IOSR Journal of Economics and Finance (IOSR-JEF)
e-ISSN: 2321-5933, p-ISSN: 2321-5925.
PP 92-96
www.iosrjournals.org

An Intelligent Approach for Option Pricing

Vijayalaxmi[1], C.S.Adiga[1], H.G.Joshi[2]
[1]*(Department of E&E, MIT, Manipal University, Manipal)*
[2]*(MIM, Manipal University, Manipal)*

Abstract: *Option pricing is one of the exigent and elementary problems of computational finance. The paper aims to determine the option price and time to exercise the option. In order to optimize the model, Ant Colony Optimization Technique (ACO) has been used here and complete simulation is done using MATLAB 2015a simulation environment. Emphasis is also given on comparing the proposed method with the other existing models and come up with an optimum model for option pricing which will benefit traders and risk managers to obtain the computed results very fast with high accuracies. The algorithm which uses ACO technique has three important steps. Initially it starts by injecting ants from the valuation date. This is considered to be the root node. It can acquire whichever path based on arbitrary behavior. Later, individual ants compute the payoff at each node based on a set expression. As soon as an ant finds a value, it updates the pheromone concreteness leading to the node. Finally updating the pheromone on the path makes it more attractive for other ants to explore in the neighboring areas.*
Keywords: *Financial derivatives, Option pricing, Ant Colony Optimization*

I. Introduction

In recent years options have turn out to be more and more significant in the world of finance and investments. They are popular with investors. There are two types of options namely call options and put options. A call option gives the holder 'the right to buy an asset by a certain date for a certain price.' A put option gives the holder 'the right to sell an asset by a certain date for a certain price.' Briefly, options give the holder 'the right' to do something. Important aspect is the holder does not have to exercise the right. Option pricing involves computationally intensive calculations and requires an efficient system to perform the task in a faster rate. In Financial Mathematics, usually the Black Scholes algorithm is used to price these contracts. In this model, an important variable is volatility. Volatility is a measure of the amount of fluctuation in the asset prices, a measure of randomness. It has a major impact on the value of the option. It describes the variation in the stock prices, interest rates and other factors around their means. In the Black-Scholes equation for pricing European options, volatility is assumed to be constant. This is a major drawback of this model. Estimating the present price of an option by predicting the fundamental asset prices is the general way to compute option price. Pricing options has been a demanding problem for a long time due to unpredictability in market conditions which gives rise to unpredictability in the option prices. Also the time when the options have to be exercised has to be determined to maximize the profits. As shown in the Fig 1, profit depends on the stock price, strike price, time to expire and option price. Estimating the option price would help the finance managers, financial institutions and even one who wants to know the option price before investing. Keeping all these factors in mind an effort is made to determine the option price using Binomial approach. This approach is derived from the Black Scholes equation.

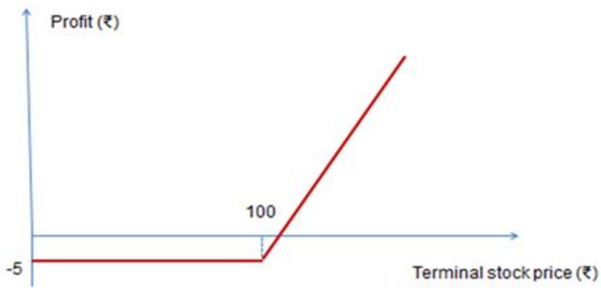

Fig.1 Dependency of profit on stock price

In the next subsequent sections methodology and results are discussed. Final section comprises conclusions and future scope of this work

II. Methodology

The Black-Scholes Option Pricing Model is an approach for calculating the value of a stock option. The basic idea behind the Black-Scholes model is that the price of an option is determined implicitly by the price of the underlying stock. The Black-Scholes model is a mathematical model based on the concept that prices of stock track a stochastic procedure, which means that the price of a stock in time‘t+1’ is independent from the price in time‘t’. This is also considered as random walk. Assuming that stock prices follow random walk, it is very much essential to involve mathematics and statistics in the calculations. Binomial Lattice option pricing model works backwards from exercise time to the current price. This uses a predetermined lattice of asset prices and times. It includes binomial trees as shown in Fig 2. In the Binomial model, it is required to divide the time between the present date and date of expiry into certain number of steps. Then principal asset price is determined for each of the time steps. Each of these asset price computation, results in a node in the binomial tree. The assessment of the binomial tree is iterative. Once all the nodes in the tree are created, it is essential to work backwards from date of expiry towards the source node to find the best node. At each step it is obligatory to compute the payoff at each node as shown in the figure 2. The option price at the assessment date is determined by pricing option at all the in-between nodes between expiration date and the valuation date. This process is called discounting. This involves moving backwards in time from the expiration date to the valuation date. Computing option value using this iterative method involves three steps. They are price tree generation, computing price at each leaf node, and iterative computation of option values at earlier nodes using a discounting factor. The value at the root node is the value of the option.

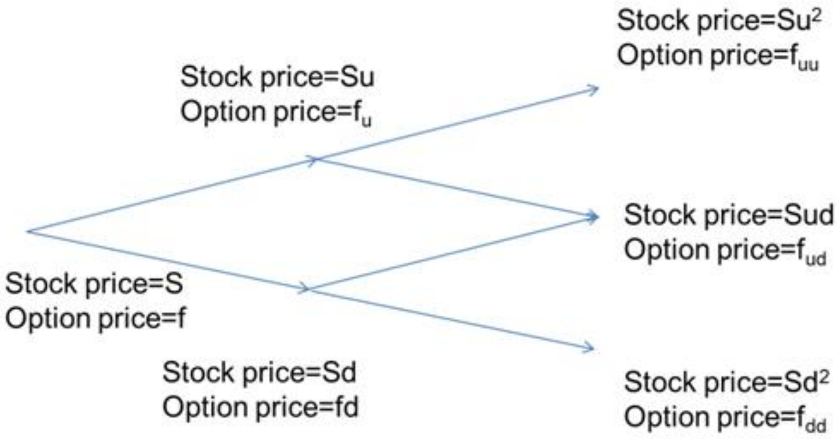

Figure1. Binomial Tree

Ant Colony Optimization, which is one of the Swarm intelligence techniques is used to optimize the model. Steps involved in pricing the option value are given below. It involves generating stock and option matrix, calculating the value of leaf and other nodes in option binomial tree and searching for optimal solution.

begin
define the input and output signals
define the matrices
initialize the input and output signals
initialize the matrices
generating stock and option matrix
begin
calculate the value of leaf nodes in option binomial option tree
calculate the value of other nodes in option binomial option tree
end
search for optimal solution in the first subset of the solution set
search for optimal solution in the second subset of the solution set
search in the rest of the solution subsets
check if the optimum solution generated is within the specified limits of the solution space
If optimum solution generated is within limits, generate optimum time
End

III. Results

```
stockM =

    9.0000   11.6701         0         0         0
         0         0         0         0         0
         0         0         0         0         0
         0         0         0         0         0
         0         0         0         0         0

stockM =

    9.0000   11.6701         0         0         0
         0    6.9408         0         0         0
         0         0         0         0         0
         0         0         0         0         0
         0         0         0         0         0

stockM =

    9.0000   11.6701   15.1324         0         0
         0    6.9408         0         0         0
         0         0         0         0         0
         0         0         0         0         0
         0         0         0         0         0
```

```
stockM =

    9.0000   11.6701   15.1324         0         0
         0    6.9408    9.0000         0         0
         0         0         0         0         0
         0         0         0         0         0
         0         0         0         0         0

stockM =

    9.0000   11.6701   15.1324         0         0
         0    6.9408    9.0000         0         0
         0         0    5.3527         0         0
         0         0         0         0         0
         0         0         0         0         0

stockM =

    9.0000   11.6701   15.1324   19.6219         0
         0    6.9408    9.0000         0         0
         0         0    5.3527         0         0
         0         0         0         0         0
         0         0         0         0         0
```

```
stockM =

    9.0000   11.6701   15.1324   19.6219         0
         0    6.9408    9.0000   11.6701         0
         0         0    5.3527    6.9408         0
         0         0         0         0         0
         0         0         0         0         0

stockM =

    9.0000   11.6701   15.1324   19.6219         0
         0    6.9408    9.0000   11.6701         0
         0         0    5.3527    6.9408         0
         0         0         0    4.1280         0
         0         0         0         0         0

stockM =

    9.0000   11.6701   15.1324   19.6219   25.4434
         0    6.9408    9.0000   11.6701         0
         0         0    5.3527    6.9408         0
         0         0         0    4.1280         0
         0         0         0         0         0
```

```
stockM =

    9.0000   11.6701   15.1324   19.6219   25.4434
         0    6.9408    9.0000   11.6701   15.1324
         0         0    5.3527    6.9408    9.0000
         0         0         0    4.1280         0
         0         0         0         0         0

stockM =

    9.0000   11.6701   15.1324   19.6219   25.4434
         0    6.9408    9.0000   11.6701   15.1324
         0         0    5.3527    6.9408    9.0000
         0         0         0    4.1280    5.3527
         0         0         0         0         0

stockM =

    9.0000   11.6701   15.1324   19.6219   25.4434
         0    6.9408    9.0000   11.6701   15.1324
         0         0    5.3527    6.9408    9.0000
         0         0         0    4.1280    5.3527
         0         0         0         0    3.1835
```

Fig. 2 Developed stock matrices

```
optionM =

         0         0         0         0         0
         0         0         0         0         0
         0         0         0         0    1.0000
         0         0         0         0    4.6473
         0         0         0         0         0

optionM =

         0         0         0         0         0
         0         0         0         0         0
         0         0         0         0    1.0000
         0         0         0         0    4.6473
         0         0         0         0    6.8165

optionM =

         0         0         0         0         0
         0         0         0         0         0
         0         0         0         0    1.0000
         0         0         0    5.8720    4.6473
         0         0         0         0    6.8165
```

```
optionM =

         0         0         0         0         0
         0         0         0    0.4560         0
         0         0    4.6473    3.0592    1.0000
         0         0         0    5.8720    4.6473
         0         0         0         0    6.8165

optionM =

         0         0         0         0         0
         0         0    1.6230    0.4560         0
         0         0    4.6473    3.0592    1.0000
         0         0         0    5.8720    4.6473
         0         0         0         0    6.8165

optionM =

         0         0    0.2079         0         0
         0         0    1.6230    0.4560         0
         0         0    4.6473    3.0592    1.0000
         0         0         0    5.8720    4.6473
         0         0         0         0    6.8165
```

```
optionM =

         0         0         0         0         0
         0         0         0         0         0
         0         0         0    3.0592    1.0000
         0         0         0    5.8720    4.6473
         0         0         0         0    6.8165


optionM =

         0         0         0         0         0
         0         0         0    0.4560         0
         0         0         0    3.0592    1.0000
         0         0         0    5.8720    4.6473
         0         0         0         0    6.8165


optionM =

         0         0         0         0         0
         0         0         0    0.4560         0
         0         0         0    3.0592    1.0000
         0         0         0    5.8720    4.6473
         0         0         0         0    6.8165
```

```
optionM =

         0         0    0.2079         0         0
         0    3.0592    1.6230    0.4560         0
         0         0    4.6473    3.0592    1.0000
         0         0         0    5.8720    4.6473
         0         0         0         0    6.8165


optionM =

         0    0.8441    0.2079         0         0
         0    3.0592    1.6230    0.4560         0
         0         0    4.6473    3.0592    1.0000
         0         0         0    5.8720    4.6473
         0         0         0         0    6.8165


optionM =

    1.8171    0.8441    0.2079         0         0
         0    3.0592    1.6230    0.4560         0
         0         0    4.6473    3.0592    1.0000
         0         0         0    5.8720    4.6473
         0         0         0         0    6.8165
```

Fig. 3 Developed Option matrices

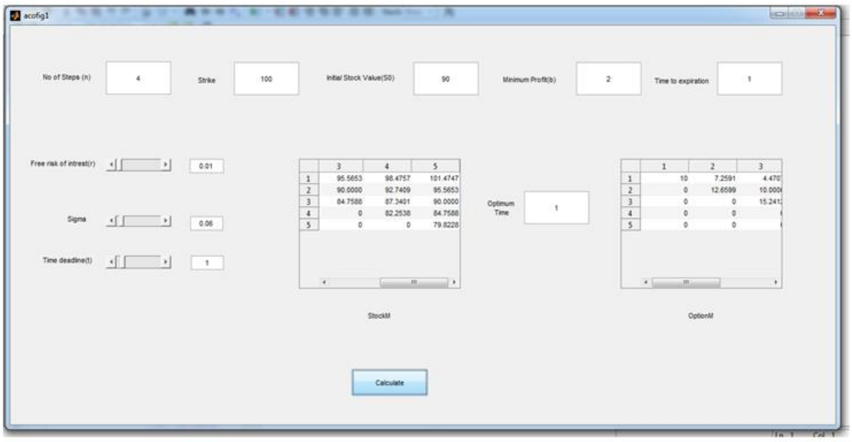

Fig. 4 Graphical User Interface to determine option price

In this section screen shots of different stock and option matrices are shown in Fig 3 and 4 respectively. The root node is the actual option price. By increasing the number of steps further accuracy can be achieved. This algorithm is generic in nature and can be used for any data. In order to validate the results Graphical User Interface (GUI) is developed for the Black Scholes model and screen shot of the same is shown in Figs.5 and 6.

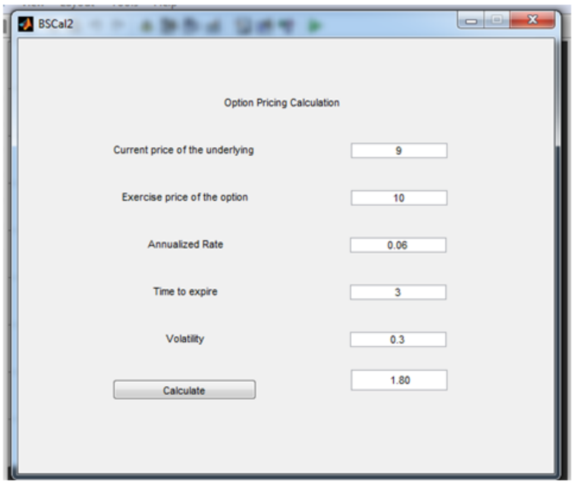

Fig. 6. GUI to validate the results

It is very much clear from the results that the developed Intelligent Binomial model for option pricing has converged with accurate results.

IV. Conclusions

The goal of this work was to price American Put Options using Binomial Model, optimize the model using ACO Technique to obtain optimum solution under user defined constraints. A range of computational approaches to price options demand computing a larger number of pre-set nodes which symbolize the probable price movements of the fundamental asset of an option. The motivation for this work is to abolish the need for computing all the potential nodes in finding the best possible node to exercise an option. The idea is to discover the solution space using ants that would recognize a accurate trail towards the most favorable solution. This would be the right node to exercise the option. The volatility in this work, is not constant here .A predictive neural network model, based on the past data gives the volatility, which in turn acts as input to this intelligent model.

References

[1] Ashutosh Nigam and Yogesh K. Agarwal.,Journal of Applied Operational Research (2013) 5(3), 96–104 David Martens, Tony Van Gestel, Manu De Backer .,Raf Haesen,Jan Vanthienen, Bart Baesens.,Credit Rating Prediction Using Ant Colony Optimization,2008

[2] K. Forqandoost Haqiqi ,T. Kazemi.,Ant colony optimization approach to portfolio optimization.,2012 International Conference on Economics, Business and Marketing Management

[3] Kambiz Forqandoost Haqiqi and Tohid Kazemi,Ant Colony Optimization Approach to Portfolio Optimization – A Lingo Companion, International Journal of Trade, Economics and Finance, Vol. 3, No. 2, April 2012

[4] Sameer Kumar, Ruppa K. Thulasiram, Parimala Thulasiraman.,A Bioinspired Algorithm to Price Options, ACM,2008

[5] T. Karthikeyan, J. Mohana Sundaram., A Study on Ant Colony Optimization with Association Rule, International Journal of Advanced Research in Computer Science and Software Engineering, 2012

[6] Wei-Mou Hung1 and Wei-Chiang Hong,Journal of Control and Cybernetics vol. 38 (2009) No. 3 Application of SVR with improved ant colony optimization algorithms in exchange rate forecasting

[7] Yannis Marinakis, Magdalene Marinaki, Michael Doumpos, Constantin Zopounidis., Ant colony and particle swarm optimization for financial classification problems, Expert system with applications, 2009, Elsevier

[8] Yeshwanth Udayshankar, Sameer Kumar, Girish K. Jha, Ruppa K. Thulasiram and Parimala Thulasiraman, An Aggregated Ant Colony Optimization Approach for Pricing Options,2009,IEEE

[9] Younes Elahi and Mohd Ismail Abd Aziz., The Ant colony Pseudocode for Mean-Variance- CVaR model of Multi-Portfolio Optimization

[10] Parimala Thulasiraman. A bio inspired algorithm to price options, proceedings of the 2008 C3S2E conference on C3S2E 2008.

[11] Sameer Kumar, Ant colony optimization to price exotic options,2009 IEEE congress on Evolutionary Computation 2009

[12] Amster P.A parabolic problem arising in Financial mathematics, Non linear Analysis, Real world Applications 201004