)} &= \frac{h_t}{2 \eta_t^2 \gamma^2} + \frac{(r_f - h_t/2)}{2 \eta_t \gamma}, \\
P_{t}^{(m)} &= 1 - \frac{h_t}{\eta_t^2 \gamma^2}, \\
P_{t}^{(d)} &= \frac{h_t}{2 \eta_t^2 \gamma^2} - \frac{(r_f - h_t/2)}{2 \eta_t \gamma},
\end{align*}
\]

again \( h_t \) will be either \( h_{t+} \) or \( h_{t-} \).

By adding detailed subscripts, the jump parameter \( \eta \) can be defined by a suitable integer, satisfying either \( \eta_{t,i+} - 1 < \sqrt{h_{t,i+}}/\gamma \leq \eta_{t,i+} \) or \( \eta_{t,i-} - 1 < \sqrt{h_{t,i-}}/\gamma \leq \eta_{t,i-} \). The current logarithmic stock price is denoted by \( y_{t,i} \), and the corresponding three branching prices are respectively defined as

\[
y_{t+1,a(i)} = y_{t,i} + \eta_{t,i+} \gamma, \quad y_{t+1,b(i)} = y_{t,i} \quad \text{and} \quad y_{t+1,c(i)} = y_{t,i} - \eta_{t,i+} \gamma,
\]

where \( a(i) \) is decided by \( i \) and \( \eta_{t,i+} \), \( b(i) = a(i) - \eta_{t,i+} \) and \( c(i) = b(i) - \eta_{t,i+} \). The corresponding three
branches of \( \epsilon \) values are respectively defined as

\[
e_{t+1, \alpha(i)}^{(u)} = \frac{\eta_i - \gamma - \left( r_f - h_{t,i} \right)}{\sqrt{h_{t,i}}} \quad \text{and} \quad e_{t+1, b(i)}^{(m)} = \frac{-\left( r_f - h_{t,i} \right)}{\sqrt{h_{t,i}}} \quad \text{and} \quad e_{t+1, \epsilon(i)}^{(d)} = \frac{-\eta_i - \gamma - \left( r_f - h_{t,i} \right)}{\sqrt{h_{t,i}}}.
\]

Assuming that the inequality \( e_{t+1, \alpha(i)}^{(u)} - \lambda \geq 0 \) holds, the conditional variance of the upper out-path is defined by

\[
h_{t+1, \alpha(i), u} = \delta_1 + \alpha_1 h_{t,i} + \beta_1 h_{t,j} \left( e_{t+1, \alpha(i)}^{(u)} - \lambda \right)^2;
\]

otherwise

\[
h_{t+1, \alpha(i), u} = \delta_2 + \alpha_2 h_{t,i} + \beta_2 h_{t,j} \left( e_{t+1, \alpha(i)}^{(u)} - \lambda \right)^2.
\]

Similarly, according to the signs of \( e_{t+1, b(i)}^{(m)} - \lambda \) and \( e_{t+1, \epsilon(i)}^{(d)} - \lambda \), the conditional variances of the middle out-path, \( h_{t+1, b(i), m} \), and the lower out-path, \( h_{t+1, \epsilon(i), l} \), are defined respectively.

To capture the path dependence of a GARCH tree, Ritchken et al. [5] approximated the state space of variances at each node by interpolating equally-spaced nodes, ranging between the maximum and minimum, to span the variance space. However, the Threshold-GARCH tree is much more complicated than the GARCH tree, because the variances have two regimes from which to choose. Each node is divided into two parts: the conditional variance \( h_i \), with \( \epsilon_i - \lambda \geq 0 \), which is saved in the left part; and the conditional variance \( h_j \), with \( \epsilon_j - \lambda < 0 \), which is saved in the right part. To ensure that the variance pattern of the Threshold-GARCH tree keeps in a moderate manner, a rather simple but robust method is proposed. Here we suggest that at each node, the median of the elements in the variance space just be recorded. The interpolation methods for the GARCH models, as proposed by Ritchken et al. [5] and Cakici et al. [4], will be applied to the Threshold-GARCH models and discussed in the following section for comparison.

3.1. "Median" Method

In the "Median" method, it is suggested that only one conditional variance in each node be saved, for the forward step of the Threshold-GARCH tree. To avoid creating extra conditional variance that does not originally belong to any path, the "Median" of the elements of the variance space is defined as follows: for a sequence of ordered conditional variances, say, \( h_{(1)} \leq h_{(2)} \leq \cdots \leq h_{(s)} \), the "median" is defined as \( h_{(m)} \), if \( s=2m \); \( h_{(m+1)} \), if \( s=2m+1 \).
Figure 1. Lattice of state variables over three days. (Median Method) The figure shows the first three days of a tree for the Threshold-GARCH model under the "Median" method. Only the median variance is recorded at each part of lattice node.

Thus, there is simply one conditional variance saved inside each node. To help describe the backward construction of the “Median” method, a special Threshold-GARCH tree is illustrated in Figure 1. For simplicity, let the current time be 0, and a European put option, with a maturity date of three days is considered. At each node, a suitable
logarithmic stock price, $y$, is listed. Each node is separated by two regimes, denoted by the left part or the right part, depending on whether "$\varepsilon - \lambda$" is positive or not. Each part contains the corresponding jump parameter, $\eta$, and the option value, $f$.

Moreover, three signs, denoted by, “sgn”, are recorded to distinguish which part of the upper, middle, or lower out-paths it respectively goes in. For instance, $\text{sgn} = "++-"$, means that the upper out-path is saved in the left part, the middle out-path is also saved in the left part, and the lower out-path is saved in the right part. Essentially, there are eight possibilities for the sign combinations. However, according to the ordering of nodes, values of $\varepsilon$ decrease; thus there are only four possibilities, say: $\text{sgn} = "+++", \text{sgn} = "++-", \text{sgn} = "+-+", \text{sgn} = "---"$. The detailed notations of the forward construction procedure are demonstrated as follows. Suppose inequality $\varepsilon_0 - \lambda \geq 0$ is true, then the initial conditional variance is saved at the left part of the node $(0,0)$, and other initial values are given: say, $y_{0,0}$, $h_{0,0,+}$ and $\gamma \left(= \sqrt{h_{0,0,+}} \right)$.

Starting from this node, three branches shoot out with a jump size of $\eta_{0,0,+} = 1$. The corresponding three logarithmic stock prices are given by

$$y_{1,1} = y_{0,0} + \gamma', \quad y_{1,0} = y_{0,0} \quad \text{and} \quad y_{1,-1} = y_{0,0} - \gamma.$$  

Suppose, $\text{sgn} = "++-"$, then the associated $\varepsilon$’s are denoted by

$$c_{1,1,+}^{(u)} = \frac{\gamma - \left( r_f - h_{0,0,+}/2 \right)}{\sqrt{h_{0,0,+}}}, \quad c_{1,0,+}^{(m)} = -\frac{\gamma - \left( r_f - h_{0,0,+}/2 \right)}{\sqrt{h_{0,0,+}}}, \quad \text{and} \quad c_{1,-1,+}^{(d)} = -\frac{\gamma - \left( r_f - h_{0,0,+}/2 \right)}{\sqrt{h_{0,0,+}}}.$$  

In this representation it is intrinsically assumed that $\varepsilon_{1,1,+} - \lambda \geq 0$, $\varepsilon_{1,0,+} - \lambda \geq 0$ and $\varepsilon_{1,-1,+} - \lambda < 0$, and the corresponding conditional variances are given by

$$h_{1,1,+} = \delta_1 + \alpha h_{0,0,+} + \beta h_{0,0,+} \left( c_{1,1,+}^{(u)} - \lambda \right)^2, \quad h_{1,0,+} = \delta_1 + \alpha h_{0,0,+} + \beta h_{0,0,+} \left( c_{1,0,+}^{(m)} - \lambda \right)^2 \quad \text{and} \quad h_{1,-1,+} = \delta_1 + \alpha h_{0,0,+} + \beta h_{0,0,+} \left( c_{1,-1,+}^{(d)} - \lambda \right)^2,$$

respectively. Adopting the same procedure, the remaining nodes are sequentially constructed as shown in Figure 1.

Now, we are ready to proceed with the backward induction procedure. Beginning at the maturity date, the put option price $f_{3,+,2}$ is just the intrinsic value, $f_{3,+,2} = \max\{\exp(y_3) - K, 0\}$. We then move to time $t=2$ and calculate $f_{2,+,3}$, for those non-empty parts of the nodes. First, assume at node (2,3), the $\text{sgn}_{2,3,3} = "++-"$; the trinomial probabilities of the left part is given as follows: by substituting $\eta_{2,3,3} = 1$, into eq. (4), the trinomial probabilities become

$$p_{2,3,+}^{(u)} = \frac{h_{2,3,+} + \left( r_f - h_{2,3,+}/2 \right)}{2\gamma}, \quad p_{2,3,+}^{(m)} = 1 - \frac{h_{2,3,+}}{2\gamma}, \quad \text{and} \quad p_{2,3,+}^{(d)} = \frac{h_{2,3,+} - \left( r_f - h_{2,3,+}/2 \right)}{2\gamma}.$$  

Thus, the corresponding option price $f_{2,3,+}$ is given by

$$f_{2,3,+} = \exp\left( -r_f \left( p_{2,3,+}^{(u)} f_{2,3,3} + p_{2,3,+}^{(m)} f_{2,3,3} + p_{2,3,+}^{(d)} f_{2,3,3} \right) \right),$$

where $r_f$ is the daily continuously compounded risk-free interest rate.

Similarly, according to the tree structure of Figure 1, the other option prices at $t = 2$, can be calculated as follows:

$$\eta_{2,1,+} = 1, \quad \text{sgn}_{2,1,+} = "+++"; \quad f_{2,1,+} = \exp\left( -r_f \left( p_{2,1,+}^{(u)} f_{2,1,3} + p_{2,1,+}^{(m)} f_{2,1,3} + p_{2,1,+}^{(d)} f_{2,1,3} \right) \right)$$

$$\eta_{2,0,+} = 1, \quad \text{sgn}_{2,0,+} = "++-"; \quad f_{2,0,+} = \exp\left( -r_f \left( p_{2,0,+}^{(u)} f_{2,0,3} + p_{2,0,+}^{(m)} f_{2,0,3} + p_{2,0,+}^{(d)} f_{2,0,3} \right) \right)$$

$$\eta_{2,0,-} = 1, \quad \text{sgn}_{2,0,-} = "+-+"; \quad f_{2,0,-} = \exp\left( -r_f \left( p_{2,0,-}^{(u)} f_{2,0,3} + p_{2,0,-}^{(m)} f_{2,0,3} + p_{2,0,-}^{(d)} f_{2,0,3} \right) \right)$$

$$\eta_{2,-1,+} = 1, \quad \text{sgn}_{2,-1,+} = "++-"; \quad f_{2,-1,+} = \exp\left( -r_f \left( p_{2,-1,+}^{(u)} f_{2,-1,3} + p_{2,-1,+}^{(m)} f_{2,-1,3} + p_{2,-1,+}^{(d)} f_{2,-1,3} \right) \right)$$

$$\eta_{2,-2,+} = 1, \quad \text{sgn}_{2,-2,+} = "++-"; \quad f_{2,-2,+} = \exp\left( -r_f \left( p_{2,-2,+}^{(u)} f_{2,-2,3} + p_{2,-2,+}^{(m)} f_{2,-2,3} + p_{2,-2,+}^{(d)} f_{2,-2,3} \right) \right).$$

After finishing the calculations, we move to evaluate values of $f_{1,+,2}$ at $t = 1$, by
Finally, the current option value is given by:
\[
\eta_{1,0,+} = 1, \ sgn = ++: \; f_{1,0,+} = \exp(-r_f) \left( \rho^{(m)}_{0,1,0} f_{2,1,0} + \rho^{(m)}_{1,0,0} f_{1,2,0} + \rho^{(d)}_{0,1,1} f_{0,2,1} \right)
\]

This completes the Threshold-GARCH tree pricing algorithm under the proposed “Median” method.

3.2. Ritchken and Trevor interpolation Method

According to Ritchken et al. [5], there could be as many interpolated points in each node as desired. To help demonstrate the construction of the interpolating method to the discussed model, a special case of the Threshold-GARCH tree is illustrated in Figures 2 and 3. Since the paths of a Threshold-GARCH tree are far more complicated, in the following example, a simple interpolation, accomplished by saving three points at each node, is illustrated.

Again, the current time is assumed to be 0 and at the initial state, data are saved in the left part of the node. For simplicity, a European put option with a maturity date of three days is investigated. Essentially, each notation listed in Figure 1 should be extended to three cases, which respectively correspond to the maximum variance, the mean variance and the minimum variance. Therefore, the number of empty cells should be shortened. However, the list of the logarithmic stock price $y_{t,i}$ still remains the same. For simplicity, the pattern shown in Figure 1 will be applied again and the upper subscript is used to distinguish the three types of conditional variance:
Figure 2. Lattice of variances over three days. (Interpolating Method) The figure shows the first three days of the tree for the Threshold-GARCH model under the Three-point interpolating method. Three variances, maximum, mean, and minimum variances, are recorded at each part of lattice node.
Let \( h_{i,j}^{(1)} \) denote the maximum variance, \( h_{i,j}^{(2)} \) denote the mean variance and \( h_{i,j}^{(3)} \) denote the minimum variance. Accordingly, the associated 3-dimension parameter vector for the left part are defined as: the jump vector, \( \Lambda_{i,j} = (\eta_{i,j}^{(1)}, \eta_{i,j}^{(2)}, \eta_{i,j}^{(3)}) \), the sign vector, \( \Theta_{i,j} = (\text{sgn}_{i,j}^{(1)}, \text{sgn}_{i,j}^{(2)}, \text{sgn}_{i,j}^{(3)}) \), and the option value vector, \( f_{i,j} \).
For the right part, just replace the "+" sign by the "−" sign. In detail, the notation \( \text{sgn}^{(1)}_{i,j,+} = ^+^+^-^- \) means that at node \((t,i)\), for the branching of the maximum variance \( h_{i,j,+}^{(1)} \), the upper out-path is saved in the left part of the \((t+1,a(i))\) node, that is \( \epsilon_{i,j+1,a(i)+}^{(1a)} - \lambda \geq 0 \), where \( a(i) \) is a function of \( i \) and \( h_{i,j,+}^{(1)} \). Similarly, the middle out-path, with \( \epsilon_{i,j+1,b(i)+}^{(1m)} - \lambda \geq 0 \), is saved in the left part of \((t+1,b(i))\) node, where \( b(i) = a(i) - h_{i,j,+}^{(1)} \).

Finally, the lower out-path, with \( \epsilon_{i,j+1,c(i)+}^{(1c)} - \lambda < 0 \), is saved in the right part of the \((t+1,c(i))\) node, where \( c(i) = b(i) - h_{i,j,+}^{(1)} \). Similarly, \( \text{sgn}^{(2)}_{i,j,+} \) and \( \text{sgn}^{(3)}_{i,j,+} \) are defined according to the values of \( \epsilon_{i+1,j,+}^{(2a)} - \lambda \) and \( \epsilon_{i+1,j,+}^{(3a)} - \lambda \), respectively. It is worth of note that at the beginning, the jump size is assumed to be 1 and \( \text{sgn} = ^+^+^-^- \). Therefore, at each \((1,i)\) node, the conditional variances satisfy the following conditions:

\[
\begin{align*}
    h_{1,j,+}^{(1)} &= h_{2,0,+}^{(1)} = h_{1,0,+}^{(1)} = h_{0,0,+}^{(1)} = h_{0,0,+}^{(1)} = h_{0,0,+}^{(1)}, \quad \text{and} \quad h_{1,j,-}^{(1)} = h_{2,0,-}^{(1)} = h_{1,0,-}^{(1)} = h_{0,0,-}^{(1)} = h_{0,0,-}^{(1)}. \tag{5}
\end{align*}
\]

These results imply that

\[
\begin{align*}
    \eta_{1,1,+}^{(1)} &= \eta_{1,1,-}^{(1)} = \eta_{1,0,+}^{(2)} = \eta_{1,0,-}^{(2)} = \eta_{1,-1,+}^{(3)} = \eta_{1,-1,-}^{(3)}. \tag{6}
\end{align*}
\]

In summary, the forward construction is almost the same as the aforementioned "Median" method. The major difference is that there are three values for each parameter saved in each part of the nodes, instead of one value. Adopting the same procedure, the variances and other parameter vectors saved inside each suitable position are constructed sequentially, as shown in Figures 2 and 3. Now, we are ready to proceed with the backward induction procedure ready to proceed. Beginning at the maturity date, the option price \( F_{3,3}^{(1)} \) is the intrinsic value,

\[
F_{3,3}^{(1)} = \max \{ K - \exp(y_{3,3}) \}.
\]

Moving to node \((2,3)\), we are ready to compute \( F_{2,3}^{(1)} \) for each non-empty node.

Take node \((2,3)\) for example: the vector \( F_{2,3}^{(1)} = \{ f_{2,3}^{(1)}, f_{2,3}^{(2)}, f_{2,3}^{(3)} \} \) is obtained as follows: Suppose \( \eta_{1,3}^{(1)} = 1 \); the trinomial probabilities are given by

\[
\begin{align*}
    p_{2,3}^{(1a)} &= \frac{h_{2,3}^{(1a)} + (r_f - h_{2,3}^{(1a)})/2}{2\gamma^2}, \quad p_{2,3}^{(1m)} = 1 - \frac{h_{2,3}^{(1m)}}{2\gamma^2} \quad \text{and} \quad p_{2,3}^{(1c)} = \frac{h_{2,3}^{(1c)} - (r_f - h_{2,3}^{(1c)})/2}{2\gamma^2}.
\end{align*}
\]

Also, assume that \( \text{sgn}_{2,3}^{(1)} = ^+^+^-^- \); then we firstly use the set of option values \( \{ f_{3,4}^{(1)}, f_{3,4}^{(2)}, f_{3,4}^{(3)} \} \) and the corresponding conditional variances \( \{ h_{3,4}^{(1)}, h_{3,4}^{(2)}, h_{3,4}^{(3)} \} \) to interpolate a suitable option value, say \( c_{3,4}^{(1a)} \), with an associated probability \( p_{2,3}^{(1a)} \). Similarly, using \( \{ f_{3,4}^{(1)}, f_{3,3}^{(2)}, f_{3,3}^{(3)} \} \) and the corresponding conditional variances \( \{ h_{3,4}^{(1)}, h_{3,3}^{(2)}, h_{3,3}^{(3)} \} \) to interpolate a suitable option value, say \( c_{3,4}^{(1m)} \), with an associated probability \( p_{2,3}^{(1m)} \); and finally using \( \{ f_{3,2}^{(1)}, f_{3,2}^{(2)}, f_{3,2}^{(3)} \} \) and the corresponding conditional variances \( \{ h_{3,2}^{(1)}, h_{3,2}^{(2)}, h_{3,2}^{(3)} \} \) to interpolate a suitable option value, say \( c_{3,2}^{(1c)} \), with an associated probability \( p_{2,3}^{(1c)} \).

Therefore, the corresponding option price, \( f_{2,3}^{(1)} \), is given by

\[
\begin{align*}
    f_{2,3}^{(1)} = \exp(-r_f) \left( p_{2,3}^{(1a)} c_{3,4}^{(1a)} + p_{2,3}^{(1m)} c_{3,4}^{(1m)} + p_{2,3}^{(1c)} c_{3,4}^{(1c)} \right).
\end{align*}
\]

Suppose that \( \eta_{2,3}^{(2)} = 1 \) and \( \text{sgn}_{2,3}^{(2)} = ^+^+^-^- \), then the option price is

\[
\begin{align*}
    f_{2,3}^{(2)} = \exp(-r_f) \left( p_{2,3}^{(2a)} c_{3,4}^{(2a)} + p_{2,3}^{(2m)} c_{3,4}^{(2m)} + p_{2,3}^{(2c)} c_{3,4}^{(2c)} \right).
\end{align*}
\]

And under assumptions that \( \eta_{2,3}^{(3)} = 1 \) and \( \text{sgn}_{2,3}^{(3)} = ^+^+^-^- \), then

\[
\begin{align*}
    f_{2,3}^{(3)} = \exp(-r_f) \left( p_{2,3}^{(3a)} c_{3,4}^{(3a)} + p_{2,3}^{(3m)} c_{3,4}^{(3m)} + p_{2,3}^{(3c)} c_{3,4}^{(3c)} \right).
\end{align*}
\]

This completes the backward step for the option price vector, \( F_{2,3}^{(1)} = \{ f_{2,3}^{(1)}, f_{2,3}^{(2)}, f_{2,3}^{(3)} \} \).

---

138
According to the aforementioned backward development, $F_{2,i,+}, F_{2,0,+}, F_{2,0,-}, F_{2,-1,-}$ and $F_{2,-2,-}$ are similarly derived. Now, we move to compute the values of $F_{1,i,+}$, $F_{1,0,+}$ and $F_{1,-1,-}$. By the results of eqs. (5)-(6), it can be easily proved that the saved option prices at a specific $(1,i)$ node are the same, that is, $f_{1,1,+}^{(1)} = f_{1,1,+}^{(2)} = f_{1,1,+}^{(3)}$, $f_{1,0,+}^{(1)} = f_{1,0,+}^{(3)}$ and $f_{1,-1,-}^{(1)} = f_{1,-1,-}^{(2)} = f_{1,-1,-}^{(3)}$. Again suppose inequality $e_0 - \lambda \geq 0$ is true, then, as a result, the current option value is given by:

$$f_{0,0,+} = p_{0,0,+}^{(1)} f_{1,1,+}^{(1)} + p_{0,0,+}^{(1)} f_{1,0,+}^{(1)} + p_{0,0,+}^{(1)} f_{1,-1,-}^{(1)}.$$  

This finishes the pricing algorithm for the Threshold-GARCH tree under the Ritchken and Trevor interpolation method.

### 3.3. Cakici and Topyan interpolation Method

As pointed out by Cakici et al. [4], the Ritchken et al. [5] interpolating method is rather time-consuming for GARCH option pricing. To enhance computational efficiency, Cakici et al. [4] modified the Ritchken et al. [5] method by saving only the maximum and minimum conditional variances in the forward construction step, and using interpolated variances in the backward step. Despite its efficiency and convergence property, the Cakici et al. algorithm has an inherent flaw, and it is unfortunately on this drawback that the improvement is based on. This problem occurs in the forward construction adopted by Cakici et al. Take a three-pin interpolating GARCH tree as an example. Suppose that at the node $(t,i)$, the three variances $\left(h_{1,i,+}^{(1)}, h_{2,i,+}^{(2)}, h_{3,i,+}^{(3)}\right)$ and the associated jumps $(\eta_{1,i,+}^{(1)}, \eta_{2,i,+}^{(2)}, \eta_{3,i,+}^{(3)})$ are systematically derived from information for all nodes at time $t - 1$. More specifically, assume that $(\eta_{1,i,+}^{(1)}, \eta_{2,i,+}^{(2)}, \eta_{3,i,+}^{(3)}) = (4, 2, 1)$.

For instance, under the Ritchken et al. [5] algorithm, each of the three variances $\left(h_{1,i,+}^{(1)}, h_{2,i,+}^{(2)}, h_{3,i,+}^{(3)}\right)$ has their out-paths associated with non-empty nodes at $t + 1$ when the backward step is processed. For the Cakici and Topyan (2000) algorithm, however, only the out-paths of $h_{2,i,+}^{(2)}$ and $h_{3,i,+}^{(3)}$ are recorded at some suitable nodes $(t+1,i)$. By assumption, $\eta_{2,i,+}^{(2)} = 2$, $h_{2,i,+}^{(2)}$ may find an empty node during the backward step, for $h_{2,i,+}^{(2)}$ does not construct its own out-paths in the forward construction step. One simple solution to the Cakici et al. GARCH pricing algorithm is to restrict the jump size, say, $\eta \leq 2$. Similar to the Ritchken and Trevor procedure illustrated in section 3.2, it is feasible to develop the Cakici and Topyan interpolating algorithm by slightly modifying the procedure mentioned in section 3.2. A detailed description of the procedure is omitted. However, the threshold property separates each node into two parts, so that the deficiency of the jump size may still remain, even though the maximum jump size is restricted, say, $\eta \leq 2$. Numerical comparisons of the convergence and computing time of all the three Threshold-GARCH option pricing algorithms are given in the next section.

### 4. NUMERICAL ILLUSTRATIONS

A set of parameters obtained from S&P 100 data, dating from August, 2001 to March, 2002, serves as the base case in the following numerical illustration: $S_0 = 600$, $r_f = 4 \times 10^{-4}$, $\lambda = 0.07$, $h_0 = 2 \times 10^{-4}$, $\delta_1 = 1.1 \times 10^{-5}$, $\alpha_1 = 0.73$, $\beta_1 = 0.035$, $\delta_2 = 4.2 \times 10^{-5}$, $\alpha_2 = 0.74$ and $\beta_2 = 0.183$. All parameters in the daily base are estimated by the Bayesian method, and the maturity date $T$ is 10 days later. Furthermore, to maintain a moderate pattern of conditional variances, we restrict the maximum jump size of 2 for $\eta$, by setting $\eta = \max \left(2, \left\lfloor \sqrt{h_i / \gamma} \right\rfloor \right)$, here $\left\lfloor x \right\rfloor$ is the smallest integer greater than $x$.

Based on the base case, the prices of the European and American put options are calculated (Tables 1 and 2) under three strike prices: $K=620$ for in-the-money put, $K=600$ for at-the-money put, and $K=580$ for out-of-the-money put. The notation "RT_3" denotes the Ritchken et al. interpolation method with 3 interpolated points, and "RT_5" denotes that with 5 interpolated points. Notations "C&T_3" and "C&T_5" denote the Cakici et al. interpolation method with 3 and 5 interpolated points, respectively. Finally, "MED" denotes the suggested "Median" method. The convergence of each method is examined by partitioning the maturity date, ten days, into suitable steps, from 10 to 300. The results in Table 1 and Table 2 indicate that the convergence patterns of the R&T and C&T methods are
almost identical, which verifies the viability of extending from GARCH option pricing to Threshold-GARCH option pricing.

Moreover, the relative difference between the "Median" method and the other two pricing methods is quite small. Take the at-the-money European put option for instance: the relative difference between "R&T_3" and "MED" is 1.98% when the partition is 10. As the partition gets finer, the relative difference becomes even tinier, as small as 0.05%. Similarly, for the at-the-money American put option, the relative difference is 2.36% when the partition is 10 and 0.02% when the partition is up to 300. In short, the overall relative difference is no more than 0.05%, when the partition increases to 300. This partially verifies the accuracy of the suggested "Median" method.

<table>
<thead>
<tr>
<th>Table 1. Convergence of Threshold-GARCH pricing algorithms for European put option</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Partition</strong></td>
</tr>
<tr>
<td>50</td>
</tr>
<tr>
<td>100</td>
</tr>
<tr>
<td>200</td>
</tr>
<tr>
<td>300</td>
</tr>
</tbody>
</table>

| **Partition** | R&T_3 | R&T_5 | C&T_3 | C&T_5 | MED |
|---|
| 20 | 6.0911 | 6.0920 | 6.0911 | 6.0920 | 5.9611 |
| 50 | 5.1061 | 5.1061 | 5.1061 | 5.1061 | 5.0811 |
| 100 | 4.8602 | 4.8602 | 4.8602 | 4.8602 | 4.8511 |
| 200 | 4.7490 | 4.7490 | 4.7490 | 4.7490 | 4.7453 |
| 300 | 4.7136 | 4.7136 | 4.7136 | 4.7136 | 4.7112 |

Notes: 1. Parameters are based on the Base Case: \( S_0 = 600, r_f = 4 \times 10^{-4}, \delta_1 = 0.07, h_0 = 2 \times 10^{-4}, \delta_2 = 4.2 \times 10^{-4}, \alpha_1 = 0.73, \beta_1 = 0.035, \beta_2 = 0.183, \delta_3 = 1.1 \times 10^{-5}, \delta_4 = 0.2, \text{ and } T = 10 \).
2. R&T_3 and R&T_5 denote the Ritchken and Trevor interpolation method with 3 and 5 interpolated points, respectively.
3. C&T_3 and C&T_5 denote the Cakici and Topyan interpolation method with 3 and 5 interpolated points, respectively.
4. MED denotes the suggested "Median" method.

In addition, the computational efficiency of the discussed methods based on the base case is demonstrated in Table 3. The computation time required for pricing the at-the-money European put option price is recorded. The results indicate that when the partition is 10, the "R&T_3" takes 0.93 seconds to calculate, the "C&T_3" takes 0.69 seconds, and the "MED" takes only 0.22 seconds. Amazingly, when the partition rises to 300, the "R&T_3" takes 523.74 seconds, the "C&T_3" takes 432.36 seconds, but the "MED" takes merely 78.88 seconds. These numerical results show that with the "Median" method, at least 80% of the computing time can be saved, compared to the other two interpolating methods, when the partition is raised to 300. Therefore, the "Median" method, as shown in Tables 1 and 2, is not only an accurate computing algorithm for option pricing under the Threshold-GARCH framework, but is a very efficient computing algorithm as well.
Table 2. Convergence of Threshold-GARCH pricing algorithms for American put option

<table>
<thead>
<tr>
<th>K=620</th>
<th>Partition</th>
<th>R&amp;T_3</th>
<th>R&amp;T_5</th>
<th>C&amp;T_3</th>
<th>C&amp;T_5</th>
<th>MED</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>20.4739</td>
<td>20.4740</td>
<td>20.4739</td>
<td>20.4740</td>
<td>20.4433</td>
<td></td>
</tr>
<tr>
<td>50</td>
<td>20.1701</td>
<td>20.1701</td>
<td>20.1701</td>
<td>20.1700</td>
<td>20.1636</td>
<td></td>
</tr>
<tr>
<td>100</td>
<td>20.0979</td>
<td>20.0979</td>
<td>20.0979</td>
<td>20.0979</td>
<td>20.0955</td>
<td></td>
</tr>
<tr>
<td>300</td>
<td>20.0588</td>
<td>20.0588</td>
<td>20.0588</td>
<td>20.0588</td>
<td>20.0582</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>K=600</th>
<th>Partition</th>
<th>R&amp;T_3</th>
<th>R&amp;T_5</th>
<th>C&amp;T_3</th>
<th>C&amp;T_5</th>
<th>MED</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>20.4739</td>
<td>20.0979</td>
<td>20.1701</td>
<td>20.0979</td>
<td>20.0955</td>
<td></td>
</tr>
<tr>
<td>50</td>
<td>20.0979</td>
<td>20.0678</td>
<td>20.0979</td>
<td>20.0678</td>
<td>20.0669</td>
<td></td>
</tr>
<tr>
<td>100</td>
<td>20.0588</td>
<td>20.0588</td>
<td>20.0588</td>
<td>20.0588</td>
<td>20.0582</td>
<td></td>
</tr>
<tr>
<td>300</td>
<td>20.0588</td>
<td>20.0588</td>
<td>20.0588</td>
<td>20.0588</td>
<td>20.0582</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>K=580</th>
<th>Partition</th>
<th>R&amp;T_3</th>
<th>R&amp;T_5</th>
<th>C&amp;T_3</th>
<th>C&amp;T_5</th>
<th>MED</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>20.1701</td>
<td>20.1701</td>
<td>20.1701</td>
<td>20.1700</td>
<td>20.1636</td>
<td></td>
</tr>
<tr>
<td>50</td>
<td>20.0979</td>
<td>20.0979</td>
<td>20.0979</td>
<td>20.0979</td>
<td>20.0955</td>
<td></td>
</tr>
<tr>
<td>100</td>
<td>20.0588</td>
<td>20.0588</td>
<td>20.0588</td>
<td>20.0588</td>
<td>20.0582</td>
<td></td>
</tr>
<tr>
<td>300</td>
<td>20.0588</td>
<td>20.0588</td>
<td>20.0588</td>
<td>20.0588</td>
<td>20.0582</td>
<td></td>
</tr>
</tbody>
</table>

Notes: Parameters are based on the Base Case: \( S_0 = 600, r = 4 \times 10^{-4}, \lambda = 0.07, h_0 = 2 \times 10^{-4}, \delta_1 = 1.1 \times 10^{-5}, \alpha_1 = 0.73, \beta_1 = 0.035, \delta_2 = 4.2 \times 10^{-5}, \alpha_2 = 0.74, \beta_2 = 0.183, \) and \( T = 10. \)

5. CONCLUSION

This paper discusses a trinomial tree option pricing algorithm under a Threshold-GARCH model assumption. The Threshold-GARCH structure allows the conditional variances to have two regimes from which to choose. It provides a more sophisticated description for the changing of conditional variances than do the other GARCH structures. Some empirical results show that the simulated Threshold-GARCH option prices are rather closer to the market prices than those obtained from the GARCH models. In this work, trinomial tree option pricing algorithms under a Threshold-GARCH framework are developed: In addition, extensions of the Ritchken et al. and Cakici et al. methods, in particular, a simple method, "Median" method, are outlined.

A numerical illustration demonstrates that the suggested "Median" method, in which the "Median" conditional variance, rather than the methods of saving interpolated variances utilized by Ritchken et al. and Cakici et al. are saved, not only ensures the accuracy of the algorithm, but tremendously enhances the computing efficiency. Some details about controlling the pattern of the variances, so as to ensure the robustness of the pricing algorithm, are mentioned and discussed. Finally, the discussed trinomial tree option pricing algorithm for the Threshold-GARCH model can be extended to a multinomial tree structure. In addition, the Greek letters for risk management, such as the delta hedge ratio, could also be numerically derived under the proposed lattice approach. Furthermore, other more complex exotic European and American options could be approximated by extending the discussed delicate trinomial tree algorithm. This would be a direction of future research.

Table 3. Comparisons of computation time (in seconds)

<table>
<thead>
<tr>
<th>Partition</th>
<th>R&amp;T_3</th>
<th>R&amp;T_5</th>
<th>C&amp;T_3</th>
<th>C&amp;T_5</th>
<th>MED</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>0.931</td>
<td>0.991</td>
<td>0.691</td>
<td>0.791</td>
<td>0.221</td>
</tr>
<tr>
<td>20</td>
<td>2.033</td>
<td>2.345</td>
<td>1.472</td>
<td>2.554</td>
<td>0.560</td>
</tr>
<tr>
<td>50</td>
<td>14.221</td>
<td>23.073</td>
<td>11.557</td>
<td>17.044</td>
<td>2.153</td>
</tr>
<tr>
<td>100</td>
<td>56.091</td>
<td>95.127</td>
<td>47.608</td>
<td>69.350</td>
<td>8.792</td>
</tr>
<tr>
<td>200</td>
<td>227.057</td>
<td>386.816</td>
<td>196.533</td>
<td>287.203</td>
<td>35.951</td>
</tr>
<tr>
<td>300</td>
<td>523.744</td>
<td>874.147</td>
<td>432.362</td>
<td>634.462</td>
<td>78.883</td>
</tr>
</tbody>
</table>

Notes: 1. The program is coded by using Matlab software version 7.0.1.
2. Computation platform: Intel® Pentium® 4 CPU 2.40 GHz, 25

6. ACKNOWLEDGEMENTS
This research was supported by the National Science Council, the Republic of China under contract #NSC94-2415-H-128-003. The author wishes to thank Mr. Yu-Chung Liu for his assistance in carrying out part of the computing work. Parts of this work had been presented at the Second International Conference on Innovative Computing, Information and Control, ICICIC, 2007, Kumamoto, Japan.

7. APPENDIX
Show that \( X_t \) of the discussed Threshold-GARCH option pricing model is a martingale process: Since

\[
X_{t+1} = \exp \left( y_t + r_f - \frac{h_t}{2} + \sqrt{h_t} \varepsilon_t \right) = X_t \exp \left( r_f - \frac{h_t}{2} \right) \cdot \exp \left( \sqrt{h_t} \varepsilon_{t+1} \right),
\]

then

\[
E(X_{t+1} | I_t) = E\left( X_t \exp \left( r_f - \frac{h_t}{2} \right) \cdot \exp \left( \sqrt{h_t} \varepsilon_{t+1} \right) | I_t \right)
\]

\[
= \begin{cases} 
X_t \exp \left( r_f - \frac{h_{t,+}}{2} \right) \cdot E \left( \exp \left( \sqrt{h_{t,+}} \varepsilon_{t+1} \right) | I_t \right), & \text{if } \varepsilon_t \geq \lambda, \\
X_t \exp \left( r_f - \frac{h_{t,-}}{2} \right) \cdot E \left( \exp \left( \sqrt{h_{t,-}} \varepsilon_{t+1} \right) | I_t \right), & \text{if } \varepsilon_t < \lambda,
\end{cases}
\]

\[
= \begin{cases} 
X_t \exp \left( r_f - \frac{h_{t,+}}{2} \right) \cdot \exp \left( 0 + \frac{h_{t,+}}{2} \right), & \text{if } \varepsilon_t \geq \lambda, \\
X_t \exp \left( r_f - \frac{h_{t,-}}{2} \right) \cdot \exp \left( 0 + \frac{h_{t,-}}{2} \right), & \text{if } \varepsilon_t < \lambda,
\end{cases}
\]

\[
= \exp \left( r_f \right) X_t.
\]

This completes the proof.

8. REFERENCES


