Computational Methods for Differential Equations
http://cmde.tabrizu.ac.ir
Vol. 9, No. 4, 2021, pp. 1059-1068
DOI:10.22034/cmde.2020.27563.1369

A new methodology to estimate constant elasticity of variance

Ali Beiranvand*
Faculty of Mathematical Sciences,
University of Tabriz, Tabriz, Iran.
E-mail: a_beiranvand@tabrizu.ac.ir

Karim Ivaz
Faculty of Mathematical Sciences,
University of Tabriz, Tabriz, Iran.
E-mail: ivaz@tabrizu.ac.ir

Hamzeh Beiranvand
Department of Electrical Engineering,
Lorestan University, Khoramabad, Lorestan, Iran.
E-mail: beiranvand.ha@fe.lu.ac.ir

Abstract This paper introduces a novel method for estimation of the parameters of the constant elasticity of variance model. To do this, the likelihood function will be constructed based on the approximate density function. Then, to estimate the parameters, some optimization algorithms will be applied.

Keywords. Finance, Constant elasticity of variance; Likelihood function; Particle swarm algorithm; General relativity search algorithm; Nelder-Mead algorithm.

2010 Mathematics Subject Classification. 91Gxx, 62G07, 90Cxx.

1. Introduction

The constant elasticity of variance evolves according to the following stochastic differential equation (SDE)

$$dS_t = \mu S_t dt + \delta S_t^{\frac{\beta}{2}} dW_t, \tag{1.1}$$

where μ is a constant, δ is scaling parameter, β is called elasticity and W_t is Wiener process. In finance this model is used as an underlying asset model to describe share price. To capture shortcomings of the Black-Scholes model [4] such as leverage effect and volatility smile, Cox introduced the CEV model [6]. The inverse relationship between the share price and volatility are shown by this model for $\beta < 1$. This model has been used by many authors to price and hedge in option pricing problems. In case of European options see for example [6, 7, 17] and for American options refer to [5, 12, 16].

Beckers used the regression approach to estimate the parameters of CEV model [2]. Emanuel and MacBeth [9] applied an optimization technique to find β. Constructing

Received: 25 May 2018 ; Accepted: 5 September 2020.
* Corresponding.

three likelihood functions, based on the values of β, for different values of δ, Marsh and Rosenfeld estimated β and δ for interest rates [13]. To maximize the likelihood function, Tucker et al. used the Newton-Raphson method to estimate β and δ jointly for foreign currency exchange rates [18]. To estimate the parameters of the CEV model, Melino and Turnbull maximized the likelihood function constructed from a modified form of the density function [14].

In this paper, a novel approach to estimate the parameters of the CEV model is presented which is based on optimization techniques. The proposed formulation is a simple and straightforward approach and it can be applied to any CEV process. It can also be used to estimate a vector of parameters simply.

The rest of the paper is organized as follows: In section 2 construction of the likelihood function based on the approximate transition density function will be described. Presentation of the optimization algorithms is the subject of the section 3. Numerical result will be given in section 4.

2. Estimation strategy

The transition density function of the SDE (1.1) is given by, Cox for $\beta < 2$ [6] and Emanuel and MacBeth for $\beta > 2$ [9],

$$p_{S_{t+1}|S_t}(s_{t+1},1) = |2-\beta| k^{\frac{1}{2-\beta}} (xz^{1-2\beta})^{\frac{1}{4-2\beta}} e^{-x-z} I_{\frac{1}{|2-\beta|}}(2\sqrt{xz}), \tag{2.1}$$

where I_ν is the modified Bessel function of the first kind of order ν and, for $\tau = 1$ day,

$$\begin{aligned} k &= \frac{2\mu}{\delta^2(2-\beta)(e^{\mu(2-\beta)\tau}-1)}, \\ x &= kS_t^{2-\beta} e^{\mu(2-\beta)\tau}, \\ z &= ks^{2-\beta}. \end{aligned} \tag{2.2}$$

In the case of $\beta = 2$ the CEV model turns to Black-Scholes model, that is, inserting $\beta = 2$ in (1.1) yields

$$dS_t = \mu S_t dt + \delta S_t dW_t,$$

which is the Black-Scholes equation for underlying asset. The transition density of the Black-Scholes SDE is given by lognormal distribution [4]. To estimate δ and β jointly, instead of using the true likelihood function, we will use an approximate likelihood function. In the case of using the true likelihood function a number of disadvantages occur. The true likelihood function includes the modified Bessel function, then at each evaluation the modified Bessel function must be computed which is computationally expensive. Another disadvantage is that the true likelihood as a function of β has a complicated nature. Based on the values of β, true likelihood function has three cases. However, the approximate likelihood function lacks these problems.

Now we want to describe the approximate likelihood technique used to estimate δ and β jointly. This method is easier to understand and to implement. Considering the transformation $X_t = \ln S_t$ and using Ito's lemma for $X_t = \ln S_t$ yields

$$d\ln S_t = (\mu - \frac{1}{2}\delta^2 S_t^{\beta-2})dt + \delta S_t^{\frac{\beta-2}{2}} dW_t. \tag{2.3}$$

Integrating from n to $n+1$, one can get

$$\begin{aligned}\ln S_{n+1} - \ln S_n &= \mu - \frac{1}{2}\delta^2 \int_n^{n+1} S_t^{\beta-2} dt + \delta \int_n^{n+1} S_t^{\frac{\beta-2}{2}} dW_t \\ &\approx \mu - \frac{1}{2}\delta^2 S_n^{\beta-2} + \delta S_n^{\frac{\beta-2}{2}} \epsilon_n,\end{aligned} \tag{2.4}$$

where ϵ_n is a forward $\mathcal{N}(0,1)$ increment which is independent of S_n. So using the fact that $X_n = \ln S_n$, we can write the time discrete form of SDE (2.3)

$$\begin{aligned}X_{n+1} =& X_n + \mu(X_n) + \delta e^{\frac{\beta-2}{2}X_n}\epsilon_n \\ &\sim \mathcal{N}(X_n + \mu(X_n), \delta^2 e^{(\beta-2)X_n}),\end{aligned} \tag{2.5}$$

where $\mu(X_n) = \mu - \frac{1}{2}\delta^2 e^{(\beta-2)X_n}$ is the expected value of the daily returns $X_{n+1} - X_n$, conditional on X_n and $\mathcal{N}$ is the normal distribution. Hence the approximate transition density function can be written as

$$f_{X_{n+1}|X_n}(x_{n+1}, 1) = \frac{1}{\sqrt{2\pi}\delta e^{\frac{\beta-2}{2}x_n}} \exp\{-\frac{1}{2}(\frac{x_{n+1} - x_n - \mu(x_n)}{\delta e^{\frac{\beta-2}{2}x_n}})^2\}. \tag{2.6}$$

Now the likelihood function for series $\{X_n : n = 0, 1, \cdots, N\}$ corresponding to a sample of share price $\{S_n : n = 0, 1, \cdots, N\}$ can be formed as follows

$$L(\beta, \delta) = f_{X_0}(x_0) \prod_{n=0}^{N-1} f_{X_{n+1}|X_n}(x_{n+1}, 1). \tag{2.7}$$

An advantage of using this likelihood function instead of the true likelihood function is that it can be applied for any CEV process. Then

$$l(\beta, \delta) = \ln L(\beta, \delta) = \ln f_{X_0}(x_0) + \sum_{n=0}^{N-1} \ln f_{X_{n+1}|X_n}(x_{n+1}, 1). \tag{2.8}$$

To estimate β, δ jointly, the above function must be maximized with respect to β, δ. Instead of maximizing the above function, it is appropriate to maximize the function

$$\frac{1}{N} l(\beta, \delta) = \frac{1}{N} \ln f_{X_0}(x_0) + \frac{1}{N} \sum_{n=0}^{N-1} \ln f_{X_{n+1}|X_n}(x_{n+1}, 1). \tag{2.9}$$

For large N, one can get approximately

$$\begin{aligned}\bar{l}(\beta, \delta) &= \frac{1}{N} \sum_{n=0}^{N-1} \ln(\frac{1}{\sqrt{2\pi}\delta e^{\frac{\beta-2}{2}x_n}} \exp\{-\frac{1}{2}(\frac{x_{n+1} - x_n - \mu(x_n)}{\delta e^{\frac{\beta-2}{2}x_n}})^2\}) \\ &= -\ln\delta - \frac{\beta-2}{2N} \sum_{n=0}^{N-1} x_n - \frac{1}{2}\ln 2\pi - \frac{1}{2N} \sum_{n=0}^{N-1} \frac{z_n^2}{\delta e^{\frac{\beta-2}{2}x_n}},\end{aligned} \tag{2.10}$$

where $z_n = x_{n+1} - x_n - \mu(x_n)$, for $n = 0, 1, \cdots, N-1$, are realization of the random variable $Z_n = X_{n+1} - X_n - \mu(X_n)$. Now we can define an optimization problem to estimate β, δ jointly as follows

$$\begin{aligned} & Max \;\; l_1(\beta, \delta) \\ & S.to: \\ & \quad \beta \in \mathbb{R}, \;\; \delta > 0, \end{aligned} \tag{2.11}$$

where

$$l_1(\beta, \delta) = -\ln \delta - \frac{\beta - 2}{2N} \sum_{n=0}^{N-1} x_n - \frac{1}{2N} \sum_{n=0}^{N-1} \frac{z_n^2}{\delta e^{\frac{\beta-2}{2} x_n}}. \tag{2.12}$$

3. Optimization algorithms

In order to estimate β, δ simultaneously using the above optimization problem (2.11), we need to bound β, δ. Let β_0, β_1, δ_0 and δ_1 be real numbers with $\beta_0 < \beta_1$ and $0 < \delta_0 < \delta_1$. Setting $\beta_0 \leq \beta \leq \beta_1$ and $\delta_0 \leq \delta \leq \delta_1$ one can write

$$\begin{aligned} & Max \;\; l_1(\beta, \delta) \\ & S.to: \\ & \quad \beta_0 \leq \beta \leq \beta_1, \\ & \quad \delta_0 \leq \delta \leq \delta_1. \end{aligned} \tag{3.1}$$

In the literature this is called a constrained optimization problem. First we change the above maximization problem to a minimization one. Then setting $x = \beta$ and $y = \delta$, we have

$$\begin{aligned} & Min \;\; l_2(x, y) = -l_1(x, y) \\ & S.to: \\ & \quad g_1(x, y) = x_0 - x \leq 0, \\ & \quad g_2(x, y) = x - x_1 \leq 0, \\ & \quad g_3(x, y) = y_0 - y \leq 0, \\ & \quad g_4(x, y) = y - y_1 \leq 0. \end{aligned} \tag{3.2}$$

In the second step, using penalty function method [1], we will transform the resulted constrained minimization problem to a unconstrained problem. To do this, we need some appropriate penalty function. Following [1], we find that $\varphi(x) = \max(0, x)^2$ is a suitable choice for inequality constraints. Thus

$$\begin{aligned} & Min \;\; F(x, y) = l_2(x, y) + Cp(x, y) \\ & S.to: \\ & \quad x, y \in \mathbb{R}, \end{aligned} \tag{3.3}$$

in which $C > 0$ and $p(x, y)$ are called penalty parameter and function respectively and

$$p(x, y) = \varphi(g_1(x, y)) + \varphi(g_2(x, y)) + \varphi(g_3(x, y)) + \varphi(g_4(x, y)). \tag{3.4}$$

The above minimization problem is an unconstrained problem.

3.1. **Nelder-Mead.** The Nelder-Mead algorithm [15] is minimization algorithm which can be applied to minimize multivariate function. This algorithm is summarized below for solving a two dimensional problem:

(1) Let a, b and c be three initial guess of the solution with $F(a) < F(b) < F(c)$.
(2) If three points or their function values are sufficiently close to each other, then declare a to be the minimum and terminate the procedure.
(3) Otherwise, expecting that the minimum may be at the opposite side of the worst point cover the line $\overline{ab}$, take $e = m + 2(m - c)$, where $m = \frac{a+b}{2}$,
    - and if $F(e) < F(b)$, take e as the new c; otherwise take $r = \frac{m+e}{2} = 2m - c$,
    - and if $F(r) < F(c)$, take r as the new c; if $F(r) = F(b)$, take $s = \frac{c+m}{2}$,
    - and if $F(s) < F(c)$, take s as the new c; otherwise, give up the two points b, c and take m and $c_1 = \frac{a+c}{2}$ as the new b and c, reflecting our expectation that the minimum would be around a.
(4) Return to Step 1.

3.2. **PSO.** Many conventional techniques such as gradient-based search algorithms and various mathematical programming methods have been proposed to deal with optimization problems. However, these techniques have severe limitations in handling nonlinear, discontinuous or multi-modal functions and constraints because exact methods depend on gradient information and are more likely to get stuck at local optima. While, meta-heuristic algorithms such as Genetic Algorithms (GAs) [10] and Particle Swarm Optimization (PSO) [11], do not depend on gradient information. In this situation, meta-heuristic algorithms are still able to work satisfactorily.

Particle swarm optimization is another evolutionary computation technique developed by Eberhart and Kennedy [11, 8] in 1995, which was inspired by the social behavior of bird flocking and fish schooling. PSO has its roots in artificial life and social psychology, as well as in engineering and computer science. It utilizes a population of particles that fly through the problem hyperspace with given velocities. At each iteration, the velocities of the individual particles are stochastically adjusted according to the historical best position for the particle itself and the neighborhood best position. Both the particle best and the neighborhood best are derived according to a user defined fitness function. The movement of each particle naturally evolves to an optimal or near-optimal solution. In PSO algorithm, each individual possible solution can be modeled as a particle that moves through the problem hyperspace. The position of each particle is determined by the vector $x_i \in \mathbf{R}^n$ and its movement by the velocity of the particle $v_i \in \mathbf{R}^n$, as shown in (3.5)

$$x_i(t) = x_i(t-1) + v_i(t), \tag{3.5}$$

The information available for each individual is based on its own experience (the decisions that it has made so far and the success of each decision) and the knowledge of the performance of other individuals in its neighborhood. Since the relative importance of these two factors can vary from one decision to another, it is reasonable to apply random weights to each part, and therefore the velocity will be determined by

$$v_i(t) = wv_i(t-1) + c_1d_1(p_i - x_i(t-1)) + c_2d_2(p_g - x_i(t-1)), \tag{3.6}$$

where $w \in [0,1)$ is an inertia weight specifying how much of the particle's previous velocity is preserved, c_1, c_2 are two positive numbers and d_1, d_2 are two random numbers with uniform distribution in the range of $[0,1]$, p_i is the ith particle best position so far; and p_g is the best position found by the entire search space so far.

3.3. **GRSA.** In 2015, Beiranvand and Rokrok intruduced General Relativity Search Algorithm (GRSA) [3]. This is a new meta-heuristic optimization algorithm. GRSA inspired by General Relativity Theory. In GRSA, a population of particles (solution agents) is considered in a space free from all external non-gravitational fields and propel toward a position with least action. Step length and step direction for updating the agents are separately computed using particles velocity and geodesics, respectively. By inspiring these notions of General Relativity Theory, GRSA will evolve variables of an optimization problem toward the global optimal point. For detailed discussion of this algorithm see [3].

4. Result Analysis

In this section, we want to implement the above mentioned optimization algorithms to estimate β and δ jointly. To minimize the problem (3.3), we need to choose the penalty parameter and the upper and the lower bound for x and y. Assume that these values have been chosen as in the table 1. Thus we have the following minimization

Table 1. Parameter values

Name	x_0	x_1	y_0	y_1	C
Value	-10	10	10^{-4}	10	10^5

problem

$$\begin{aligned} &Min \quad l_2(x,y) = \ln y + \frac{x-2}{2N}\sum_{n=0}^{N-1} x_n + \frac{1}{2N}\sum_{n=0}^{N-1}\frac{z_n^2}{ye^{\frac{x-2}{2}x_n}} \\ &S.to: \\ &\qquad g_1(x,y) = -x - 10 \le 0, \\ &\qquad g_2(x,y) = x - 10 \le 0, \\ &\qquad g_3(x,y) = -y + 10^{-4} \le 0, \\ &\qquad g_4(x,y) = y - 10 \le 0. \end{aligned} \tag{4.1}$$

Using penalty function yields

$$\begin{aligned} &Min \quad F(x,y) = l_2(x,y) + 10^5 p(x,y) \\ &S.to: \\ &\qquad x, y \in \mathbb{R}, \end{aligned} \tag{4.2}$$

where $p(x, y) = \sum_{j=1}^{4} \varphi(g_j(x, y))$. For the share price of telecommunication company of Iran (TCI) from the 10^{th} of August 2008 to the 17^{th} of Jun 2015, we want to estimate β and δ. To evaluate the above objective function, we need to compute the natural logarithm of the share price and take the first difference of the logarithms to form z_n, the mean $\mu(X_n)$ is estimated using Lowess filter non parameterically.

First of all Nelder-Mead (NM) algorithm is implemented. To do this, we should pick some initial points. The results of the simulation of the problem (4.2) by Nelder-Mead algorithm are shown in the Table 2.

Table 2. Numerical results related to the NM algorithm

x_i	Initial points	Optimal solutions	Objective function value	Elapsed time
1	(-10,4)	(0.7731,1.4890)	-3.8994	0.358403
2	(-4,-3)	(0.7732,1.4888)	-3.8994	0.904574
3	(-1,9)	(0.7730,1.4897)	-3.8994	0.355850
4	(3,5)	(0.7730,1.4895)	-3.8994	0.318242
5	(6,-8)	(0.7732,1.4885)	-3.8994	0.399244

In the Table 2, the optimal solutions show the estimated values of β and δ respectively. The fourth column represents the minimum values of the objective function corresponding to the estimated values of β and δ. The fifth column shows the related elapsed times.

In another simulation, we apply the PSO algorithm. In this method, we need to determine the parameters of the algorithm which are the number of initial population or swarm M, the constants c_1, c_2 and the number of iteration Num_Itr. Suppose that these parameters are given in the Table 3. Running the PSO, we get the optimal

Table 3. PSO parameters

Name	M	w	c_1	c_2	Num_Itr
Value	100	0.8	2	2	100

solution $(0.7732, 1.4885)$, the optimal value of the objective function -3.8994 and the elapsed time 17.666011 seconds for this algorithm. The graph of the evaluation of the objective function per iteration is given in the Figure 1. Now we want to implement

Table 4. GRSA parameters

Name	M	$\mathrm{g}_{\max}$	$\mathrm{g}_{\min}$	Num_Itr
Value	100	0.999	0.011	100

GRSA. Assume that the GRSA's parameters and their values are given in the table 4, where $g_{\max}$ and $g_{\min}$ are geometry coefficients of spacetime with $g_{\max} + g_{\min} = 1$, $g_{\max}, g_{\min} \in [G_{\min}, G_{\max}]$, in which $G_{\max} = 1, G_{\min} = 0$ are maximum and minimum

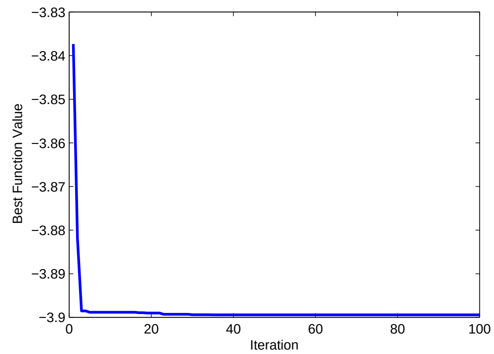

FIGURE 1. Convergence curve of PSO

limits of geometry coefficient, respectively, and M, Num_Itr are the number of initial populations or particles and maximum iteration respectively.

By running this algorithm, one can obtain the optimal solution $(0.7732, 1.4888)$, the optimal value of the objective function -3.8994 and the elapsed time 17.6495 for GRSA algorithm. Also the graph of the evaluation of the objective function per iteration is given in the Figure 2.

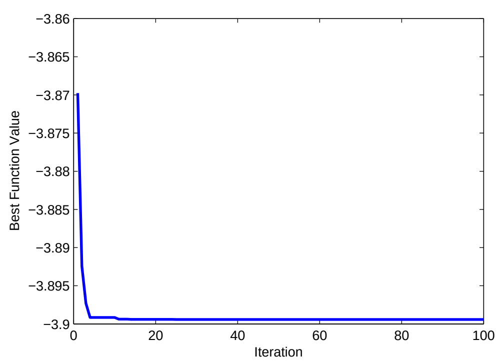

FIGURE 2. Convergence curve of GRSA

Now we want to test the estimation strategy used in this paper. In estimating β and δ jointly, we assumed that the relation

$$\frac{Z_n}{\sqrt{\delta^2 S^{\beta-2}}} \sim \mathcal{N}(0,1), \tag{4.3}$$

holds at least approximately, where $\mathcal{N}$ is unit normal probability distribution. Assume that, using the results of the above optimization problem, the estimated values are

$\hat{\beta} = 0.7732$ and $\hat{\delta} = 1.4888$. Let us consider

$$E_n = \frac{Z_n}{\sqrt{\hat{\delta}^2 S_n^{\hat{\beta}-2}}}. \tag{4.4}$$

These values will be compared graphically to a unit normal probability density function. If the two distributions are close, the estimation strategy will be good. In the Figure 3, the histogram of E_n is shown along with the graph of the unit normal density function. As it is clear they are close to each other and this proves the reliability of the method used in this paper.

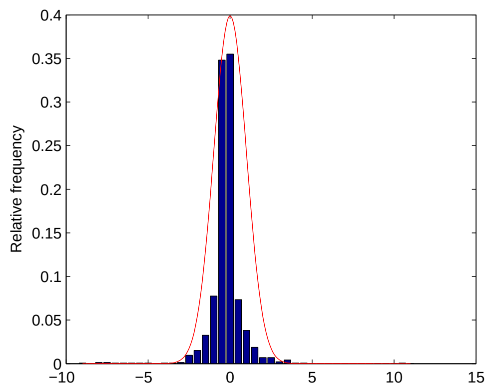

FIGURE 3. E_n and unit normal

It is seen, by the result of simulations, that $\hat{\beta} < 1$ which is in agreement with theoretical predict given in introduction. This fact proves that the share price of TCI has the inverse relationship with volatility.

To show the ease of access to the global optimal point of the proposed estimation formulation, three different optimization algorithms are tested. First of all, Nelder-Mead algorithm is applied which is a simple search algorithm and traps in local optimal points easily. Thereafter, PSO and GRSA are applied which are global search methods and find global optimal point or near-global optimal point of an optimization problem. Results of this comparative study show that even simple and conventional optimization algorithm like Nelder-Mead method solves the proposed estimation formulation satisfactorily and finds its global solution effectively.

5. Conclusion

In this paper, we considered a stochastic differential equation called constant elasticity of variance. The aim was to estimate the parameters of constant elasticity of variance as the model of share price using the time series data of telecommunication company of Iran. For this purpose, the density function of constant elasticity of variance was approximated by density function of the normal distribution. Using maximum likelihood function, we formulated the parameter estimation problem as an optimization problem. Finally, the optimal parameters have been obtained by solving the optimization problem using three minimization algorithms such as Nelder-Mead, PSO and GRSA. The results of simulations showed that the estimated elasticity $\hat{\beta} < 1$. This fact confirms that the share price of TCI has inverse relationship with volatility as theoretical results predict. As a direction for future research, one can consider the stochastic differential equation with stochastic volatility and jump to fit the time series data of share price.

References

[1] M. S. Bazaraa, H. D. Sherali, and C. M. Shetty, *Nonlinear Programming*, Third Edition, John Wiley & Sons, Inc., 2006.

[2] S. Beckers, *The constant elasticity of variance model and its implications for option pricing*, Journal of Finance, *35*(3) (1980), 661–673.

[3] H. Beiranvand and E. Rokrok, *General Relativity Search Algorithm: A Global Optimization Approach*, International Journal of Computational Intelligence and Application, *14*(3) (2015), 1550017 (29 pages).

[4] F. Black and M. Scholes, *The pricing of options and corporate liabilities*, The Journal of Political Economy, *81*(3) (1973), 637–654.

[5] A. Chockalingam and K. Muthuraman, *Pricing American options when asset prices jump*, Operations Research Letters, (2009) doi:10.1016/j.orl.2009.11.001.

[6] J. Cox, *Notes on option pricing I: Constant elasticity of variance diffusions*, Working paper, Stanford University, 1975 (reprinted in Journal of Portfolio Management, *22* (1996), 15–17).

[7] F. Delbaen and E. Shirakawa, *A note on option pricing for constant elasticity of variance model*, Financial Engineering and the Japanese Markets, *9*(2) (2002), 85–99.

[8] R. Eberhart and J. Kennedy, *A new optimizer using particle swarm theory*, in Proc. 6th Int. Symp. Micro Machine and Human Science (MHS), (1995), 39–43.

[9] D. C. Emanuel and J. D. MacBeth, *Further results on the constant elasticity of v ariance call option pricing model*, Journal of Financial and Quantitative Analysis, *17*(4) (1982), 533–554.

[10] J. H. Holland, *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology*, Control, and Artificial Intelligence(U Michigan Press, 1975).

[11] J. Kennedy and R. Eberhart, *Particle swarm optimization*, in Proc. IEEE Int. Conf. Neural Networks (1995), 1942–1948.

[12] G. W. Liu and L. J. Hong, *Revisit of stochastic mesh method for pricing American options*, Operations Research Letters, *37* (2009), 411–414.

[13] T. Marsh and E. Rosenfeld, *Stochastic processes for interest rates and equilibrium bond prices*, Journal of Finance, *38*(2) (1983), 635–650.

[14] A. Melino and S. Turnbull, *The pricing of foreign currency options*, Canadian Journal of Economics, *24* (1991), 251-281.

[15] J. A. Nelder and R. Mead, *A simplex method for function minimization*, Computer J., *7* (1965), 308–313.

[16] J. P. V. Nunes, *Pricing American options under the constant elasticity of variance model and subject to bankruptcy*, Journal of Financial and Quantitative Analysis, *44* (2009), 1231–1263.

[17] M. Schroder, *Computing the constant elasticity of variance option pricing formula*, J. of Finance, *44* (1989), 211–219.

[18] A. L. Tucker, D. R. Peterson, and E. Scott, *Tests of the Black-Scholes and constant elasticity of variance currency call option valuation models*, The Journal of Financial Research, *11*(3) (1988), 201–213.