A NUMERICAL METHOD FOR PRICING AMERICAN-STYLE
ASIAN OPTIONS UNDER GARCH MODEL

DAN SHAO

ABSTRACT. This article develops a numerical method to price American-style Asian option in the context of the generalized autoregressive conditional heteroscedasticity (GARCH) asset return process. The development is based on dynamic programming coupled with the replacement of the normally distributed variable with a binomial one and the whole procedure is under the locally risk-neutral valuation relationship (LRNVR). We investigate the computational and implementation issues of this method and compare them with those of a candidate procedure which involves piecewise-polynomial approximation of the value function. Complexity analysis and computational results suggest that our method is superior to the candidate one and the generated GARCH option prices are capable of reflecting the changes in the conditional volatility of underlying asset.

1. INTRODUCTION

Following the celebrated work of Black and Scholes [4] and Merton [22], researchers have developed the option valuation models to incorporate stochastic volatility which is an indisputable empirical fact. The time-varying volatility models can be generally classified into continuous-time ones and discrete-time Generalized Autoregressive Conditional Heteroscedasticity (GARCH) ones. The early attempts to model continuous-time stochastic volatility include Cox [9], Merton [23], and Geske [16]. Hull and White [19] proposed an additional process to govern the evolution of volatility, which is known as bivariate diffusion model. However, all of these model face the difficulty of implementing and testing because of the nonobservability of variance.

Since it was first proposed by Bollerslev [5], GARCH process has increasingly gained prominence as a powerful econometric tool. Moreover, as pointed out by Heston and Nandi [18], under a GARCH option model, one can calculate the volatilities directly from the historical data of asset returns, which makes it easier to value an option and estimate the model parameters from the discrete observations. The first attempt to price an option in the GARCH framework is done by Duan [11], in which, however, the risk-neutral valuation was incorrectly applied. Amin and Ng [1] developed their model in which the risk-neutral valuation relationship was not employed. By exploring a generalized version of risk neutralization, referred to as the locally risk-neutral valuation relationship (LRNVR), Duan [12] provided sufficient conditions for LRNVR to hold and derived the asset return process under this risk-neutralized measure. Unfortunately these existing GARCH models have

Key words and phrases. GARCH process, heteroscedasticity, Asian options, American options, dynamic programming, piecewise polynomial.
to be solved by Monte Carlo simulation. Heston and Nandi [18] developed a closed-form solution for European option values and hedge ratios in a GARCH model. Their model allows for multiple lags in the time dynamics of the return variance and also allows for the correlation between the return and its variance. The only difference between their option value under GARCH model and the Black-Scholes model option value is that with heteroscedastic variance the value is a function of current and lagged spot asset price while with homoscedastic variance the value just depends on current asset price.

To solve for American option in a GARCH model, Monte Carlo simulation has been the only numerical method for a very long time. Tilley [34], Barraquand and Martineau [2] and Broadie et al. [8] presented three different simulation methods which are numerically feasible for the simple pricing framework where the numbers of early exercise possibilities are limited. By generalizing the binomial tree to time-varying volatility, Ritchken and Trevor [26] provided a lattice approximation to value American options under GARCH process. Duan and Simonato [13] proposed a Markov chain approximation method for American option pricing. They developed an explicit scheme for the GARCH model and proved its convergence.

But until recently applying GARCH process in the pricing of exotic options, such as Asian options, is not well studied. Because Asian option’s payoff depends on the average price of a primitive asset over a certain time period, it is less sensitive to changes in underlying asset price and costs less than the plain vanilla options, which make it popular in financial market. It can be used to hedge the risk exposure of a firm which plans to sell or buy some resources regularly during some period of time.

In the context of constant volatility, analytical solutions of discretely sampled geometric Asian option pricing models are available (Turnbull and Wakeman [35]). For arithmetic average case, which involves estimating an integral for which no computable analytical solution can be easily found, there are rich literatures about how to approximate the solution, such as Bouaziz et al.[6], Rogers and Shi [27], and Hull and White [20], just name a few.

For the heteroscedastic variance case, the research literature is relatively less. Fouque and Han [14] proposed a way to price arithmetic Asian options under the fast mean-reverting stochastic volatility hypothesis by means of the method in Fouque et al. [15]. Wong and Cheung [36] derived a semi-analytical solution to the geometric Asian options and examined the implied volatilities.

This paper develops an approximation method to price arithmetic Asian options under a very flexible GARCH specification. As in Ben-Ameur et al.[3], to price American-style options is formulated as a Markov decision process here, and the option value function satisfies a dynamic programming (DP) recurrence. We write the option value as a function of current time, current primitive asset price, current average price and asset return’s conditional variance, and solve the DP system recursively with backward induction.

We first formulate a numerical solution approach for our DP equation based on piecewise trilinear interpolation over finite grids, which follows immediately from Ben-Ameur et al.[3]. We prove that because of the conditional variance added as an additional variable the time complexity and calculation amount increase exponentially, which makes the algorithm practically unimplementable. Then we propose our alternative solution which involves in replacing a normally distributed variable
with a discrete random variable which only takes finite values. Based on the established properties of the value function, we provide a convergence proof for the proposed method. How to choose the grid in a 3-dimension space will be discussed. We also test the sensitivity of option value to the parameters of GARCH process, which helps us to calibrate those parameters.

The remainder of this paper is organized as follows. Section 2 describes our GARCH model, Asian option contract, and recurrence structure of our model. The properties of value function will be established in Section 3. In Section 4 we develop the DP formulation and elaborate the approximation procedure. Complexity analysis and convergence proof will also be provided. Numerical experiments will be made in Section 5, where the sensitivity test of option value with respect to model parameters will be covered. The characteristics of implied volatility will also be discussed. Section 6 concludes.

2. The GARCH Model and Dynamic Programming Formulation

Under the classical mathematical setting of Harrison and Pliska [17], our discrete-time market, consisting of one primitive asset and one default-free bond, is defined on the probability space \((\Omega, \mathcal{F}, \mathbb{P})\). Let \(T\) be a positive real number (the terminal time), then we assume \(\mathcal{F}_{[0\leq t\leq T]}\) is the \(\mathbb{P}\)-completion of the filtration generated by a Brownian motion on \((\Omega, \mathcal{F}, \mathbb{P})\) and \(\mathcal{F}_{[0\leq t\leq T]}\) satisfies the usual conditions, which are: \(\mathcal{F}_0\) contains all the null sets of \(\mathbb{P}\) and \(\mathcal{F}_{[0\leq t\leq T]}\) is right continuous.

We also assume that this primitive asset, whose price is denoted by \(S_t\), does not pay any dividend and the continuously compounded return on the default-free bond is \(r\), which is a constant. Two basic assumptions should be laid out. The first one is that the log-spot price of the asset follows a particular GARCH process.

**Assumption 2.1.** The one-period rate of return is assumed to be conditionally lognormally distributed under the probability measure \(\mathbb{P}\). That is

\[
\ln \frac{S_t}{S_{t-1}} = r + \lambda \sigma_t - \frac{1}{2} \sigma_t^2 + \sigma_t \epsilon_t, \tag{2.1}
\]

and

\[
\sigma_t^2 = \omega + \sum_{i=1}^{p} \alpha_i \sigma_{t-i}^2 (|\epsilon_{t-i}| - \theta_i \epsilon_{t-i})^s + \sum_{j=1}^{q} \beta_j \sigma_{t-j}^2, \tag{2.2}
\]

with

\[
\epsilon_t \sim \mathcal{N}(0, 1),
\]

where \(\lambda\) is the constant unit risk premium (per unit of conditional standard deviation), \(\epsilon_t\) is i.i.d., \(\theta_i (-1 < \theta_i < 1)\) reflects the asymmetric responses of volatility to positive and negative shocks, and \(s(s > 0)\) acts as a Box-Cox transformation of the conditional standard deviation \(\sigma_t\). \(\omega, \alpha_i,\) and \(\beta_j\) are parameters of the GARCH specification and all of them must be positive to ensure the conditional variance stays positive. Furthermore, to ensure the unconditional expectation \(E^\mathbb{P}[\sigma_t^2]\) exit, we impose that

\[
\frac{1}{\sqrt{2\pi}} \sum_{i=1}^{p} \alpha_i [(1 + \theta_i)^s + (1 - \theta_i)^s] 2^{-\frac{s+1}{2}} \Gamma\left(\frac{s+1}{2}\right) + \sum_{j=1}^{q} \beta_j < 1.
\]
The above specified GARCH process is known as an asymmetric power ARCH (APARCH) model (see Ding et al. [10]). Note that when \( p = q = 0 \), the return process reduces to the standard homoscedastic lognormal process in Black-Scholes model.

Yet at this point we cannot value any option because we don’t know the risk-neutral distribution of asset price. Duan [12] provided sufficient conditions to apply a locally risk-neutral valuation methodology which is applied in the following theorem.

**Theorem 2.1.** Under the locally risk-neutral probability measure \( Q \), the process for asset price is

\[
\ln \frac{S_t}{S_{t-1}} = r - \frac{1}{2} \sigma_t^2 + \sigma_t \xi_t, \tag{2.3}
\]

and

\[
\sigma_t^2 = \omega + \sum_{i=1}^{p} \alpha_i \sigma_{t-i}^2 (|\xi_{t-i} - \lambda| - \theta_i (\xi_{t-i} - \lambda))^s + \sum_{j=1}^{q} \beta_j \sigma_{t-j}^2, \tag{2.4}
\]

with

\[
\xi_t \sim N(0, 1),
\]

where one should note that \( \xi_t - \lambda = \epsilon_t \). To ensure the unconditional expectation \( \mathbb{E}^Q[\sigma_t^2] \) exit, we impose that

\[
\frac{1}{\sqrt{2\pi}} \sum_{i=1}^{p} \alpha_i [(1 - \theta_i)^s A(s, \lambda) + (1 + \theta_i)^s A(s, -\lambda)] + \sum_{j=1}^{q} \beta_j < 1,
\]

where

\[
A(s, \lambda) = \int_{\lambda}^{+\infty} (x - \lambda)^s \exp(-x^2/2)dx
\]

\[
= 2^{\frac{s+1}{2}} \pi^{\frac{s}{2}} \frac{\pi^2 - 2\lambda^2 + \lambda^4 + 2\lambda^2s + s - 1}{s - 1}
\]

\[
L(-\frac{1}{2}s + \frac{1}{2}, 2 \cdot \frac{1}{2}, 2 \cdot \frac{1}{2}) \frac{1}{\cos(\frac{s\pi}{2})} \Gamma(-\frac{1}{2}s + 2)
\]

\[
- \frac{\lambda^2\pi^2}{2} \frac{1}{s - 1} L(-\frac{1}{2}s + \frac{3}{2}, 2 \cdot \frac{1}{2}, 2 \cdot \frac{1}{2}) \frac{1}{\cos(\frac{s\pi}{2})} \Gamma(-\frac{1}{2}s + 2)
\]

\[
- \frac{\sqrt{2}}{4} \pi^2\lambda^2 + s) L(-\frac{s}{2}, 2 \cdot \frac{1}{2}, 2 \cdot \frac{1}{2}) \frac{1}{\sin(\frac{s\pi}{2})} \Gamma(-\frac{1}{2}s + \frac{3}{2})
\]

\[
+ \frac{\sqrt{2}}{4} \pi^2\lambda^2 L(-\frac{s}{2}, 2 \cdot \frac{1}{2}, 2 \cdot \frac{1}{2}) \frac{1}{\sin(\frac{s\pi}{2})} \Gamma(-\frac{1}{2}s + \frac{3}{2})
\]

and \( L(\cdot, \cdot, \cdot) \) is the Laguerre polynomial.

**Proof.** One can refer to the proof of Theorem 2.2 of Duan [12]. \( \square \)

Then immediately from Theorem 2.1 we have the following corollary

\[
S_T = S_t \exp((T-t)r - \frac{1}{2} \sum_{i=t+1}^{T} \sigma_i^2 + \sum_{i=t+1}^{T} \sigma_i \xi_i), \text{ under measure } Q. \tag{2.5}
\]
This paper focuses on the single lag version of the APARCH specification where \( p = q = 1 \). We use the following simplified volatility equation
\[
\sigma_t^2 = \omega + \alpha \sigma_{t-1}^2 (|\xi_{t-1} - \lambda| - \theta (\xi_{t-1} - \lambda))^\beta + \beta \sigma_{t-1}^2.
\] (2.6)

Now we introduce our second assumption.

**Assumption 2.2.** The value function of a contingent claim with one period to maturity can be calculated by Black-Scholes-Rubinstein formula.

This assumption can also be found in Duan [12] and Heston and Nandi [18]. By appealing to arguments of Rubinstein [28] and Brennan [7], we can have Black-Scholes price with discrete-time trading. Thus with Assumption 2.1 and 2.2 we are ready to derive the values of contingent claims whose prices can be written as functions of underlying asset prices.

Then we consider an American-style Asian option contract similar to that of Ben-Ameur et al.[3]. Let \( T \) be the maturity date, and we equally space the time horizon from 0 to \( T \) into \( n \) time-steps, \( 0 = t_0 < t_1 < t_2 < \cdots < t_n = T \), with \( t_i - t_{i-1} = \Delta t \) for \( i = 1, \ldots, n \). Let \( m^* \) be an integer satisfying \( 1 \leq m^* \leq n \), and the option can be exercised only at dates \( t_m \) when \( m^* \leq t_m \leq t_n \). If the option is exercised at \( t_m \), we define the payoff as \((\overline{S}_{t_m} - K)^+ \), \( m^* \) with \( K \) is the predetermined strike price and \( \overline{S}_{t_m} = (S_{t_1} + S_{t_2} + \cdots + S_{t_m})/m \) is the arithmetic average of the discretely sampled asset prices. Note that when \( m^* = n \), the option is actually European-style.

We denote the value function of Asian option at time \( t_m \) by \( V_{t_m}(S_{t_m}, \sigma_{t_{m+1}}^2, \overline{S}_{t_m}) \), which is a function of asset spot price, average price and conditional variance in the state space \([0, \infty)^3\). Thus we can write the exercise value of the option (when \( t_m \geq t_{m^*} \)) as
\[
V_{t_m}^e(\overline{S}_{t_m}) = (\overline{S}_{t_m} - K)^+,
\] (2.7)

while the holding value as
\[
V_{t_m}^h(S_{t_m}, \sigma_{t_{m+1}}^2, \overline{S}_{t_m}) = \rho E^Q[V_{t_{m+1}}(S_{t_{m+1}}, \sigma_{t_{m+2}}^2, \overline{S}_{t_{m+1}})|\mathcal{F}_{t_m}],
\] (2.8)

where \( \rho = e^{-r\Delta t} \) is the discount factor over period \([t_m, t_{m+1}]\). The holding value is the conditional expected value of option, under measure \( Q \), at time \( t_{m+1} \) discounted to time \( t_m \), which represents typically recursive nature. We can summarize the optimal value function as follows
\[
V_{t_m}(S_{t_m}, \sigma_{t_{m+1}}^2, \overline{S}_{t_m}) = \begin{cases} 
V_{t_m}^h(S_{t_m}, \sigma_{t_{m+1}}^2, \overline{S}_{t_m}) & \text{if } 0 \leq t_m \leq t_{m^*} - 1 \\
\max(V_{t_m}^e(\overline{S}_{t_m}), V_{t_m}^h(S_{t_m}, \sigma_{t_{m+1}}^2, \overline{S}_{t_m})) & \text{if } t_{m^*} \leq t_m \leq t_{n-1} \\
V_{t_m}^e(\overline{S}_{t_m}) & \text{if } t_m = T.
\end{cases}
\] (2.9)

To solve equation (2.9), we should use backward induction. From the known value \( V_T \) at maturity, we can calculate \( V_{t_{n-1}} \) based on (2.9), and then \( V_{t_{n-2}} \), and so on. Although we can express \( V_{t_{n-1}} \) analytically, the closed-forms for \( V_{t_m} \) where \( m \leq n - 2 \) are not available. In next section we will elaborate on the approximation methods for \( V_{t_{m}}(m \leq n - 2) \) and discuss their efficiency.

3. Characterization of the Value Function

3.1. The Value Function \( V_{t_{n-1}} \). From the known value function at maturity \( V_T = (\overline{S}_T - K)^+ \), we now derive the closed-form of \( V_{t_{n-1}} \), the value one period
before maturity. We know that
\[ V_T(S_T, \sigma_T^2, S_T) = \max\left( \frac{S_T + (n - 1)S_{t_n - 1}}{n} - K, 0 \right), \] (3.1)
and then at \( t_{n-1} \), we have
\[ V_{t_{n-1}}^h(S_{t_{n-1}}, \sigma_{t_{n-1}}^2, S_{t_{n-1}}) = \frac{\rho E^Q[\max(V_T(S_T, \sigma_T^2, S_T)|\mathcal{F}_{t_{n-1}})]}{\rho E^Q[(S_T + (n - 1)S_{t_n - 1})/n - K]^+|\mathcal{F}_{t_{n-1}}]} \]
\[ = \frac{\rho E^Q[(S_T - K)^+|\mathcal{F}_{t_{n-1}}]}{\rho E^Q[(S_T - K')^+|\mathcal{F}_{t_{n-1}}]}, \] (3.2)
where \( K' = nK - (n - 1)S_{t_n - 1} \).

For \( K' < 0 \), one immediately have
\[ V_{t_{n-1}}^h(S_{t_{n-1}}, \sigma_{t_{n-1}}^2, S_{t_{n-1}}) = \frac{1}{n}(S_{t_{n-1}} - \rho K'). \] (3.3)

By comparing the holding value and the exercise value \( V_{t_{n-1}}^e(S_{t_{n-1}}) = S_{t_{n-1}} - K > 0 \), one can easily find the optimal strategy.

When \( K' > 0 \), the holding value itself is actually the value of a European call option under Black-Scholes model, with spot price \( S_{t_{n-1}} \), strike price \( K' \), time to maturity \( \Delta t \), volatility \( \sigma_T \), and risk-free rate \( r \). Then with the classic Black-Scholes pricing formula, we have
\[ V_{t_{n-1}}^h(S_{t_{n-1}}, \sigma_{t_{n-1}}^2, S_{t_{n-1}}) = \frac{1}{n}(S_{t_{n-1}} N(d_1) - \rho K' N(d_2)|\mathcal{F}_{t_{n-1}})), \] (3.4)
where
\[ d_1 = \frac{\ln(S_{t_{n-1}}/K') + (r + \sigma^2/2)\Delta t}{\sigma\sqrt{\Delta t}}, \quad d_2 = d_1 - \sigma\sqrt{\Delta t}, \]
and \( N(\cdot|\mathcal{F}_{t_{n-1}}) \) is conditional standard normal distribution function. Then by comparing this holding value with the exercise value \( V_{t_{n-1}}^e(S_{t_{n-1}}) \) (when \( S_{t_{n-1}} - K > 0 \)), one should easily decide whether to exercise or not.

Unfortunately, for \( t_m < t_{n-1} \), no analytical solution is available, so we have to resort to numerical method.

3.2. **General Features of the Value Function.** As Ben-Ameur et al.[3], we now prove the monotonicity and convexity properties of the value function, which will contribute to the convergence analysis of our procedure.

**Proposition 3.1.** At each time step \( t_m \), where \( 1 \leq m < n \), the holding value \( V_{t_m}^h(S_{t_m}, \sigma_{t_m}^2, S_{t_m}) \) is a continuous, strictly positive, strictly increasing, and convex function of both \( S_{t_m} \) and \( S_{t_m} \). It’s also continuous, strictly positive and non-decreasing in \( \sigma_{t_m}^2 \). \( V_{t_m}(S_{t_m}, \sigma_{t_m}^2, S_{t_m}) \) shares the same properties with \( V_{t_m}^h \) of its three variables except that it’s non-decreasing in \( S_{t_m} \). Also the value function \( V_0(S_0) \) has the same properties as \( V_{t_m} \) in \( S_0 \).

**Proof.** The proof of the properties of \( V_{t_m}^h \) and \( V_{t_m} \) in \( S_{t_m} \) and \( S_{t_m} \) is similar to that of Proposition 1 of Ben-Ameur et al.[3], so we omit the details here. We only focus on the properties of value function in \( \sigma_{t_m}^2 \).

By denoting \( S_{t_{m+1}}/S_{t_m} = \tau_{t_{m+1}}, \) we have
\[ \ln \tau_{t_{m+1}} = r\Delta t - \frac{1}{2}\sigma_{t_{m+1}}^2 \Delta t + \sigma_{t_{m+1}} \sqrt{\Delta t} \xi_{t_{m+1}}, \] (3.5)
and note that it is lognormally distributed under $\mathbb{Q}$ as follows
\[
\ln \tau_{t_m+1} | \mathcal{F}_{t_m} \sim N(\tau \Delta t - \frac{1}{2} \sigma_{t_m+1}^2 \Delta t, \sigma_{t_m+1}^2 \Delta t). \tag{3.6}
\]
For $m = n - 1$, the holding value is
\[
V_{t_{n-1}}^h(S_{t_{n-1}}, \sigma_{t_{n-1}}, \mathbb{S}_{t_{n-1}}) = \frac{\rho}{n} \mathbb{E}^\mathbb{Q}[V_{t_{n-1}}^h(S_{t_{n-1}} \tau - K) + | \mathcal{F}_{t_{n-1}}]
\]
\[
= \frac{\rho}{n} \int_{0}^{+\infty} (S_{t_{n-1}} \tau - K)^+ f(\tau | \mathcal{F}_{t_{n-1}}) d\tau,
\]
where $f$ is the conditional density function of $\tau_T$ and is continuous and bounded over $\sigma_T^2$. Then by Lebesgue’s dominated convergence theorem, the integral $V_{t_{n-1}}^h$ is also continuous. To show that $V_{t_{n-1}}^h$ is nondecreasing in $\sigma_T^2$, one should note that equation (3.4) implies that it’s an increasing function of $\sigma_T^2$ while equation (3.3) is independent of $\sigma_T^2$.

The value function
\[
V_{t_{n-1}}(S_{t_{n-1}}, \sigma_{t_{n-1}}, \mathbb{S}_{t_{n-1}}) = \max((\mathbb{S}_{t_{n-1}} - K)^+, V_{t_{n-1}}^h)
\]
is also continuous, strictly positive and nondecreasing in $\sigma_T^2$ because it’s the maximum of two functions which satisfy these properties.

We now use mathematical induction to show that these result hold for $m < n - 1$. First we assume that these properties hold for $m + 1$, where $1 \leq m \leq n - 2$, and our target is to prove that this implies the results should hold for $m$. We know that the holding value at time $t_m$ of equation (2.8) is
\[
V_{t_m}^h = \rho \mathbb{E}^\mathbb{Q}[V_{t_{m+1}}^h(S_{t_m} \tau_{t_{m+1}}, \sigma_{t_{m+1}}^2, (m \mathbb{S}_{t_m} + S_{t_m} \tau_{t_{m+1}})/(m + 1)) | \mathcal{F}_{t_m}]
\]
\[
= \rho \int_{0}^{+\infty} V_{t_{m+1}}(S_{t_m} \tau, \sigma_{t_{m+2}}^2, (m \mathbb{S}_{t_m} + S_{t_m} \tau)/(m + 1)) f(\tau | \mathcal{F}_{t_m}) d\tau,
\]
where $f$ is the conditional density function of $\tau_{t_{m+1}}$ and $\sigma_{t_{m+2}}^2$ is a known continuous, bounded, and increasing function of $\sigma_{t_{m+1}}^2$. Since the integrand is continuous, strictly positive, and bounded, so is $V_{t_m}^h$. Also note that $V_{t_m}^h$ is a positively weighted average of $V_{t_{m+1}}^h$ which is a nondecreasing function of its inputs, and that with the increase of $\sigma_{t_{m+1}}^2$, the integral will allocate higher weights to higher values of $V_{t_{m+1}}^h$ and lower weights to lower values. These facts imply that $V_{t_m}^h$ will not decrease on the increase in $\sigma_{t_{m+1}}^2$. The properties of $V_{t_m}$ can be proved by the similar logic as the case where $m = n - 1$. For $V_0$, we could use the same arguments above to prove its properties as well.

**Lemma 3.1.** For $S_{t_m} > 0$, and $\mathbb{S}_{t_m} > \mathbb{S}_{t_{m+1}} > 0$, we have
\[
V_{t_m}^h(S_{t_m}, \sigma_{t_m}^2, \mathbb{S}_{t_m}) - V_{t_m}^h(S_{t_m}, \sigma_{t_{m+1}}^2, \mathbb{S}_{t_{m+1}}) < \frac{m}{m+1}(\mathbb{S}_{t_m}^2 - \mathbb{S}_{t_{m+1}}^2) \rho,
\]
for $1 \leq m < n$,
and
\[
V_{t_m}(S_{t_m}, \sigma_{t_m}^2, \mathbb{S}_{t_m}) - V_{t_m}(S_{t_m}, \sigma_{t_{m+1}}^2, \mathbb{S}_{t_{m+1}}) < \mathbb{S}_{t_m}^2 - \mathbb{S}_{t_{m+1}}^2,
\]
for $1 \leq m \leq n$. 

Proof. The proof is similar to that of Lemma 2 of Ben-Ameur et al. [3]. \qed

4. Numerical Procedures for DP Equations

Before starting to fit the approximation to the value function, we rewrite the value function as \( V_m(S_m, \sigma^2_{m+1}, S_{m-1}) \), by noting that \( S_{m-1} = \frac{mS_m - S_{m-1}}{m-1} \), which will greatly simplify the integration when the approximation is implemented.

4.1. Trilinear Approximation. The approximation method we first consider is a piecewise polynomial which is actually an extension of Ben-Ameur et al. [3]. While there are a lot of potential polynomial functions available, including a piecewise constant function, a piecewise linear function over cone, and high-dimensional splines, etc., the one we consider here is a function which is linear in all of its variables. This is a trade-off in terms of the amount of calculation and a desirable precision. The simple method such as piecewise constant requires much finer partitions to achieve good precision, whereas complicated methods such as high-dimensional spline will lead to overwhelming calculation and more time.

To apply the linear approximation in our three variables \( S_m, \sigma^2_{m+1}, S_{m-1} \), also called trilinear approximation, we let \( 0 = a_0 < a_1 < a_2 < \cdots < a_p < a_{p+1} = \infty \), \( 0 = c_0 < c_1 < c_2 < \cdots < c_z < c_{z+1} = \infty \), and \( 0 = b_0 < b_1 < b_2 < \cdots < b_q < b_{q+1} = \infty \), which generate our grid points

\[
G = \{(a_i, c_j, b_j) : 0 \leq i \leq p, 0 \leq j \leq q\}.
\]

Here we abuse the notations a little bit: the \( p \) and \( q \) here have nothing to do with the GARCH specification APARCH\((s,p,q)\). These grid points partition our positive state space \([0, \infty)^3\) into \((p+1)(q+1)(z+1)\) cubes

\[
C_{ij} = \{(S_m, \sigma^2_{m+1}, S_{m-1}) : a_i \leq S_m < a_{i+1}, c_j \leq \sigma^2_{m+1} < c_{j+1}, \text{ and } b_j \leq S_{m-1} < b_{j+1}\},
\]

where \( i = 0, \ldots, p \), \( g = 0, \ldots, z \), and \( j = 0, \ldots, q \).

The idea now is to approximate the value function \( V_m \) by a trilinear function of \( (S_m, \sigma^2_{m+1}, S_{m-1}) \) over each cube \( C_{ij} \), being continuous at the boundaries. We propose the following trilinear function

\[
\tilde{V}_{m}(S_m, \sigma^2_{m+1}, S_{m-1}) = \phi_{ij}^m + \gamma_{ij}^m S_m + \delta_{ij}^m \sigma^2_{m+1} + \xi_{ij}^m S_{m-1} + \kappa_{ij}^m S_m S_{m-1} + \epsilon_{ij}^m S_m \sigma^2_{m+1} + \mu_{ij}^m \sigma^2_{m+1} S_{m-1} + \nu_{ij}^m S_m \sigma^2_{m+1} S_{m-1},
\]

for any \( (S_m, \sigma^2_{m+1}, S_{m-1}) \in C_{ij} \). To determine those coefficients of this polynomial we first compute the approximation of \( V_m \) denoted by \( \tilde{V}_{m} \), at each vertex of \( C_{ij} \) via equation (2.7) to (2.9) by the available approximation \( \tilde{V}_{m+1} \) of \( V_{m+1} \). Then we impose that \( \tilde{V}_{m} = \tilde{V}_{m} \) at every vertex, which gives us a system of eight equations for each \( C_{ij} \) with eight unknowns. After solving the linear systems we have the values at all the vertexes, so for those points not at the vertex, we simply use interpolation by the adjacent vertexes.

We now show how to compute the approximation \( \tilde{V}_{m} \) given the available approximation \( \tilde{V}_{m+1} \) of \( V_{m+1} \). Note that there is only one random variable \( \xi_{m+1} \) in
the expectation of equation (2.8), where \( S_{t_{m+1}}/S_t = \tau_{t_{m+1}} \) is a function of \( \xi_{t_{m+1}} \).
Also we have
\[ \sigma_{t_{m+2}}^* = \omega + \alpha \sigma_{t_{m+1}}^* (|\xi_{t_{m+1}} - \lambda| - \theta(\xi_{t_{m+1}} - \lambda))^s + \beta \sigma_{t_{m+1}}^*, \] (4.2)
which contains the random variable \( \xi_{t_{m+1}} \) as well. The fact that our approximation is piecewise linear in its three variables makes the integral very easy to compute explicitly. More specifically, we have
\[
\tilde{V}_{t_m}^h (S_{t_m}, \sigma_{t_{m+1}}^2, \mathbb{S}_{t_m}) = \rho E^Q [\tilde{V}_{t_{m+1}}^h (S_{t_{m+1}}, \sigma_{t_{m+2}}^2, \mathbb{S}_{t_{m+1}}) | \mathcal{F}_{t_m}] 
\]
\[= \rho \sum_{i=0}^p \sum_{g=0}^q \sum_{j=0}^q \left[ (\phi_{igj} + \zeta_{igj} S_{t_m}) E^Q [I_{igj} (S_{t_{m+1}}, \sigma_{t_{m+2}}^2, \mathbb{S}_{t_m}) | \mathcal{F}_{t_m}] 
+ (\delta_{igj} + \kappa_{igj} \mathbb{S}_{t_m}) E^Q [I_{igj} (S_{t_{m+1}}, \sigma_{t_{m+2}}^2, \mathbb{S}_{t_m}, \sigma_{t_{m+2}}^2) | \mathcal{F}_{t_m}] 
+ (\psi_{igj} + \nu_{igj} \mathbb{S}_{t_m}) E^Q [I_{igj} (S_{t_{m+1}}, \sigma_{t_{m+2}}^2, \mathbb{S}_{t_m}, \sigma_{t_{m+2}}^2) | \mathcal{F}_{t_m}]) \right], \] (4.3)
where \( I_{igj}(x, y, z) = I\{ (x, y, z) \in C_{igj} \} \) is an indicator function, and \( \varphi \) is chosen to be an integer \( l \) such that \( \mathbb{S}_{t_m} \in [b_l, b_{l+1}) \). The function \( \tilde{V}_{t_m}^h \) is then to be evaluated at the points of \( G(a_k, c_h, b_l) \) for \( k = 0, \ldots, p \), \( h = 0, \ldots, z \) and \( l = 0, \ldots, q \). Observe that in the integration of equation (4.3) for every pair of \( i \) and \( g \), the indicator function \( I_{igj} (S_{t_{m+1}}, \sigma_{t_{m+2}}^2, \mathbb{S}_{t_m}) = I_{igj} (a_k, c_h, b_l) = 1 \) only when the following two conditions to be satisfied at the same time
\[ a_i \leq a_k \tau_{t_{m+1}} < a_{i+1}, \] (4.4)
\[ c_g \leq \sigma_{t_{m+2}}^2 = [\omega + \alpha \chi_h^{s/2} (|\xi_{t_{m+1}} - \lambda| - \theta(\xi_{t_{m+1}} - \lambda))^s + \beta \chi_h^{s/2} ]^{2/s} < c_{g+1}, \] (4.5)
for the random variable \( \xi_{t_{m+1}} \). We denote the interval for \( \xi_{t_{m+1}} \) to satisfy condition (4.4) and (4.5) to be \( [x_{ig,kh}^u, x_{ig,kh}^d] \). Let \( d_{kl} = ((m - 1)b_l + a_k)/m \) for \( k = 0, \ldots, p \) and \( l = 0, \ldots, q \), then at every vertex of our partitioned space we have
\[
\tilde{V}_{t_m}^h (a_k, c_h, b_l) = \rho \sum_{i=0}^p \sum_{g=0}^q \left[ (\phi_{igj}^m + \zeta_{igj}^m d_{kl}) H_{ig,kh} 
+ (\delta_{igj}^m + \kappa_{igj}^m d_{kl}) P_{ig,kh} 
+ (\psi_{igj}^m + \nu_{igj}^m d_{kl}) Q_{ig,kh} \right], \] (4.6)
where \( \varphi \) is chosen such that \( d_{kl} \in [b_{l+1}, b_{l+2}) \),
\[ H_{ig,kh} = E^Q [I\{ a_i \leq a_k \tau_{t_{m+1}} < a_{i+1}, c_g \leq \sigma_{t_{m+2}}^2 < c_{g+1} \} | \mathcal{F}_{t_m}] \]
\[ = N(x_{ig,kh}^u) - N(x_{ig,kh}^d), \]
\[ P_{ig,kh} = E^Q[I\{a_i \leq a_k \tau_{t_{m+1}} < a_{i+1}, c_g \leq \sigma_{t_{m+2}}^2 < c_{g+1}\} a_k \tau_{t_{m+1}} | \mathcal{F}_{t_m}] \]
\[ = a_k \exp(r \Delta t)(N(x_{ig,kh}^u - \sqrt{c_h \Delta t}) - N(x_{ig,kh}^d - \sqrt{c_h \Delta t})), \]

\[ Q_{ig,kh} = E^Q[I\{a_i \leq a_k \tau_{t_{m+1}} < a_{i+1}, c_g \leq \sigma_{t_{m+2}}^2 < c_{g+1}\} \sigma_{t_{m+2}}^2 | \mathcal{F}_{t_m}] \]
\[ = \int_{x_{ig,kh}^d}^{x_{ig,kh}^u} \frac{1}{\sqrt{2\pi}} \left[ \omega + \alpha c_h^{s/2}(|x - \lambda| - \theta(x - \lambda))^s + \beta c_h^{s/2}2/s \exp(-\frac{x^2}{2}) \right] dx, \]

\[ R_{ig,kh} = E^Q[I\{a_i \leq a_k \tau_{t_{m+1}} < a_{i+1}, c_g \leq \sigma_{t_{m+2}}^2 < c_{g+1}\} a_k \tau_{t_{m+1}} \sigma_{t_{m+2}}^2 | \mathcal{F}_{t_m}] \]
\[ = a_k \exp(r \Delta t) \int_{x_{ig,kh}^d}^{x_{ig,kh}^u} \frac{1}{\sqrt{2\pi}} \left[ \omega + \alpha c_h^{s/2}(|x - \lambda| - \theta(x - \lambda))^s + \beta c_h^{s/2}2/s \right] \]
\[ \exp\left(-\frac{(x - \sqrt{c_h \Delta t})^2}{2}\right) dx, \]

and \( Q_{ig,kh} \) and \( R_{ig,kh} \) will be evaluated numerically. Then we can easily find the approximate value function

\[ \tilde{V}_m(a_k, c_h, b_t) = \max(\tilde{V}_{t_m}^h(a_k, c_h, b_t), (d_{kl} - K)^+) \tag{4.7} \]

With these values we can obtain \( \tilde{V}_{t_m} \) by interpolation as explained previously. We iterate all of these integration and interpolation from terminal date to initial state where the value \( \tilde{V}_0 \) is finally found.

Note that in homoscedasticity case we can choose the constant grid which allows us to precompute the expectations \( H_{ig,kh}, P_{ig,kh}, Q_{ig,kh}, \) and \( R_{ig,kh} \) before the iteration and makes the evaluation along the vertexes very fast. However due to the heteroscedastic nature of GARCH, the probability distribution of the state variable varies over time. An adapting grid is more appropriate which means that the values of \( H_{ig,kh}, P_{ig,kh}, Q_{ig,kh}, \) and \( R_{ig,kh} \) depend on time \( t_m \), so we have to recompute these expectations at each time step, which increases the calculation amount substantially. And with an additional variable \( \sigma_{t_m}^2 \), the calculation here is exponentially heavier than that of Ben-Ameur et al.\([3]\).

**Remark 4.1.** The time complexity of this algorithm to compute the value function is \( O(np^4z^4q) \) to calculate the sum in equation (4.6), plus \( O(npzq) \) to solve the linear system for determining the coefficients in equation (4.1). So the overall time complexity is \( O(np^4z^4q) \). For comparison, the time complexity of the algorithm of Ben-Ameur et al.\([3]\) is \( O(np^4q) \). This substantiates that with a linear piecewise polynomial approximation, conditional time-varying variance aggravates the calculation greatly. As for the memory usage, this algorithm needs to store value function matrix at time \( t_{n-1} \) and \( t_m \) one of which has \( pqzq \) entries, plus \( 64[pqzq - (pq + pz + qz) + (p + z + q) - 1] \) coefficients in equation (4.1) and \( p + z + q \) elements in vector \( a, c, \) and \( b \). So at least we need a total of

\[ 8(2pqzq + 64(pqzq - (pq + pz + qz) + (p + z + q) - 1) + (p + z + q)) = 528pqzq - 512(pq + pz + qz) + 520(p + z + q) - 512 \]

bytes of memory where integers occupy 4 bytes and reals 8 bytes.
4.2. Distribution Approximation. We now propose an alternative method to approximate the value function $V_{t_m}$ which involves in approximating the normal distribution of $\xi_{t_{m+1}}$ instead of approaching the value function itself. More specifically, based on the De Moivre-Laplace theorem, we know that

$$\Pr(a' < (X - n'p')(n'p'q')^{-1/2} < b') = \frac{1}{\sqrt{2\pi}} \int_{a'}^{b'} e^{-u^2/2} du = N(b') - N(a'),$$

where $X$ follows a binomial distribution with parameters $n'$, $p'$ and $q' = 1 - p'$. In this paper we use an improved version of this, which is obtained by a continuity correction,

$$\Pr(X \leq x) \approx N((x + 0.5 - n'p')(n'p'q')^{-1/2}). \tag{4.8}$$

Its accuracy for various values of $n'$ and $p'$ has been assessed by Raff [25] and Peizer and Pratt [24]. And we use the rule of thumb $n'p'q' > 9$, which is studied by Schader and Schmidt [30]. Their study also showed that for a fixed $n'$ the maximum absolute error is minimized when $p' = q' = 1/2$, which implies that we have to choose a $n'$ greater than 36.

Then we impose that

$$\xi_{t_{m+1}} = \frac{x + 0.5 - n'p'}{\sqrt{n'p'q'}},$$

which means we replace the continuous variable $\xi_{t_{m+1}}$ with a discrete random variable, so it now can only take finite values. As a result, we can approximate the equation (2.8) by

$$\hat{V}_{t_m}(S_{t_m}, \sigma^2_{t_{m+1}}, \overline{S}_{t_{m-1}}) = \rho E_Q[\hat{V}_{t_{m+1}}(S_{t_{m+1}}(\xi_{t_{m+1}}), \sigma^2_{t_{m+2}}(\xi_{t_{m+1}}), \overline{S}_{t_m}) | \mathcal{F}_{t_m}]$$

$$= \rho \sum_{x=0}^{n'} \Pr(X = x)V_{t_{m+1}}$$

$$\left( S_{t_{m+1}} \left( \frac{x + 0.5 - n'p'}{\sqrt{n'p'q'}} \right), \sigma^2_{t_{m+2}} \left( \frac{x + 0.5 - n'p'}{\sqrt{n'p'q'}} \right), \overline{S}_{t_m} \right), \tag{4.9}$$

where $S_{t_{m+1}}(\xi_{t_{m+1}})$ and $\sigma^2_{t_{m+2}}(\xi_{t_{m+1}})$ mean they are functions of $\xi_{t_{m+1}}$, and

$$\Pr(X = x) = \left( \frac{n'}{x} \right) p^{x} q^{n'-x}.$$

Thus as usual we have

$$\hat{V}_{t_m}(S_{t_m}, \sigma^2_{t_{m+1}}, \overline{S}_{t_{m-1}}) = \max(\hat{V}_{t_m}^h(S_{t_m}, \sigma^2_{t_{m+1}}, \overline{S}_{t_{m-1}}), V_{t_m}^e(\overline{S}_{t_m})).$$

We build the same partitions for $S_{t_m}$, $\sigma^2_{t_{m+1}}$, and $\overline{S}_{t_{m-1}}$ as in the previous section, and start the iteration from the time $t_{n-1}$, where we have closed-form solution to the value function, towards the initial state $t_0$. For those points which are not at any vertex we simply use interpolation and extrapolation to find the values of them.

Remark 4.2. The overall time complexity of the algorithm to compute the value function is $O(npzqn')$ to evaluate the function in equation (4.9), which makes this algorithm quite promising when compared with the trilinear approximation introduced in last section. And it only takes $16pzn + 8(p + z + q)$ bytes to store value
function matrix and the vector \( a, c, \) and \( b \). Moreover, the following convergence analysis will guarantee its accuracy.

4.3. **Convergence Analysis.** As discussed by Ben-Ameur et al.\cite{Ben-Ameur}, because the state space is unbounded and the value function is an increasing function of its inputs to prove the convergence of our algorithm as the partition becomes smaller and smaller is not an easy thing. Following the work of Ben-Ameur et al.\cite{Ben-Ameur} we first show that even with heteroscedasticity when \( mc = \min(a_p, b_q) \rightarrow +\infty \) the probability that the trajectory of \( \{(S_{t_m}, \overline{S}_{t_m}), 0 \leq m \leq n\} \) ever exits the box \((0, a_p] \times (0, b_q] \) still decreases to 0 at a rate faster than \( O(1/pl(mc)) \) where the \( pl(mc) \) could be any polynomial of \( mc \).

\[
\text{Proposition 4.3. If } \begin{matrix}
S_{t_{m+1}}/S_{t_m} \\
& \leq n-1
\end{matrix}
\text{ we have }
\begin{align}
\text{Prob} \left( \max_{0 \leq m \leq n-1} S_{t_{m+1}}/S_{t_m} \right) &= \text{Prob} \left( \max_{0 \leq m \leq n-1} \ln(S_{t_{m+1}}/S_{t_m}) > \ln mc \right) \\
&= 1 - N \left( \frac{\ln mc - \left( r - \frac{1}{2} \sigma_{t_{m+1}}^2 \right) \Delta t}{\sigma_{t_{m+1}} \sqrt{\Delta t}} \right) \\
&\quad + \exp \left( \frac{2(r - \frac{1}{2} \sigma_{t_{m+1}}^2) \ln mc}{\sigma_{t_{m+1}}^2} \right) \\
&\quad \times N \left( - \frac{\ln mc - \left( r - \frac{1}{2} \sigma_{t_{m+1}}^2 \right) \Delta t}{\sigma_{t_{m+1}} \sqrt{\Delta t}} \right) \\
&= O \left( \frac{1}{\ln mc} \exp \left( - \frac{\ln^2 mc}{2 \sigma_{t_{m+1}}^2 \Delta t} \right) + O(\ln mc) \right) \\
&= O \left( \frac{1}{\ln mc} mc^{-\ln mc/(2\sigma_{t_{m+1}}^2 \Delta t)} \right). \quad (4.10)
\end{align}
\]

Ben-Ameur et al.\cite{Ben-Ameur} also shows that when one of \( S_{t_m} \) and \( \overline{S}_{t_m} \) tends to be infinity the error of the value function only increases linearly. As for the \( \sigma_{t_{m+1}}^2 \) dimension of \( V_{t_m} \), we think 1 is large enough and use it as its upper limit. Consequently, the approximation error outside of the cube \( C = (0, a_p] \times (0, b_q] \) is negligible if \( a_p \) and \( b_q \) are large enough. This will help us to prove the following proposition.

Define \( \sigma_a = \sup_{1 \leq i \leq p} (a_i - a_{i-1}), \sigma_c = \sup_{1 \leq g \leq z} (c_g - c_{g-1}), \) and \( \sigma_b = \sup_{1 \leq j \leq q} (b_j - b_{j-1}) \).

\textbf{Proposition 4.3. If } p \rightarrow +\infty, z \rightarrow +\infty, q \rightarrow +\infty, a_p \rightarrow +\infty, c_z \rightarrow 1, b_q \rightarrow +\infty, \sigma_a \rightarrow 0, \sigma_c \rightarrow 0, \text{ and } \sigma_b \rightarrow 0, \text{ then for any constant } \epsilon^0, \text{ we have }

\[
\sup_{0 \leq m < n} \sup_{(S_{t_{m+1}}^2, \overline{S}_{t_{m+1}} \in (0, \epsilon^0) \times (0, \epsilon^0)} \left| V_{t_m} (S_{t_m}, \sigma_{t_{m+1}}^2, \overline{S}_{t_{m-1}}) - V_{t_m} (S_{t_m}, \sigma_{t_{m+1}}^2, \overline{S}_{t_{m-1}}) \right| \rightarrow 0.
\]

\textbf{Proof.} We first show that with heteroscedasticity the derivatives of \( V_{t_m}^h \) \( V_{t_m}^h (S_{t_m}, \sigma_{t_{m+1}}^2, \overline{S}_{t_m}) \) and \( V_{t_m} (S_{t_m}, \sigma_{t_{m+1}}^2, \overline{S}_{t_m}) \) with respect to \( S_{t_m} \) are also bounded by a constant \( L_m (0 \leq m \leq n) \), which is defined as

\[
L_n = 0,
\]

and

\[
L_m = \left( L_{m+1} + \frac{m}{(m+1)^2} \right) pE \left[ \tau_{t_{m+1}} | \mathcal{F}_{t_m} \right].
\]
\[ L_{m+1} = \frac{m}{(m+1)^2}. \]

The arguments are similar to that of proposition 4 of Ben-Ameur et al.\[3\]. Specifically, for \( S_{tm}^1 \leq S_{tm}^2 \), we have
\[ V^h_{tm}(S_{tm}^2, \sigma_{tm+1}^2, \bar{S}_{tm}) - V^h_{tm}(S_{tm}^1, \sigma_{tm+1}^2, \bar{S}_{tm}) \leq (S_{tm}^2 - S_{tm}^1)L_m, \tag{4.11} \]
and
\[ V^h_{tm}(S_{tm}^2, \sigma_{tm+1}^2, \bar{S}_{tm}) - V^h_{tm}(S_{tm}^1, \sigma_{tm+1}^2, \bar{S}_{tm}) \leq (S_{tm}^2 - S_{tm}^1)L_m. \tag{4.12} \]

Moreover because \( V^h_{tm}(S_{tm}^2, \sigma_{tm+1}^2, \bar{S}_{tm}) \) is continuous, bounded, and nondecreasing in \( \sigma_{tm+1}^2 \), then in a large enough span \((0, 1]\) for it, the derivative of \( V^h_{tm} \) with respect to \( \sigma_{tm+1}^2 \) is always bounded by a positive sequence \( B_m \). The sequence \( B_m \) exists because a bounded value less another bounded value is still bounded. This means, for \( \sigma_{tm+1}^2 \leq \sigma_{tm+1}^2 \):
\[ V^h_{tm}(S_{tm}^2, \sigma_{tm+1}^2, \bar{S}_{tm}) - V^h_{tm}(S_{tm}^1, \sigma_{tm+1}^2, \bar{S}_{tm}) \leq (\sigma_{tm+1}^2 - \sigma_{tm+1}^2)B_m, \tag{4.13} \]
and
\[ V^h_{tm}(S_{tm}^2, \sigma_{tm+1}^2, \bar{S}_{tm}) - V^h_{tm}(S_{tm}^1, \sigma_{tm+1}^2, \bar{S}_{tm}) \leq (\sigma_{tm+1}^2 - \sigma_{tm+1}^2)B_m. \tag{4.14} \]

Furthermore, the Lemma 3.1, equation (4.11), (4.12), (4.13), and (4.14) also hold for \( V^h_{tm}(\cdot, \cdot, \bar{S}_{tm-1}) \) and \( V^h_{tm}(\cdot, \cdot, \bar{S}_{tm-1}) \) because their derivatives do not exceed those of \( V^h_{tm}(\cdot, \cdot, \bar{S}_{tm}) \) and \( V^h_{tm}(\cdot, \cdot, \bar{S}_{tm}) \).

With these slope bounds and Proposition 3.1 we have
\[ \sup_{1 \leq k \leq p, 1 \leq h \leq c, 1 \leq i \leq q} |V^h_{tm}(a_k, a_i, b_{l+1}) - V^h_{tm}(a_k, c_i, b_{l+1})| \leq L_m \omega_c + B_m \omega_c + \omega_b. \tag{4.15} \]

Similar to Ben-Ameur et al.\[3\] we define \( \varrho_n = 0 \), and \( \varrho_m = 2(\varrho_{m+1} + L_m \omega_a + B_m \omega_c + \omega_b) \) for \( 0 \leq m \leq n \). Also we define an increasing sequence of cubes \( C_m \neq (0, c_m^0) \times (0, 1] \times (0, c_m^0) \) for \( 0 \leq m < n \), where \( c_m^0 \) is chosen arbitrarily and \( c_m^0 \geq c_0 \) such that
\[ E^q_m[|\bar{V}_{tm+1}(S_{tm} \tau_{tm+1}, \sigma_{tm+1}^2, \bar{S}_{tm}) - V^h_{tm}(S_{tm} \tau_{tm+1}, \sigma_{tm+1}^2, \bar{S}_{tm})| \times I((S_{tm} \tau_{tm+1}, \sigma_{tm+1}^2, \bar{S}_{tm}) \notin C_{m+1}) |\mathcal{F}_{tm}] \leq \varrho_{m+1}. \tag{4.16} \]

Such a \( \varrho_{m+1} \) always exists because both \( \bar{V}_{tm+1} \) (equation (4.9)) and \( V_{m+1} \) are bounded and the probability that \( (S_{tm} \tau_{tm+1}, \sigma_{tm+1}^2, \bar{S}_{tm}) \) exerts the cube \( C_{m+1} \) in the first and third dimension decreases faster than the inverse of any polynomial of \( c_{m+1}^0 \) when \( c_{m+1}^0 \) tends to be infinity.

Then we should show that \( |\bar{V}_{tm} - V_{tm}| \) is bounded by \( \varrho_m \) over the cube \( C_m \). Again we will use mathematical induction. First note that this holds for \( m = n \) and \( m = n-1 \), where \( \bar{V}_{tm} = V_{tm} \). Next we assume that \( |\bar{V}_{tm+1} - V_{tm+1}| \leq \varrho_{m+1} \) for \((S_{tm+1} \tau_{tm+1}, \sigma_{tm+1}^2, \bar{S}_{tm+1}) \in C_{m+1} \). Consequently, for \((S_{tm} \tau_{tm+1}, \sigma_{tm+1}^2, \bar{S}_{tm+1}) \in C_{m+1} \), we have
\[ \bar{V}_{tm}(S_{tm} \tau_{tm+1}, \sigma_{tm+1}^2, \bar{S}_{tm+1}) - V^h_{tm}(S_{tm} \tau_{tm+1}, \sigma_{tm+1}^2, \bar{S}_{tm+1}) \leq \bar{V}_{tm+1}(S_{tm} \tau_{tm+1}, \sigma_{tm+1}^2, \bar{S}_{tm+1}) - V^h_{tm}(S_{tm} \tau_{tm+1}, \sigma_{tm+1}^2, \bar{S}_{tm+1}) \]
\[ \leq \rho \int_0^{\tau_{tm+1}} |\bar{V}_{tm+1}(S_{tm} \tau, \sigma_{tm+1}^2, \bar{S}_{tm+1}) - V^h_{tm+1}(S_{tm} \tau, \sigma_{tm+1}^2, \bar{S}_{tm+1})| f(\tau | \mathcal{F}_{tm}) d\tau \]
These facts imply that in the cube $C_V$ when $n$

Note that Schader and Schmid [30] has shown that under the rule of thumb $n'p'q' > 9$ the absolute error of our distribution approximation is $0.0007(n'p'q')^{-1/2}$ when $p' = 0.5$, which makes it easy for us to choose the grid size such that

$$V_m - \hat{V}_m \leq L_m \omega_a + B_m \omega_c + \omega_b.$$ 

when $V_m > \hat{V}_m$. Then we obtain

$$|\hat{V}_m - V_m| - 2(L_m \omega_a + B_m \omega_c + \omega_b) \leq \hat{V}_m - V_m \leq 2p \theta_{m+1}.$$ 

These facts imply that in the cube $C_m$ and under the assumptions of this proposition

$$|\hat{V}_m(S_{t_m}, \sigma^2_{t_{m+1}}, \tau_{t_{m-1}}) - V_m(S_{t_m}, \sigma^2_{t_{m+1}}, \tau_{t_{m-1}})| \leq 0 \theta_{m+1} + 2(L_m \omega_a + B_m \omega_c + \omega_b)$$

This completes the proof. \hfill \Box

4.4. Grid Choice. Following Ben-Ameur et al.[3] we provide a purely heuristic but not necessarily optimal way to partition our state space here. First note that based on Theorem 2.1 and with the forecast origin as time $t_0 = 0$, the $(n - 1)$-step ahead forecast for the conditional variance is

$$E^Q[\sigma^2_{t_{n-1}} | F_0] = \sigma^2_0(n - 1) = \omega \frac{1 - B^{n-2}(s, \lambda)}{1 - B(s, \lambda)} + B^{n-2}(s, \lambda) \sigma^2_0(1),$$

where

$$B(s, \lambda) = \frac{1}{2\pi} \alpha[(1 - \theta)^s A(s, \lambda) + (1 + \theta)^s A(s, -\lambda)] + \beta,$$

and $\sigma^2_0(\cdot)$ is the forecast with origin time 0. The $2s$-th conditional moment of $\{\tau_{t_{n-1}}\}$ is

$$E^Q[\sigma^2_{t_{n-1}} | F_0] = \theta \frac{1 - C^{n-2}(s, \lambda)}{1 - C(s, \lambda)} + C^{n-2}(s, \lambda) \sigma^2_0(1),$$

where

$$\vartheta = \omega^2 + 2\omega B(s, \lambda) \sigma^2_0(n - 2),$$

$$C(s, \lambda) = \int_{-\infty}^{+\infty} \frac{1}{\sqrt{2\pi}} \alpha([x - \lambda] - \theta(x - \lambda))^s + \beta^2 \exp(-\frac{x^2}{2}) \, dx$$

$$= \frac{1}{\sqrt{2\pi}} [D(s, \lambda, \theta) + D(s, -\lambda, -\theta)],$$

$$D(s, \lambda, \theta) = \int_{-\lambda}^{+\infty} [\alpha(1 - \theta)^s (x - \lambda)^s + \beta^2 \exp(-\frac{x^2}{2}) \, dx$$
\[= 2^{2s-1/2} \lambda^{2s} \alpha^2 (1 - \theta)^{2s} e^{-\frac{\beta}{2} \pi^2 \frac{1}{1 - 2s} \pi^2 \lambda^{2s}} \]

\[-2 \lambda^2 + \lambda^4 + 4 \lambda^2 s + 2s - 1 \frac{L(\frac{1}{2} - s, \frac{1}{2}, \frac{1}{2} \lambda^2) \cos^{-1}(\pi s) \frac{1}{2} - \frac{s}{2}}{\Gamma(2 - s)} \]

\[-2^{1-s} \pi^2 \lambda^{2s} \frac{-1 + \lambda^2 + 2s}{2s - 1} \frac{L(\frac{1}{2} - s, \frac{3}{2}, \frac{1}{2} \lambda^2) \cos^{-1}(\pi s) \frac{1}{2} - \frac{s}{2}}{\Gamma(2 - s)} \]

\[-2^{-s} \pi^2 \lambda^{2s+2} \frac{-1 + \lambda^2 + 2s}{2s - 1} \frac{L(-s, \frac{1}{2}, \frac{1}{2} \lambda^2) \sin^{-1}(\pi s) \Gamma^{-1}(\frac{3}{2} - s)}{\Gamma(2 - s)} \]

\[+ 2^{-s} \pi^2 \lambda^{2s+3} \frac{L(-s, \frac{3}{2}, \frac{1}{2} \lambda^2) \sin^{-1}(\pi s) \Gamma^{-1}(\frac{3}{2} - s)}{\Gamma(2 - s)} \]

\[+ 2^{s+\frac{1}{2}} \alpha (1 - \theta)^{s} \lambda^{-s} e^{-\frac{\lambda^2}{2} \pi^{-1} (2^{-s} - 1) \pi^2 \lambda^{2s}} \]

\[-2 \lambda^2 + \lambda^4 + 2 \lambda^2 s + s - 1 \frac{L(\frac{1}{2} - s, \frac{1}{2}, \frac{1}{2} \lambda^2) \cos^{-1}(\pi s) \frac{1}{2} - \frac{s}{2}}{\Gamma(2 - s)} \]

\[-2^{-\frac{1}{2}} \pi^2 \lambda^{2s+2} \frac{-1 + \lambda^2 + s}{s - 1} \frac{L(-s, \frac{1}{2}, \frac{1}{2} \lambda^2) \sin^{-1}(\pi s) \Gamma^{-1}(\frac{3}{2} - \frac{1}{2} s)}{\Gamma(2 - \frac{1}{2} s)} \]

\[-2^{-\frac{3}{2}} \pi^2 \lambda^{2s+1} \lambda^2 s + L(-s, \frac{1}{2}, \frac{1}{2} \lambda^2) \sin^{-1}(\pi s) \Gamma^{-1}(\frac{3}{2} - \frac{1}{2} s) \]

\[+ 2^{-\frac{3}{2}} \pi^2 \lambda^{2s+3} L(-s, \frac{3}{2}, \frac{1}{2} \lambda^2) \sin^{-1}(\pi s) \Gamma^{-1}(\frac{3}{2} - \frac{1}{2} s) \]

\[+ \frac{1}{\sqrt{2}} \beta^2 \pi^2 (1 - e(\frac{\lambda}{\sqrt{2}})), \]

and \(e(\cdot)\) is the error function. To partition \(\sigma_{t_{m+1}}^2\) dimension, we take

\[c_1 = (\sigma_0^2(n - 1))^{2/s} - 5 \sqrt{(E^Q[\sigma_t^2 \mid F_0])^{2/s} - (\sigma_0^2(n - 1))^{4/s}}, \]

\[c_{2-1} = (\sigma_0^2(n - 1))^{2/s} + 5 \sqrt{(E^Q[\sigma_t^2 \mid F_0])^{2/s} - (\sigma_0^2(n - 1))^{4/s}}, \]

and

\[c_2 = (\sigma_0^2(n - 1))^{2/s} + 6 \sqrt{(E^Q[\sigma_t^2 \mid F_0])^{2/s} - (\sigma_0^2(n - 1))^{4/s}}. \]

For the distance between \(c_1\) and \(c_{2-1}\), we simply space it evenly.

For \(S_{t_m}\) and \(S_{t_{m-1}}\) dimensions, we first make \(p = q, a_i = b_i\) for all \(i\). Then with the facts

\[E^Q[\ln S_{t_{n-1}}^Q \mid F_0] = rt_{n-1} - \frac{1}{2} \sum_{t=t_1}^{t_{n-1}} \sigma_t^2, \text{ and } \text{Var}^Q[\ln S_{t_{n-1}}^Q \mid F_0] = \sum_{t=t_1}^{t_{n-1}} \sigma_t^2, \]

we let

\[a_1 = S_0 \exp(rt_{n-1}) \left( \frac{1}{2} \sum_{i=1}^{n-1} (\sigma_0^2(i))^{2/s} - 5\left(\sum_{i=1}^{n-1} (\sigma_0^2(i))^{2/s}\right)^{1/2} \right), \]

\[a_{p-1} = S_0 \exp(rt_{n-1}) \left( \frac{1}{2} \sum_{i=1}^{n-1} (\sigma_0^2(i))^{2/s} + 5\left(\sum_{i=1}^{n-1} (\sigma_0^2(i))^{2/s}\right)^{1/2} \right), \]

and

\[a_p = S_0 \exp(rt_{n-1}) \left( \frac{1}{2} \sum_{i=1}^{n-1} (\sigma_0^2(i))^{2/s} + 6\left(\sum_{i=1}^{n-1} (\sigma_0^2(i))^{2/s}\right)^{1/2} \right). \]
For \( 2 \leq i \leq p - 2 \), \( a_i \) is the quantile of order \((i - 1)/(p - 2)\) of the lognormal distribution of \( S_{t-1}/S_0 \).

In our numerical analysis of next section, we run the algorithm repeatedly with increasingly finer partitions and we examine the changes in the option value. The experiments stop when there is no significant change in option value.

### 5. The Numerical Experiments

Example 1. To obtain the option values under GARCH process, we recognize that the asset price \( S_t \) and the conditional volatility \( \sigma_t \) can serve as sufficient statistics. For the GARCH model specified in equation (2.6) in our numerical example, we take the parameter values from the estimation results of Ding et al.[10], which are \( \omega = 1.4 \times 10^{-5} \), \( \alpha = 0.083 \), \( \beta = 0.92 \), \( \theta = 0.373 \), \( \lambda = 7.452 \times 10^{-3} \), and \( s = 1.43 \) respectively. These parameters together imply that the annualized (based on 365 days) stationary standard deviation is approximately 25.58\%. Then we consider an American-style Asian call option with \( S_0 = 100 \), \( K = 100 \), \( T = 1/4 \) years, \( r = 0.05 \), \( n^* = 1 \), \( n = 13 \), and initial volatility \( \sigma_0 = 0.2558 \). Also we consider some slight modifications of this example. We change the relative asset prices by decreasing and increasing the strike price \( K \) and extend the time horizon from 0.25 to 0.5 while keeping \( n = 13 \) fixed. And with GARCH model we can test the impact of different initial volatilities on the option values. To do this, we set our initial volatility to be 20\% below the stationary level and 20\% higher. As we know, the higher initial volatility will lead to a higher option value and the lower, the lower, which shows a positive relationship.

<table>
<thead>
<tr>
<th>((K, T, \sigma_0))</th>
<th>(p \times z \times q)</th>
<th>(20 \times 20 \times 20)</th>
<th>(30 \times 20 \times 30)</th>
<th>(40 \times 20 \times 40)</th>
<th>(50 \times 20 \times 50)</th>
<th>(60 \times 20 \times 60)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(95, 0.25, 0.2047)</td>
<td>4.99339</td>
<td>4.99567</td>
<td>4.99678</td>
<td>4.99744</td>
<td>4.99787</td>
<td></td>
</tr>
<tr>
<td>CPU(hh:mm:ss)</td>
<td>00:02:40</td>
<td>00:12:05</td>
<td>00:39:59</td>
<td>01:38:02</td>
<td>03:33:46</td>
<td></td>
</tr>
<tr>
<td>(100, 0.25, 0.2047)</td>
<td>1.46930</td>
<td>1.47048</td>
<td>1.47105</td>
<td>1.47138</td>
<td>1.47161</td>
<td></td>
</tr>
<tr>
<td>CPU(hh:mm:ss)</td>
<td>00:02:05</td>
<td>00:09:39</td>
<td>00:31:45</td>
<td>01:19:24</td>
<td>02:49:27</td>
<td></td>
</tr>
<tr>
<td>(100, 0.25, 0.2558)</td>
<td>1.82496</td>
<td>1.82609</td>
<td>1.82664</td>
<td>1.82697</td>
<td>1.82719</td>
<td></td>
</tr>
<tr>
<td>CPU(hh:mm:ss)</td>
<td>00:02:10</td>
<td>00:10:21</td>
<td>00:43:32</td>
<td>01:20:47</td>
<td>02:44:54</td>
<td></td>
</tr>
<tr>
<td>(100, 0.25, 0.3070)</td>
<td>2.18040</td>
<td>2.18152</td>
<td>2.18206</td>
<td>2.18238</td>
<td>2.18250</td>
<td></td>
</tr>
<tr>
<td>CPU(hh:mm:ss)</td>
<td>00:02:04</td>
<td>00:10:22</td>
<td>00:36:08</td>
<td>01:20:40</td>
<td>03:05:19</td>
<td></td>
</tr>
<tr>
<td>(105, 0.25, 0.3070)</td>
<td>0.40899</td>
<td>0.41502</td>
<td>0.41426</td>
<td>0.41075</td>
<td>0.41443</td>
<td></td>
</tr>
<tr>
<td>CPU(hh:mm:ss)</td>
<td>00:01:04</td>
<td>00:03:58</td>
<td>00:11:43</td>
<td>00:28:25</td>
<td>00:49:29</td>
<td></td>
</tr>
<tr>
<td>(105, 0.50, 0.3070)</td>
<td>0.71218</td>
<td>0.71314</td>
<td>0.71362</td>
<td>0.71651</td>
<td>0.71442</td>
<td></td>
</tr>
<tr>
<td>CPU(hh:mm:ss)</td>
<td>00:00:56</td>
<td>00:04:02</td>
<td>00:11:14</td>
<td>00:26:10</td>
<td>00:53:15</td>
<td></td>
</tr>
</tbody>
</table>
We report our simulation results in Table 1, where we implemented the approximation method introduced in Section 4.2 with \( n' = 40 \) in a MATLAB R14 programming environment. Computations have been executed on a 1.70GHz Pentium M PC with 512 Mbytes of memory and running the Windows operating system. The CPU times are given for every set of parameter values with different grid spaces and are reported in hours:minutes:seconds format.

The approximation of option value converges rapidly even at the very coarse grid. With the grid size refined the changes of values are still at one in a thousand but the CPU times increase exponentially, which is a sharp contrast to the results of Ben-Ameur et al.[3]. The finest grid used by those authors is \( 2400 \times 2400 \), but the CPU time required by our finest grid \( 60 \times 20 \times 60 \), which is even sparser than the coarsest grid of theirs, is much longer than the time required by their finest grid. The main reason behind this is that they considered a homogeneous volatility model which implies a 2-dimension value function, while we incorporate heteroscedastic conditional volatility and the additional variable leads directly to the huge calculation amount. As analyzed in Section 4.1, if we applied the trilinear approximation, just following Ben-Ameur et al.[3], to this 3-dimension case the CPU time would be practically intolerable. A grid scheme \( 150 \times 150 \times 150 \) like the coarsest one in Ben-Ameur et al.[3] will require 1.748Gbytes of memory for data storage, which is far beyond the capacity of our machine. Instead we compromise the normally distributed variable to be a binomial variable which only takes finite elements. This method reduces the time complexity greatly, does not require huge memory, and the convergence is still satisfactory. Furthermore, like the approach of Longstaff and Schwartz [21], our method also could take the advantage of parallel computing architecture. While they can separate the path generation and the estimation of conditional expectation function across CPUs, we actually could divide our whole cube into different small cubes and distribute the approximation in those small cubes to different CPUs, which should conduct the calculation simultaneously. The approximated small cubes then could be aggregated across CPUs to form the composite cube in the whole state space. As such parallel computation should significantly improve the computational speed.

After a close look at the prices in Table 1, we find that a lower strike price corresponds to a higher option value, a higher one to a lower value, and the extended maturity results in larger option prices, just as expected. We also check the impact of different initial volatilities which are at the stationary level (0.2558), 20% below the stationary level (0.2047), and 20% above (0.307), respectively. The simulation results are consistent with our intuition: a higher volatility leads to a more expensive option contract because it’s more possible for the underlying asset to achieve a larger average.

Example 2. As documented by Rubinstein [29] and Sheikh [32], the implied volatilities of traded options exhibit a systematic pattern with respect to time to maturity and the relative relationship between asset price and strike price. They suggest a U-shape implied volatility graph with respect to different asset price to strike price ratios, which is also known as volatility smile. Since there is no analytical value function for the American-style Asian option, we therefore use the approach of Ben-Ameur et al.[3] to invert the GARCH option prices instead, which enables us to peek the pattern of the implied volatility in a parsimonious way. We plot the polynomial trendline of implied volatility extracted in this way
vs asset price to strike price ratio for low and high initial volatility respectively. In both Figure 1 and Figure 2, the implied volatility trendlines for different maturities roughly show a U-shape pattern.

We recognize that the estimated parameters of Ding et al.[10] suggest a significant GARCH process, so the volatility must show a strong clustering phenomenon. Also we note that the GARCH model is known to be asymmetrical and skewed to the left side. The implication of these facts is that for a longer maturity the likelihood of observing low variance is higher. Figure 1 and Figure 2 plot the implied volatilities for three different maturities: $T = 1/12$ years, $T = 1/4$ years, and $T = 1/2$ years where the $T = 1/2$ case shows more low variance states, which is consistent with Duan [12].

Rubinstein [29] and Sheikh [32] also showed a positive relationship between the time to maturity and the option’s implied volatility for the at-the-money calls and a possible reversal over a different time period. In our case as shown by Figure 1 and Figure 2, the implied volatility for the shortest-maturity option is always the highest for both low and high initial conditional volatility case. This example could be thought as an evidence of reversal reported by Rubinstein [29] and Sheikh [32].

**Figure 1. Implied Volatility of the GARCH Option Price with a Low Initial Conditional Volatility**

Example 3. To examine the impact of the chances of early exercise we now alter the parameter $n$ (the times of observations before maturity) while holding the other parameters in Table 1 unchanged. The simulation results are reported in Table 2. We can see clearly from Table 2 that the option price is a decreasing function of $n$, which agrees with the findings of Ben-Ameur et al.[3]. Those authors suggest a possible reason that goes as follows. The increasing observation dates stabilize the average value of underlying asset and this stabilization overrides the advantage of early exercise. Our findings here actually lend support to their explanation.
Example 4. In this example we will test the sensitivity of option price to the parameters of GARCH model. Figure 3, Figure 4 and Figure 5 plot the correlation between the GARCH option price and $s$ (the Box-Cox transformation parameter), $\theta$ (the leverage effect), and $\lambda$ (the unit risk premium), respectively. The contract we use here is $(100, 0.25, 0.2558)$ for $(K, T, \sigma_0)$.

The trend in Figure 3 shows that the GARCH option price is a positive function of the parameter $s$, especially when $s > 1$ the price increases with an increasing rate. The implication is that Taylor [33]/Schwert [31]’s GARCH with $s = 1$ generates a relatively low price whereas the GARCH of Bollerslev [5] with $s = 2$ will lead to a very high option value.
The U-shape of Figure 4 means that if $\theta$ significantly differs from 0, the asymmetric response of volatility to different shocks will give rise to higher option values. Moreover, no matter it is negative shock or positive shock that has a deeper impact on current conditional volatility than past positive or negative shocks (i.e., positive value of $\theta$ or negative $\theta$), the $\theta$s with different signs all result in higher option prices.

Figure 5 illustrates that the option price moves in the same direction with the unit risk premium $\lambda$. This implies that for a riskier asset the option written on it under GARCH model will be more valuable.

6. Conclusion

We studied in this paper the possible numerical pricing methods for the American-style Asian option under GARCH process since the GARCH option pricing model appears to have certain desirable features and can correct the pricing biases based on the Black-Scholes model. We extend the method of Ben-Ameur et al.[3] to trilinear piecewise polynomial approximation, and analyze the time complexity and memory usage of this algorithm. We show that when the dimension of the state space becomes larger the approximation of the value function becomes much more difficult and the calculation also becomes overwhelming. This is known as the curse of dimensionality.

Instead of using piecewise polynomial to approximate the value function we propose to replace the normally distributed variable with a binomial one which only takes finite elements. The complexity analysis shows that this alternative way reduces the calculation burden significantly. We also proved the convergence of this
Figure 4. The GARCH Option Price As a Function of \( \theta \)

Figure 5. The GARCH Option Price As a Function of \( \lambda \)
method based on the continuity, monotonicity, and convexity properties of the value function. Our numerical examples illustrate that even with relatively coarse grids the option prices converge quickly. The prices dependent on the varying numbers of exercise opportunity show that the stabilization of average price offsets the advantage of more exercise chances. We also conducted the sensitivity analysis of option price with respect to model parameters such as the Box-Cox transformation parameter $s$, the leverage effect $\theta$, and the unit risk premium $\lambda$. The simulation results demonstrate that Bollerslev [5]'s GARCH will generate the highest price and the asymmetric response of volatility to different shocks and a riskier underlying asset both lead to more valuable option contract. The ultimate test of this model is, however, still its empirical performance. Since the market data is readily available and the technology to estimate GARCH parameters has already been well developed, an overall efficiency test of our numerical method is therefore easily implementable. Also note that we focus on the fixed strike Asian option in this paper, but the method can be easily adapted to apply for floating strike case since $K$ is a constant and we approximate the option value at every cube vertex.

Although we have emphasized the single lag version of GARCH model, one can extend it to multiple lags without too much difficulty. But since Assumption 2 states that the GARCH model is equivalent with the Black-Scholes model for one period options, additional lags will not improve the accuracy for short term options. Observe that we used a constant grid for all time steps, so if one uses an adapting grid which fits the distribution of value function at different time steps a better approximation is expected to be reached. Meanwhile to refine the grid is the most direct approach to improve accuracy, but to control the complexity remains a challenging problem.

References

A NUMERICAL METHOD PRICING AMERASIAN WITH GARCH


Department of Economics, Faculty of Arts and Social Sciences, National University of Singapore, AS2 Level 6, 1 Arts Link, Singapore 117570

E-mail address: dan.shao@nus.edu.sg