University of Piraeus
School of Finance and Statistics
Department of Banking and Financial Management

M.Sc. in Banking and Finance with domain expertise in Finance and Investments

Thesis Theme

Option Valuation with a Modified Tree Method

ANDREAS BOUZOUKAS
MXAN 1917

SUPERVISOR: **ASSISTANT PROFESSOR NIKOLAOS ENGLEZOS**

COMMITTEE: **PROFESSOR NIKOLAOS KOUROGENIS**
ASSISTANT PROFESSOR NIKOLAOS ENGLEZOS
ASSISTANT PROFESSOR MICHAIL ANTHROPELOS

(Piraeus, August 2021)

Περίληψη

Η παρούσα διπλωματική εργασία αναφέρεται στην «Τροποποιημένη προσέγγιση πλέγματος για την τιμολόγηση δικαιωμάτων προαίρεσης» του Tian (1993). Για να προχωρήσουμε στην τιμολόγηση και τη σύγκριση, χρησιμοποιούμε το διωνυμικό μοντέλο του Cox, Ross & Rubinstein (1979), το τριωνυμικό μοντέλο του Boyle (1986), το τροποποιημένο διωνυμικό μοντέλο του Tian (1993) και τα δύο τροποποιημένα τριωνυμικά μοντέλα του Tian (1993).

Πρώτον, παρατίθεται το θεωρητικό μέρος, το οποίο απαιτείται για την ολοκλήρωση της εμπειρικής μελέτης. Ως εκ τούτου, παρουσιάζεται μια σύντομη ιστορική ανασκόπηση σχετικά με την ανάγκη εύρεσης ενός πιο αποτελεσματικού μοντέλου τιμολόγησης δικαιωμάτων προαίρεσης και ανάλυσης των προβλημάτων που προκύπτουν χρησιμοποιώντας σταθερή μεταβλητότητα σε υφιστάμενα μοντέλα. Στην συνέχεια περιγράφονται το μοντέλα πλέγματος των Cox, Ross & Rubinstein (1979), του Boyle (1986) καθώς και του Tian (1993) για την αποτίμηση Ευρωπαϊκών Δικαιωμάτων Αγοράς και Πώλησης.

Ακολουθεί η πραγματοποίηση της ανάλυσης αριθμητικής ακρίβειας και της εμπειρικής έρευνας, στην οποία οι επιλογές προσαρμόζονται ανάλογα με το υποκείμενο περιουσιακό στοιχείο (δηλαδή τη μετοχή της Apple εντός συγκεκριμένου χρονικού πλαισίου). Στη συνέχεια, εκτιμώντας τις παραμέτρους των μοντέλων και ελέγχοντας την προβλεπτική τους ικανότητα τα συγκρίνουμε μεταξύ τους.

<u>ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ</u>

Δικαίωμα Προαίρεσης, Μοντέλο CRR, Μοντέλο Boyle, Διωνυμικό Μοντέλο Tian, Πρώτο Τριωνυμικό Μοντέλο Tian, Δεύτερο Τριωνυμικό Μοντέλο Tian, Αριθμητική Ανάλυση, Levenberg-Marquardt Αλγόριθμος, Εκτίμηση Παραμέτρων, Προβλεπτική Ικανότητα

Abstract

This thesis refers to “A Modified Lattice Approach to Option Pricing” of Tian (1993). In order pricing and comparison to be succeeded, we use the binomial model of Cox-Ross-Rubinstein (1979), the Boyle Trinomial Model (1986) and the Tian Modified Binomial and Trinomial Models (1993).

Firstly, we cite the theoretical part, which is required for the completion of the empirical study. Therefore, we present a brief historical review towards the need to find a more efficient option pricing model and analyze the problems that arise by using constant volatility in pre-existing models. We proceed with the theoretical analysis of the models starting from the binomial valuation model, continuing with the trinomial tree model, and reaching at the main topic, the construction of the modified binomial tree and the two modified trinomial trees for the valuation of European call and put options.

Then carry out numerical accuracy analysis and empirical research in turn, in which options are adjusted according to the underlying asset (that is, Apple's within a certain time frame). Then, estimate the price of the parametric model and estimate the performance of the out-of-sample model. Finally, compare the models in order to get the most effective model.

KEY WORDS

Option, CRR Model, Boyle Model, Tian Binomial Model (Tian Bin), Tian Trinomial 1 Model (Tian Trin1), Tian Trinomial 2 Model (Tian Trin2), Numerical Analysis, Levenberg-Marquardt Algorithm, Parameter Estimation, Forecasting Ability

Table of Contents

CHAPTER 1

INTRODUCTION

1.1 Problem description

There are a few contentions supporting why individuals are so excited about economy and finance. One of them lies in the case that market trading is a classic illustration of complex framework, where seemingly random fluctuations of market prices are the outcome of various origins, such as non-linear reactions from dealers and brokers with increased mutual related operations settled inside in a period developing and in a to some degree capricious way area. This adversity in predicting the behavior of trade values has indeed been disputed to be an essential property of “efficient” markets, where is stated that any undeniable unsurprising chance ought to quickly be removed by the response of the market itself. Up to the minute, math and trade theories of finance are developed according to the point of view of stochastic models where specialists can reconsider their judging consistently on schedule as a reaction to various external and individual stimulus. It is the intricacy of the agent anticipated probabilities and intercommunication that stipulate the critical difficulty in the study of economy.

Consequently, with regards to choose the amount to spend on an option agreement we manage the primary concern at the assignment of appraising options, which become substantially more intricate when we seek to predict the future conceivable cost of the option. This is feasible, if the relative likelihoods of prices going vertical, descending, or continuing as before are known. Each financial backer's longing is to make benefit on whichever sum they put in the stock market and accordingly the necessity for a reliable structured strategy that gives a nearby approximation to the market costs.

The valuation of an option has proved to be particularly important for the analysis of many dimensions of corporate finance and investment practices. The volume and variety of options traded in global exchange market has expanded considerably following the approach developed by Black & Scholes (1973).

Research reveal that the theory of options valuation is applicable to pretty much every field in finance. For instance, all corporate finance securities can be deciphered as portfolios consisting of buying and selling options over company assets.

The simplest option pricing problem we can trace out there is the “European call option”, in which case consider that an investor desires to acquire a share, at a specified period in future $t=T$ from current moment ($t=0$), at a predefined strike price K. If the worth of the share at $t=T$, $S(T)$ is higher than K, the investor will proceed with exercising his option. His gain, if he immediately resells the share, will be the difference between $S(T)-K$.

Contrariwise, if $S(T)<K$, the investor will not exercise his option. Correspondingly, a European put option provides the holder with the alternative to sell his share at time $t=T$ at a fixed price K. These types of potentialities given to the investor by a financial institute are options and have clearly themselves a price.

In some respects, the obstacle of options is the main component of the general obstacle of appraising the threats connected with market deals and mostly appraising the risks linked with human activities. Of course, stock options can be considered as insurance premiums opposed to threats prompted by stock market incertitude. When you insure a vehicle, you are acquiring an insurance company’s put option, for example, an option to sell a car at a certain price. On the off chance that you have never had a collision (you disburse the premium and do not charge anything), this option is worthless. Nonetheless, if your vehicle is obliterated, you reserve the option to leave the leftovers of the vehicle in the insurance company which is contractually obligated to purchase the vehicle and disburse the insurance total amount. In consequence, the primary purpose of options is to permit investors to handle partly the impact of the market changes on their investment portfolios. For a particular cost, the stakeholder of an option confines the loss by having almost no limit on his earnings. Option sellers who anticipate trivial market changes might have the opportunity to collect additional premiums. To put it plainly, the options address the issues of both conservative individuals and gamblers. Moreover, the evaluating of more tangled financial underlying assets is fundamentally founded on similar charges.

Therefore, the valuation of such securities poses demanding problems, as it is a complex set of embedded rights whose performance depends on two or more status variables. Such problems become more acute when the additional issue of early practice is raised.

A beneficial and quite well-received process for valuing an option, either simple or complex, is the construction of a binomial or trinomial tree. This is a schema that illustrates dissimilar feasible paths that the price of the underlying security may follow during the existence of the option.

The basic premise is that the price of the underlying title follows a "random walk". At each time step the value of the stock has a specific probability of an upward trend, a specific probability of a downward trend and the probability of remaining stable. As the time step becomes smaller and as the quantity of steps expands, the worth of the option resulting in the binomial or trinomial model is closer to that obtained from the Black, Scholes & Merton (1973) model.

1.2 Historical Overview

Options have been traded for many years but continued being quite vague financial instruments until 1973 (Appolloni and Ligori, 2014). The year 1973 marked a new era in options trading, which in a relatively short time expanded into the US stock markets. Algorithms based on tree diagram approaches have been studied since 1979 (Cox, Ross and Rubinstein, 1979), and in fact seem to be very effective, both in terms of efficiency and speed of execution, using a reverse induction in their calculations (Muroi and Yamada, 2013).

Pricing options have drawn the attention of a significant number of scholars in order to dedicate their time for researching option pricing by utilization of tree processes. Black and Scholes hype the prominent pricing option procedure. Nevertheless, the information on math of this model is excessively profound and hard to comprehend, and it is not broadly known by the overall stakeholders. Merton posted an article with headline "theory of rational option pricing" , in order to accomplish a leap forward in the area of options pricing they built a model known as "Black-Scholes" formula or "Black-Scholes-Merton". This model was significantly innovative because the price of the option was determined by known parameters.

1.2.1 Binomial Historic Overview

Cox-Ross-Rubinstein (1979) developed a popular and broadly utilized discrete time model for option valuation, the binomial approach. It is related to the composition of the binomial tree that models the movement of the price of the underlying asset. This model does not include speculative opportunities.

Jarrow and Rudd (1983) extend the classical CRR model. They developed a binomial model with time-dependent parameters that are equal to all moments of the tree valuation and increases with the corresponding snapshots of the increments of the Itô valuation process.

Hull and White (1990) propose an adjustment of the explicit finite difference technique for the valuation of safety derivatives. In this paper, as more limited time spans are examined, the estimated upsides of the safety derivative converge to the answer of the underlying differential equation. It is feasible to be utilized to appraise any security derivative contingent on one state variable and can be extended to address more derivative valuation issues in which cases there are many state factors. Via delineation, this article illustrates the methodology using bond valuation and bond options written in two distinctive loan rate cost systems.

The article by Kunitomo and Ikeda (1992) gives an overall valuation technique to European options whose installment is restricted by arched cutoff points authoritatively set in the underlying asset evaluating measure following Brown's geometric motion. Their result is depended on the speculation of Levy's equation for Brownian movement by Anderson in successive examination. It gives the unequivocal equation of likelihood that Brown's geometric motion arrives at a space in the termination date without hitting either the lower or upper arched limits. Albeit the overall valuation equations for limit rights are communicated as endless series in the overall case, their mathematical examination shows that the series union is quick. Their results contain the classes of options with a lower limit than Merton's (1973) for the Goldman, Sossin and Gatto trail-dependent options(1979) and for certain corporate assets.

Leisen and Reimer (1996) proposed that the order of speed convergence should be considered as a measure for European market options. This article examines, at first, the obstacle of defining the sequence of convergence in the valuation of US put options. After that, it focuses on analytical examination of the extrapolation and control of the "Variate" method in order to boost convergence, and furthermore to justify their unforeseen obstacles. Thus, the research discloses the demand for smoothly converging models so as to make the opening inaccuracies smaller.

The article by Broadie and Detemple (1996) the lower and maximum cutoff values of American call and put options composed on assets that provide income. It gives two value draws near, one dependent on the limit (called LBA) and one dependent on the two thresholds (called LUBA). The LUBA approach has a normal exactness tantamount to a 1,000-step binomial tree with a computational speed practically identical to a 50-step binomial tree. It likewise presents an amendment of the binomial strategy (called BBSR), which is exceptionally easy to carry out and performs amazingly well. It additionally leads a cautious enormous scope assessment of numerous new techniques at ascertaining American options values.

Heston and Zhou (1997) describe the speed of convergence of a multi-level discrete-time option. They present that the convergence rate is subject to the smoothness of payment function provided by the option and is far less than is typically thought, because of the payment. So as to spice up accuracy, they suggest two straightforward approaches, a modification of the discrete time resolution before expiration, and also the smoothing of the payment process that yields answers that converge to the corresponding continuous cut-off date at the foremost conceivable rate appreciated by smooth payment operations. Additionally, they suggest a procedure that steadily infers multinational models by combining moments of a standard distribution.

P. Boyle (1998)'s article is an extension of the CRR model within the valuation of options. This approach includes an expansion of the lattice binomial approach developed by Cox, Ross, and Rubinstein to estimate options price on one asset. The primary purpose of the study was that payment by options depends on over one variable. Specifically, it examines the option could be a function of two underlying state variables although it’s possible to expand the procedure to situations involving a bigger number of state variables.

Kou (2003)'s article argues that an option subject to a barrier may be a derivative contract that's activated or extinguished when the price of the underlying item exceeds a selected level. Most models assume continuous barrier monitoring. Nevertheless, practically, in most cases the market barrier options traded are discretely monitored. In contrast with their continuous bonds that haven't any closed form solution, whereas pricing is additionally difficult. This study extends a Broadie, Glasserman, and Kou (1997) approach to barrier discretion covering most cases and providing an easier proof. The specific methods utilized are produced through sequential analysis by Siegmund, Yuh (1982) and Siegmund (1985).

Walsh (2003)'s article examines the definite convergence of the binomial tree. It is gleaned that the figure is first class. Tracks down the specific constants and shows that it is feasible to change the Richardson diversion to get a technique for request of three parts. Furthermore, he accompanies the clarification that the delta utilized in the hedge converges at indistinguishable rate. This is mainly investigated by integrating the trees into the Black-Scholes model through the integration of Skorokhod. We notice that this strategy applies in considerably more broad cases.

The article by Francine Diener and Marc Diener (2004) argues that the value of a European simple option calculated on a binomial tree converges to the value of Black & Scholes when the time-step inclines to zero. In addition, it was spotted that this convergence is of the sequence of 1 / n in conventional approaches and that it is oscillating. Moreover, the article calculates this oscillating functioning utilizing asymptotically Laplace completions, explicitly providing the first conditions asymptotic. Therefore, it shows the lack of asymptotic extension in the ordinary perceptions but that the convergence rate is indeed of the order of 1 / n in the case of ordinary binomial procedures once the number-two term disappears. The succeeding term is type $\overline{C}_2(n)$ / n with $C_2(n)$ some precise finite function of n, which has no restriction when n inclines to infinity.

The article by Chang and Palmer (2006) considers an overall category of binomial models with a supplemental parameter λ. It shows that in the circumstance of a European market option the binomial value converges to the Black & Scholes value with a factor of 1 / n and most crucially, gives a formula for the factor of 1 / n in the error extension. This allows us to demonstrate that convergence is smooth in a new central binomial model he proposes by making specific options for λ.

Joshi (2007's) article presents another group of binomial trees as ways to deal with the Black-Scholes model. For this class of trees, the presence of complete asymptotic expansions at the costs of European straightforward options is demonstrated and the initial three conditions are expressly estimated. In other explicit cases, a tree with third sequence convergence is fabricated and the notion of Leisen and Reimer (1996) demonstrates that their tree has secondary convergence.

Author	Title	Year
Cox, Ross, & Rubinstein	Option pricing: A simplified approach	1979
Jarrow & Rudd	Option pricing	1983
Hull & White	Valuing derivative securities using the explicit finite difference method	1990
Kunitomo & Ikeda	Pricing options with curved boundaries	1992
Leisen & Reimer	Binomial models for option valuation: Examining and improving convergence	1996
Broadie & Detemple	American option valuation: new bounds, approximations, and a comparison of existing methods	1996
Heston & Zhou	On rate of convergence of discrete-time contingent claims	1997
Boyle	A lattice framework for option pricing with two state variables	1998
Kou	On pricing of discrete barrier options	2003
Walsh	The rate of convergence of the binomial tree scheme	2003
Francine Diener & Marc Diener	Asymptotic of the Price Oscillations of a European Call Option in a Tree Model-Francine Mathematical Finance	2004
Chang & Palmer	Smooth convergence in the binomial model	2006
Joshi	Achieving Higher Order Convergence for the Prices of European Options in Binomial Trees	2007

Figure 1.1 Binomial Historic Overview

1.2.2 Trinomial Historic Overview

Three-jump models are used before within the literature to explicate option valuation issues. Stapleton and Subrahmanyan (1984) talked about a three-jump approach, but they failed to investigate its numerical accuracy or find out a way to study the attitude of jump probabilities. Parkinson (1977) utilized a three-jump procedure to appraise the American put option, but his method seems hard to be generalized to circumstances entailing over one state factor. Brennan and Schwartz (1978) associated the coefficients of a modified Black & Scholes formula to the possibilities of a three-jump procedure.

Boyle (1986) made the CRR philosophy one stride further and propounded a trinomial option valuing model, known as a three-jump model, where the stock value can move towards 3 bearings, upwards, downwards, or stay unaltered for a given time frame of period. A whole arrangement of trinomial cross section boundaries was acknowledged, including three probabilities and three leaps. Besides, he looked at the mathematical exactness of his model immediately in comparison to the CRR binomial model.

In a later article Boyle (1988) presented how a five-jump, three-dimensional lattice is elaborated to appraise options on two underlying securities. The strategy accustomed calculate the jump probabilities within the lattice is outlined, and a procedure for choosing the jump amplitudes is explained. It’s supposed that the joint density of the 2 underlying assets may be a bivariate lognormal distribution. When we apply the risk-neutral valuation process, it is obliquely stated that the two assets earn the riskless rate. A five-point jump procedure can be built to satisfy the varied requirements and can be utilized to generate a two-x one-dimensional lattice appropriate for valuing the options we desire.

E. Derman, I. Kani and N. Chriss (1996) demonstrated how to fabricate implied trinomial tree approaches of the volatility grin. Trinomial trees have naturally a bigger number of factors than binomial trees. They utilized these extra factors to helpfully pick the "state space" of all hub values in the trinomial tree and let just the conversion probabilities be restrained by market options values. This opportunity of state space delivers an adaptability that is here and there worthwhile in coordinating with trees to grins.

S. Crepey (2003) showed an inventive trinomial tree execution of the technique, in which the specific angle can be determined to the detriment of valuing the options and tackling one Fokker–Planck condition. Accordingly, the precision of the first technique is retained however the computational time is diminished. Typically, the calibration time decline is an attainable errand. The utilization of parallelism permits one to acquire a further factor. Also, he extends this procedure to the American calibration issue.

Ahn and Song (2007) inspected the convergence of the trinomial tree approach and demonstrate the assembly and exactness of this model. They exhibited a trinomial tree technique for valuing European/American options and fostered that it is identical to a specific express contrast strategy. By temperance of thickness arrangements, they demonstrated the uniform convergence of the expounded trinomial tree strategy for estimating European/American options. From their mathematical reproductions, the trinomial tree strategy with N/2 time-steps shows the equivalent or indistinguishable exactness contrasted with the binomial tree technique with N time-steps. As far as effectiveness, the computational expenses for the trinomial tree strategy with N/2 time-steps is achievable to be diminished, for example, at around 33 up to 34% mitigation for large N, contrasted with the corresponding for the binomial tree technique with N time-steps.

Yuen and Yang (2010) introduced a quick and simple tree structure to evaluate simple and exotic options in MRSM - Markov Regime Switching Model with multi-regime. They altered Boyle's trinomial tree structure by controlling the risk neutral probability measure in various system states to affirm that the tree model can adjust to the information of every single distinctive system and simultaneously keeping up with its joining tree structure.

Xiong (2012) propounded a trinomial pricing option process depended on Bayesian Markov Chain Monte Carlo Approach which ordered the old style binomial tree approach, the traditional trinomial tree approach, the BS approach, and the warrant value by utilizing the genuine information of the Chinese warrant market. The outcome demonstrates that the value deviation of the trinomial tree estimating alternative model depended on Bayesian MCMC technique is considerably more less than any other models, in spite of the fact that they all underestimate the market cost.

Han (2013) inspected the trinomial tree model to value options for explicit cases in mathematical techniques and wound up to the accompanying significant outcomes: contrasted to the binomial model, the trinomial tree model can approximate in a superior manner to the continuous distribution of the fundamental asset value movements with more states and has a more prominent exactness.

Author	Title	Year
Boyle	Option valuation using a three jump process	1986
Boyle	A lattice framework for option pricing with two state variables	1988
Derman, Kani & Chriss	Implied trinomial trees of the volatility smile	1996
Crepey	Calibration of the local volatility in a trinomial tree using Tikhonov regularization	2003
Ahn & Song	Convergence of the trinomial tree method for pricing European/American options	2007
Yuen & Yang	Option pricing with regime switching by trinomial tree method	2010
Xiong	A trinomial option pricing model based on Bayesian Markov chain Monte Carlo method	2012
Han	Pricing American options based on trinomial tree model	2013

Figure 1.2 Trinomial Historic Overview

1.3 Thesis Description

The purpose of this dissertation is to present the modified lattice approach to option pricing that was elaborated by Yisong Tian in 1993. More explicitly, Tian fostered a modified way to deal with the choice of cross section boundaries including probabilities and jumps. An overall methodology can be applied to any multi-dimensional cross section approach. It is perceived that the fundamental conditions for a cross section model to join to the Black-Scholes model don't give an exceptional answer for the grid boundaries. Extra restriction(s) on the grid boundaries is (are) required. In the modified methodology the determination of cross section boundaries is made to such an extent that the exactness of estimation is improved. The methodology is then applied to modify both the CRR binomial model and the Boyle trinomial model.

In Chapter 2, we will first give a general introduction to the options. Next, we will analyze the binomial tree of Cox, Ross and Rubinstein (1979), which is a two-step method through the reproduction and valuation of the portfolio in a risk-neutral environment. Furthermore, the trinomial valuation model will be built as a two-step binomial tree, in which the price of the underlying stock moves up and down, but can stay at the same price, making it more flexible and accessible to real market data.

In Chapter 3, we will present the modifications implied in both CRR binomial model and Boyle trinomial model. Hence, the Tian Bin model varies from the CRR binomial model in the accompanying two angles. To start with, in the CRR model the difference of the stock value is just right in the breaking point as the time step h approaches zero. In the Tian Bin model, both the mean and variance are right for some random h. Second, rather than picking the right third moment (along these lines right skewness), Cox, Ross, and Rubinstein (1979) chose the accompanying condition for demonstrating effortlessness. The modified trinomial models are instinctively more engaging than the Boyle model. Review that Boyle (1986) tracked down the "ideal" arrangement through an experimentation test. A precise strategy for taking care of the issue is introduced here. The Tian Trin1 model specifically permits one to precisely adjust the probabilities. Boyle could do as such as it were "generally". The explanation is that m is permitted to contrast from 1 in the modified trinomial cross section. Then again just the mean and fluctuation of the discrete-time measure are right in the Boyle model, while the third and fourth minutes (skewness and kurtosis) are likewise right in the Tian Trin2 model. In view of this instinct, one might expect that the two adjusted variants are essentially pretty much as exact as the Boyle model. Moreover, both changed adaptations of the trinomial model act appropriately in the cutoff as $\sigma \to 0$, while the Boyle model does not.

In Chapter 4, we will perform the numerical accuracy analysis and the empirical study. At first, the numerical accuracy and convergence properties of the modified estimate strategies, when contrasted with the CRR and Boyle models are analyzed in this segment.

Inside and out five cross section strategies are examined and analyzed. Mathematical reenactments are done in Matlab to compute the worth of an alternative on a stock that pays no dividend utilizing the information boundary esteems we have used. Afterwards, we will conduct an empirical option valuation study using the Cox-Ross-Rubinstein (1979), the Boyle Trinomial model (1986) and the Tian Modified models (1993).

In Chapter 5, we will introduce the conclusions drawn from the model comparison. Finally, we attach an appendix of the Matlab codes used for research.

CHAPTER 2

CRR BINOMIAL AND BOYLE TRINOMIAL MODELS

In this chapter we will get acquainted with basic financial concepts regarding the Binomial Model elaborated by Cox-Ros-Rubenstein (1979) and the Trinomial Model developed by Boyle (1986), which will help us to better understand the way that these models study the behavior of the asset we are considering.

2.1 Options

Options are described as one of the most widespread types of financial derivative instruments. It is an agreement that gives the proprietor (purchaser) the right, yet not the commitment, to purchase or sell an asset at a foreordained cost and time span.

Despite what might be expected, the vender of an option has the commitment and not the choice to make the exchange for the underlying asset with the holder (purchaser) of the option in the event that he exercises it. Depending on the category of option and its attributes, the transaction is executed at the end of the contract or during it.

Options are delegated as either call options or put options. The basic elements that characterize an option and which are defined in the contract are as follows:

- The underlying asset, which is any financial and non-financial product. A stock, an exchange rate, a commercial good (consumer or not) are examples of underlying securities in an option contract.
- The price of the underlying security, which also defines the cash flows at the expiration of the contract if it is a European type option or the cash flows at any time during the validity of the contract if it is an American type option.
- The strike value of the option, which is the foreordained worth at which the holder of an option will make the exchange, on the off chance that he exercises his option.
- The time to maturity of the option, which denotes the time until the expiration of the option. In European type options the expiration date coincides with the date on which the options can be exercised, while in American type options the exercise can take place at any time up to the expiration date of the option.
- The premium or cost of the option which is the value paid by the buyer of the option to the seller to obtain the right to make a transaction with the underlying security.
- The position in an option. Each investor can hold two opposing positions: the long position and the short position.

- The size of the contract (contract size). This is the quantity of stocks of the underlying asset that the holder has the right to purchase or offer to the backer of the option.
- The option type determines whether the right is of European or American type.
- The option class alludes to options that are of similar kind and on a similar fundamental title.
- The option series concerns rights that belong to the same class, have the same maturity and the same exercise price.

Furthermore, the options are differentiated according to the cash flow that they cause from their direct exercise, compared to the strike price and the price of the underlying asset, so we have:

- In the money: Exercising the option outcomes in a positive cash flow. [Figure (a)]
- At the money: Exercising the option outcomes in a zero-sum cash flow. [Figure (b)]
- Out of the money: Exercising the option outcomes in a negative cash flow. [Figure (c)]

Call Option	**S > K**
Put Option	**S < K**

Figure (a)

Call Option	**S = K**
Put Option	**S = K**

Figure (b)

Call Option	**S < K**
Put Option	**S > K**

Figure (c)

Due to the characteristics mentioned above, an option is only exercised if it is in the money and the value of the option corresponds to the intrinsic value in conjunction to the time value.

Thus, the intrinsic value for a European call option and for a European put option is respectively:

$$\max\{S-K,0\} \quad , \quad \max\{K-S,0\}$$

The time value indicates the value of the option that is retained if the exercise of it is postponed. Particularly important is the fact that an "in the money" American option has at least as much value as its intrinsic value, as its holder can generate a positive cash flow through immediate exercise.

Moreover, there are four possible positions in the options:

- "Long position" in a call option
- "Short position" in a call option

- "Long position" in a put option
- "Short position" in a put option

The above positions can provide us with the payoff and the profit, which a European option produces at the maturity. For the holder of the market position in a call option, denoting the exercise price with K, the expiration time with T and the price of the underlying asset at maturity with S_T, so if we have:

- $S_T > K$, then is recommended to exercise the call option and the profit will be $S_T - K$
- $S_T \leq K$, then the call option will not be exercised and expires with zero value and zero profit

Consequently, the payment for the buyer from such a call option at maturity is:

$$Payoff = \max\{S_T - K, 0\}$$

The above payoff also indicates the value of the option and is expressed with f_{call}.
The gain from exercising the call option for the buyer (long position) will be (without taking into consideration the time value of money):

$$(S_T - K) = \max\{S_T - K, 0\} = \begin{cases} S_T - K, & S_T > K \\ 0, & S_T \leq K \end{cases}$$

In case we take into consideration the premium (c), then it will be $(S_T - K) - c$. It also important to mention that if the value of the asset at the expiration date is equal to the exercise price, then an investor will have no real interest in exercising his option.
Similarly, for the holder of the market position in a put option, we have:

- $K > S_T$, in which occasion is recommended to exercise the put option and the profit will be $K - S_T$
- $S_T \geq K$, then the put option will not be exercised and expires with zero value and zero profit

Consequently, the payment for the buyer from such a put option at maturity is:

$$Payoff = \max\{K - S_T, 0\}$$

This payoff also indicates the value of the option and is expressed with f_{put}.

The gain from exercising the put option for the buyer (long position) will be (without taking into consideration the time value of money):

$$\left(K - S_T\right) = \max\left\{K - S_T, 0\right\} = \begin{cases} K - S_T, & K > S_T \\ 0, & K \leq S_T \end{cases}$$

In case we take into consideration the premium (c), then it will be $(K - S_T) - c$. Here as well, we must mention that if the value of the asset at the expiration date is equal to the exercise price, then an investor will have no real interest in exercising his option.

The criteria by which an investor chooses his position depends on his assessment of the market. Therefore:

- If he estimates that the price of the asset, e.g., of a stock will rise enough then he can take a long position on a call option. Thus, he succeeds in gaining access to the rise and protection under a possible fall in the market value.
- If he estimates that the stock price will decrease a little, then he can take a short position in a call option. This is how the price is reaped, hoping that the share price will not augment. Of course, he is exposed to the risk of exercising the option if the share price eventually rises, unless he has already filled his position with a share purchase.
- If he estimates that the stock price will decline considerably, then he can take a long position in a put option. If this happens, then he benefits from the fall, otherwise he is protected against the rise, since the maximum he can lose is the price of the option he purchased. Alternatively, he could borrow the share and sell it, known as "short – selling".
- If he estimates that the share price will rise a bit, then he can take a short position in a put option. Along these lines, he secures the worth of the option with the assumption that the stock value cost will not decline. He is exposed to the risk of exercising the option if the share price eventually falls.

Hence, the elements that affect the price of an option are mentioned subsequently:

- Price of the underlying security: Call options acquire greater value when the price of the underlying security increases, in which case the value of the put options decreases.
- Exercise price: Put options acquire greater value when the exercise price increases, while then the value of the call options decreases.
- Maturity: In the American type of options, the one with the longest duration has a greater value, while in the European type of options the impact of time on their value is uncertain.
- Volatility of the underlying security: Instability of the hidden security: The worth of the options increments when the volatility of the underlying asset increments.

- Risk free interest rate: The value of the put option decreases when the interest rate increases, while then the value of the call option increases.
- Dividend yield of the underlying security: Any anticipated dividend is contrarily corresponded with the worth of a call option, while it is positively related with the worth of the put option.

Therefore, for the purpose of estimating the value of a call or put option at the time of the agreement in order to give the appropriate premium, its final price must be calculated as reliably as possible and then the final cash flows must be discounted. So, the difficulty of valuing an option lies in the primary difficulty of valuing any asset with uncertain income.

2.2 Binomial Tree Model

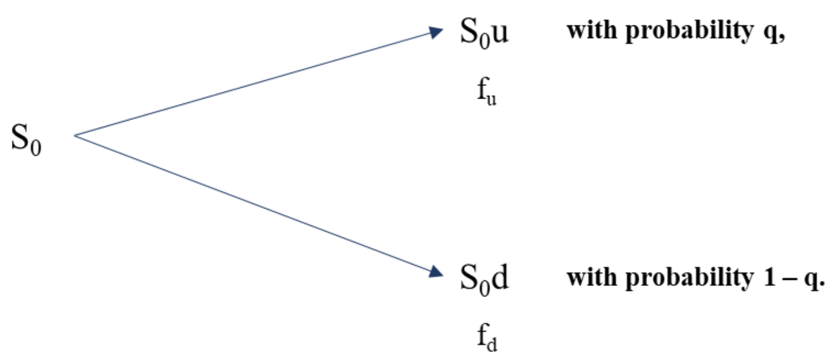
A handy and widely known method for pricing options includes building a binomial tree. This is a diagram introducing two divergent outcomes that may arise by the stock price during the life of an option. The main assumption is that the stock value brings up the rear a random walk. In each time step, it has a specific probability of moving up by a specific percentage amount and a specific probability of moving down by a specific percentage amount. In the boundary, as the time step becomes lesser, this technique appropriates the same process results as the Black–Scholes–Merton model. Indeed, it is well established that for instance, the European option price given by the binomial tree converges to the Black–Scholes–Merton price as the time step becomes lesser.

The binomial model for valuing an option was founded in 1979 by John Cox, Stephen Ross and Mark Rubinstein (CRR). The model is superstructure on the idea of Black & Scholes (1973). However, the formula presented in their article is at a discrete time, in contrast to that of Black & Scholes (1973) which is at a continuous time.

CRR (1979) constructed a binomial tree that presents different possible course of the underlying share price until the end of the option. The basic premise they make for the stock price is that it follows a random walk. At each time step the stock price has a certain probability of rising and a certain probability of falling.

We assume that the period we study the share's price lasts for one period. The kickoff of this period is time zero (0) and its end is time one (1). At time zero we are in the present and we consider a stock whose price is known to us. This value is obviously always positive and is denoted as S_0. We define the contingencies u and d, which represent the ascent and the descent of the share price, respectively. At time one (1), the stock price can take two possible prices depending on which probability comes true. These values are denoted by $S_0(u)$ and $S_0(d)$. Additionally, we will denote by "q" the probability of an increase, and "1 – q" the probability of a fall in the share price.

At time zero we do not know which of the two possibilities will take place at time one. We will only find out the outcome of the two possibilities when we finally reach at moment one, not earlier. However, we know in advance the price that the stock will receive under each circumstance, event u or event d. Let's define two positive numbers, u and d, which originate from the words "Up" and "Down". If event u occurs, the stock will take the value $u \cdot S_0$, while if event d occurs, the stock will take the value $d \cdot S_0$. The logical outcome is to take the assumption that u > d should hold. If u = d was valid, the stock price at time one would not be random, and the model would be shiftless. From the above, it is reasonable to conclude that "u" gets values greater than one and "d" values less than one, but not negative. Everything we have mentioned so far can be graphically depicted in the subsequent figure:

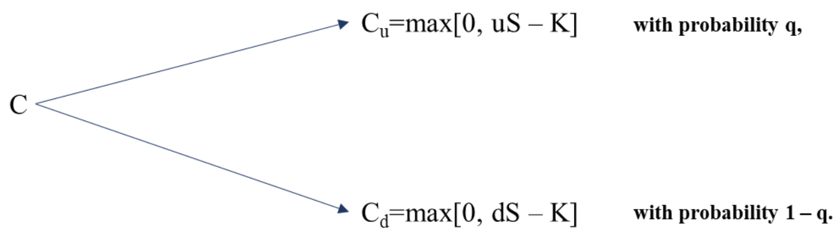


Then we set the discount rate, which we consider equal to the interest rate of a risk-free investment (e.g. government bonds) and we denote it "r_f". In plain words, someone who invests one unit of a currency at time zero, will then get $(1 + r_f)$ at time one. At the present time, we presume that the interest rate is not continuous compounded. Obviously, the interest rate will always get a positive price.

For the purpose of a market to be consider as "fair" (and as we will see later, in "equilibrium"), it is essential that there is no opportunity to make a certain (discounted) profit or otherwise arbitrage for any investor. However, how can an investor make a profit without risk? A typical example is a stock that is traded not only in the domestic market, but also in a foreign stock exchange market. There, may be a share price in one of the two markets that has not adjusted properly due to the ever-changing currency exchange rate. An investor who realizes this difference, is capable to (theoretically momentarily) buy from the stock market, where the stock is undervalued and sell it the one where it is overvalued, managing to make a definite profit instantaneously. In fact, markets sometimes offer arbitrage opportunities. Nevertheless, this happens for a very short time because it is quickly perceived and eliminated (because the demand for the undervalued and the supply of the overvalued increases as a result of which the two prices change and reach a level where there is no longer an opportunity for arbitrage) .
In the model we are considering, the relation $0 < d < 1 + r < u$ must be valid, otherwise the stock always has a better return (when $d \geq 1 + r$) and it does not make sense to issue bonds

or it always has a worse return (when $u \leq 1 + r$) and the issuance of shares does not make sense. Otherwise, we can say in an equivalent way that the above relation should be valid so that there is no possibility for arbitrage. Indeed, if $d \geq 1 + r$ is valid, then an investor can in year zero buy the share borrowing money (i.e. by issuing bonds) and in year 1 sell the share and repay its borrowers with a definite profit. In the event of, $u \leq 1 + r$ the investor can take the opposite positions (bond purchase, open stock sale) and making a definite profit, as previously.
It is obvious that the model we presented is inappropriate to describe the movement of a stock satisfactorily. But when it involves many periods, it is a very good approach to continuous time models.

Let us now see how the binomial model is used in practice. Suppose we have a European Call option on a stock with a price K. Based on what we have explained about the way that options make a profit, it is easy to see that $S_0 d < K < S_0 u$, should apply. If the share price falls, the holder does not exercise the option. Otherwise the option is exercised, and the profit is $(S_0 \cdot u - K)$. Let C_u be the value of the option (or otherwise the profit it returns) at time one if there is an increase in the share price, C_d the corresponding value in the event of price turndown, and C the requested value of the option at time zero.



In order to define C, we use a self-financed strategy. We build the so-called "hedging portfolio" which includes money deposits (or zero risk bonds) and shares and secures the buyer of the option from the risk posed by its purchase. The value of this portfolio at time one (but also any time at the multi-period model) is equal to the no-arbitrage value of the option at the same time, regardless of the event (rise or fall) that occurred. Thus, the initial value of the portfolio will be equal to the requested no-arbitrage value of the option. In order to corroborate what has just been mentioned, the appropriate interventions must be made in the composition of the portfolio at any time. Thus, the collateral portfolio is dynamic and self-financed.

Continuous Compounding

To begin with continuous compounding, we assume that we start with a portfolio which includes Δ stocks and a short position in a call option, at time zero. So, the construction cost of the portfolio is $P = \Delta S - c$. At time one (1) the value of the portfolio will be as it is depicted subsequently:

$$P = \Delta S - c \quad \nearrow \; P_u = \Delta uS - c_u \quad \textbf{with probability p,}$$
$$\qquad\qquad\quad \searrow \; P_d = \Delta dS - c_d \quad \textbf{with probability 1 – p.}$$

Since the portfolio is in the neutral risk world we can equate the two final nodes.

$$P_u = P_d \Rightarrow \Delta uS - c_u = \Delta dS - c_d \Rightarrow \boxed{\Delta = \frac{c_u - c_d}{S(u-d)}} \qquad (2.1)$$

We are aware of the value of the portfolio at time one (1), which will be $P = \Delta uS - c_u = \Delta dS - c_d$.

Additionally, since there is no arbitrage opportunity, and because the value of future option equals the value of the future portfolio, the value of today's option must equal the value of today's portfolio. Therefore: $P = P_1 e^{-rT} = (\Delta uS - c_u)e^{-rT}$

Thus, by equating the last condition with the preliminary cost of the portfolio, we can solve in terms of the value of the purchase option "c".

$$\Delta S - c = (\Delta uS - c_u)e^{-rT} \Rightarrow$$
$$c = \Delta S - (\Delta uS - c_u)e^{-rT} \overset{(2.1)}{\Rightarrow}$$
$$c = e^{-rT}\left[\frac{e^{rT} - d}{u-d}c_u + \frac{u - e^{rT}}{u-d}c_d\right] \Rightarrow$$
$$\boxed{c = e^{-rT}\left[pc_u + (1-p)c_d\right]} \qquad (2.2)$$

Where, $p = \dfrac{e^{rT} - d}{u-d} \quad and \quad 1 - p = \dfrac{u - e^{rT}}{u-d}$

In this neutral hazard environment, p = q. Therefore, we interpret p as the probability of an upward thrust in the share price, and 1 - p as the probability of a downward thrust in the share price.

The present value of the option corresponds to the discounted future risk-free income.

$$c = e^{-rT}\hat{E}(c_T)$$

The expected return on the stock corresponds to the risk-free interest rate, which means that the expected price of the stock at maturity is $\hat{E}(S_T) = Se^{rT}$.

In the world of neutral risk, since the price of the asset follows the lognormal distribution, the properties arise for time interval [t, t + Δt] are,

$$\boxed{E(S_{t+\Delta t}) = Se^{r\Delta t}} \qquad (2.3)$$

$$\boxed{Var(S_{t+\Delta t}) = S^2 e^{2r\Delta t}(e^{\sigma^2 \Delta t} - 1)} \qquad (2.4)$$

From relation (2.3) we have:

$$\begin{aligned}
&E(S_{t+\Delta t}) = Se^{r\Delta t} \Rightarrow \\
&puS + (1-p)dS = Se^{r\Delta t} \Rightarrow \\
&pu + (1-p)d = e^{r\Delta t} \Rightarrow \\
&\boxed{p = \frac{e^{r\Delta t} - d}{u - d}} \qquad (2.5)
\end{aligned}$$

Analyzing equation (2.4) it follows:

$$\begin{aligned}
&Var(S_{t+\Delta t}) = E({S_{t+\Delta t}}^2) - E(S_{t+\Delta t})^2 = S^2 e^{2r\Delta t}(e^{\sigma^2 \Delta t} - 1) \Rightarrow \\
&pu^2 S^2 + (1-p)d^2 S^2 - S^2 e^{2r\Delta t} = S^2 e^{2r\Delta t}(e^{\sigma^2 \Delta t} - 1) \Rightarrow \\
&pu^2 + (1-p)d^2 = e^{2r\Delta t}(e^{\sigma^2 \Delta t} - 1) + e^{2r\Delta t} \Rightarrow \\
&pu^2 + (1-p)d^2 = e^{2r\Delta t}(e^{\sigma^2 \Delta t} - 1 + 1) \Rightarrow \\
&pu^2 + d^2 - pd^2 = e^{2r\Delta t + \sigma^2 \Delta t} \Rightarrow \\
&p(u^2 - d^2) + d^2 = e^{2r\Delta t + \sigma^2 \Delta t} \overset{(2.5)}{\Rightarrow} \\
&\frac{e^{r\Delta t} - d}{u - d}(u-d)(u+d) + d^2 = e^{2r\Delta t + \sigma^2 \Delta t} \Rightarrow \\
&e^{r\Delta t}(u+d) - 1 = e^{2r\Delta t + \sigma^2 \Delta t} \Rightarrow \\
&e^{r\Delta t}(u + \frac{1}{u}) - 1 = e^{2r\Delta t + \sigma^2 \Delta t} \Rightarrow \\
&u^2 e^{r\Delta t} - u(1 + e^{2r\Delta t + \sigma^2 \Delta t}) + e^{r\Delta t} = 0
\end{aligned}$$

The solution of the quadratic equation obtained is:

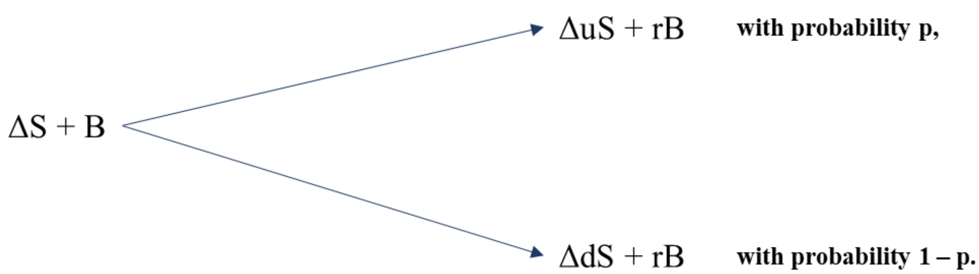
$$u = \frac{(1 + e^{2r\Delta t + \sigma^2 \Delta t}) \pm \sqrt{(1 + e^{2r\Delta t + \sigma^2 \Delta t})^2 - 4e^{2r\Delta t}}}{2e^{r\Delta t}}$$

Which through the Taylor development is simplified and the final form is achieved:

$$u = e^{\sigma\sqrt{\Delta t}} \quad , \quad d = e^{-\sigma\sqrt{\Delta t}} \quad , \quad p = \frac{e^{r\Delta t} - d}{u - d} \; .$$

<u>Discrete compounding</u>

Let's examine it more thoroughly. We assume that we start with a portfolio which consists of D shares and B cash (or zero risk bonds) which we deposit with an interest rate r_f, at time zero. The current value of the portfolio is ΔS + B. At time one (1) the value of the portfolio will be as follows:



We choose D and B in such a way that the two possible values of the portfolio at the end of the period are equal to the corresponding value of the purchase option. Grounded on the aforementioned statements, the self-financed strategy prescribes that the following relationships should be valid:

$$\boxed{\Delta u S + rB = C_u} \qquad (2.6)$$
$$\Delta d S + rB = C_d$$

Solving the system of equations, it turns out that:

$$\boxed{\Delta = \frac{C_u - C_d}{(u-d)S} \quad and \quad B = \frac{C_d u - C_u d}{(u-d)r}} \qquad (2.7)$$

Since there are no opportunities for arbitrage and since the value of the option in the future is equal to the value of the portfolio in the future, the value of the option today should be equal to the value of the portfolio today. Therefore:

$$\boxed{C = \Delta S + B = \frac{C_u - C_d}{u-d} + \frac{C_u d - C_d u}{(u-d)r} = \left[(\frac{r-d}{u-d})C_u + (\frac{u-r}{u-d})C_d \right] / r} \qquad (2.8)$$

We define:

$$\boxed{p \equiv \frac{r-d}{u-d} \quad and \quad 1-p \equiv \frac{u-r}{u-d}} \qquad (2.9)$$

Then relation (2.8) is simplified and written as follows:

$$\boxed{C = [pC_u + (1-p)C_d]/r} \qquad (2.10)$$

Relation (2.10) is the valuation formula of the purchase option, before its expiration, for a period in terms S, K, u, d, r. The formula mentioned above, has several noteworthy features. Initially, the probability q does not flash up in the formula.
This means that even if different investors have different preferences in the probability of the stock moving up or down, eventually everyone could agree on the price C, as it is independent of q.

Then the price of the option to buy is independent of investors' preferences for risk. For the construction of the formula the only assumption that is made is that every investor prefers more wealth than less and therefore there would be an incentive for arbitrage.

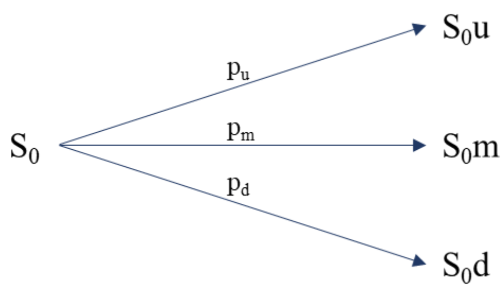
Therefore, the same formula could be used whether the investor is risk-preferring or risk-averse. Also, the only random variable on which the price of the option depends is the share price itself. Videlicet, it is not determined by the market appraisements of other securities or portfolios.

Finally, it is noted that $0 < p < 1$ and therefore could be characterized as a pseudo-probability. In fact, p is the value of q under the conjecture that all investors are risk averse. So, the value of the purchase option is represented as its expected future price, discounted in a world of neutral risk.

2.3 Trinomial Tree Model

In this section we will analyze the valuation of options through a trinomial tree model. This is a more accurate model than CRR Binomial model as it also incorporates the possibility that the share price will remain stable at the end of a period, which makes it a more realistic approach.

We are in a risk-neutral world where S is your asset's current value, which follows the lognormal distribution. The composition of the trinomial tree gives the green light for the value of the underlying asset to shift towards one of the three values assigned as up (u), down (d) and middle (m). The set of risk-neutral probabilities, p_u, p_d and p_m, are associated with these branches and offer a higher rate of convergence if the tree is built symmetrically for some applications according to Figlewski and Gao (1999). A method for determining the probabilities pu, pm, pd in the risk-neutral world results from the basic assumption that the price of the underlying asset follows the logarithmic (lognormal) distribution.



$$\mathbf{p_u + p_m + p_d = 1}$$

Specifically, five new unknown variables appear, u, d, p_u, p_m, and p_d which have yet to be found. One way leads through the types that have already emerged from the analysis of the binomial tree.

$$\boxed{ud = 1 \Rightarrow ud = m^2 = 1} \qquad (2.11)$$

This equation's motivation is as follows. Unlike the preceding section's binomial lattice, the above trinomial lattice does no longer commonly recombine properly. The previous equation guarantees that an upward pass accompanied through a downward pass is equal to a downward pass accompanied through an upward pass. A downward movement followed by a middle movement is the same as a middle movement followed by a downward movement. In fact, with this constrain, the range of nodes on an N-duration trinomial lattice is decreased from $\left(3^{N+1}-1\right)/2$ to $\left(N+1\right)^2$. In that way the trinomial lattice is considered computationally efficient.

For the alternative two extra constraints, Boyle (1986) selected $m=1$, $u=e^{\lambda\sigma\sqrt{\Delta t}}$ and obtained a trinomial model, wherein $\lambda \geq 1$ is a parameter that is defined by its users.

However, if the two extra constrains are used rather to offer accurate values for better moments of the trinomial distribution, which include the third and/or fourth moments, convergence is likely to be maximized.

The constrains which are adequate to assure the desired convergence are much like the ones withinside the binomial case. Specifically, those constrains are:

$$\boxed{p_u + p_m + p_d = 1}, \quad 0 < p_u, p_m, p_d < 1 \quad (2.12)$$

$$\boxed{p_u u + p_m m + p_d d = M} \quad (2.13)$$

$$\boxed{p_u u^2 + p_m m^2 + p_d d^2 = M^2 V} \quad (2.14)$$

$$where, \quad M = e^{r\Delta t} \quad \& \quad V = M^2\left(e^{\sigma^2 \Delta t} - 1\right).$$

Therefore, the mean value of the discrete distribution is equal to the mean value of the logarithm.

$$E(S_{t+\Delta t}) = Se^{r\Delta t} \overset{M=e^{r\Delta t}}{=} SM \Rightarrow$$
$$p_u uS + p_m S + p_d dS = SM \Rightarrow$$
$$p_u u + p_m + p_d d = M \Rightarrow$$
$$p_u u + p_m + p_d \frac{1}{u} = M \overset{(2.12)}{\Rightarrow}$$
$$p_u u + (1 - p_u - p_d) + p_d \frac{1}{u} = M \Rightarrow$$
$$p_u(u-1) + p_d(\frac{1}{u} - 1) = M - 1 \Rightarrow$$
$$\boxed{p_d = \frac{(M-1) - p_u(u-1)}{\left(\frac{1}{u} - 1\right)}} \quad (2.15)$$

At the same time, the variance of the discrete distribution is equal to the variance of the logarithmic distribution.

$$Var(S_{t+\Delta t}) = S^2 e^{2r\Delta t}(e^{\sigma^2 \Delta t} - 1) = S^2 V \Rightarrow \quad , \quad where \quad V = e^{2r\Delta t}(e^{\sigma^2 \Delta t} - 1)$$
$$E(S_{t+\Delta t}{}^2) - E(S_{t+\Delta t})^2 = S^2 V \Rightarrow$$
$$p_u u^2 S^2 + p_m S^2 + p_d d^2 S^2 - S^2 M^2 = S^2 V \Rightarrow$$
$$p_u u^2 S^2 + p_m S^2 + p_d \frac{1}{u^2} S^2 - S^2 M^2 = S^2 V \Rightarrow$$
$$p_u u^2 + p_m + p_d \frac{1}{u^2} = V + M^2 \overset{(2.12)}{\Rightarrow}$$
$$p_u u^2 + (1 - p_u - p_d) + p_d \frac{1}{u^2} = V + M^2 \Rightarrow$$
$$\boxed{p_u(u^2 - 1) + p_d\left(\frac{1}{u^2} - 1\right) = V + M^2 - 1} \quad (2.16)$$

On that note,

$$(2.16)\overset{(2.15)}{\Rightarrow} p_u(u^2-1)+\left[\frac{(M-1)-p_u(u-1)}{\left(\frac{1}{u}-1\right)}\right]\left(\frac{1}{u^2}-1\right)=V+M^2-1\Rightarrow$$

$$p_u(u^2-1)+\left[(M-1)-p_u(u-1)\right]\left(\frac{1}{u}+1\right)=V+M^2-1\Rightarrow$$

$$p_u(u^2-1)+(M-1)\left(\frac{1+u}{u}\right)-p_u(u-1)\left(\frac{1+u}{u}\right)=V+M^2-1\Rightarrow$$

$$p_u\left[(u^2-1)-\left(\frac{u^2-1}{u}\right)\right]=V+M^2-1-(M-1)\left(\frac{1+u}{u}\right)\Rightarrow$$

$$p_u\left[(u^2-1)(u-1)\right]=u\left(V+M^2-1\right)-(M-1)(1+u)\Rightarrow$$

$$p_u\left[(u^2-1)(u-1)\right]=u\left(V+M^2-1\right)-Mu+u-(M-1)\Rightarrow$$

$$p_u\left[(u^2-1)(u-1)\right]=u\left(V+M^2-M\right)-(M-1)\Rightarrow$$

$$\boxed{p_u=\frac{u\left(V+M^2-M\right)-(M-1)}{(u^2-1)(u-1)}} \qquad (2.17)$$

Last of all, for p_d we head back to equation (2.15) and by using the results from equation (2.17) we have:

$$(2.15)\Rightarrow\left(\frac{1}{u}-1\right)p_d=(M-1)-p_u(u-1)\overset{(2.17)}{\Rightarrow}$$

$$\left(\frac{1-u}{u}\right)p_d=(M-1)-\left[\frac{u\left(V+M^2-M\right)-(M-1)}{(u^2-1)(u-1)}\right](u-1)\Rightarrow$$

$$\left(\frac{u-1}{u}\right)p_d=\frac{u\left(V+M^2-M\right)-(M-1)-(M-1)(u^2-1)}{(u^2-1)}\Rightarrow$$

$$\left(\frac{u-1}{u}\right)p_d=\frac{u\left(V+M^2-M\right)-(M-1)u^2}{(u^2-1)}\Rightarrow$$

$$\boxed{p_d=\frac{u^2\left(V+M^2-M\right)-(M-1)u^3}{(u^2-1)(u-1)}} \qquad (2.18)$$

In the aggregate we have:

$$p_u=\frac{u\left(V+M^2-M\right)-(M-1)}{(u^2-1)(u-1)},\quad p_d=\frac{u^2\left(V+M^2-M\right)-(M-1)u^3}{(u^2-1)(u-1)},\quad p_m=1-p_u-p_d,$$

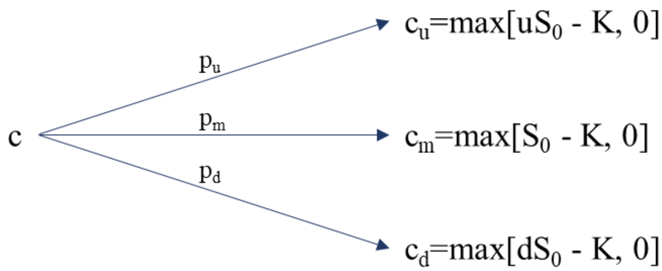
$$u=e^{\lambda\sigma\sqrt{\Delta t}}\quad \&\quad d=e^{-\lambda\sigma\sqrt{\Delta t}}.$$

It is observed that the transition probabilities of the resulting trinomial model using the u-parameters defined in the CRR (1979) binomial model and the definition of m = 1 are not between 0 and 1 but are even negative. Therefore, Boyle proposed the use of a parameter λ

> 1 on the basis of which we have $u_{bin} = e^{\lambda\sigma\sqrt{\Delta t}}$ *and* $d_{bin} = e^{-\lambda\sigma\sqrt{\Delta t}}$, although this parameter also offers negative transition possibilities for small values of λ. By testing different values for λ, a range of u-values is obtained in which there is an interval that simultaneously produces acceptable values for all probabilities. The best results were obtained when the parameter λ was set so that the transition probabilities were approximately the same.

Boyle also observed that the precision of the triple jump method with 5 time intervals for a range of values was comparable to that of the CRR method with 20 time intervals.
Later, Komorád (1990) improved the model to correct the possible problem of negative transition probabilities, giving a feasible set of probabilities for every $\lambda \geq 1$.

As shown in the one-step binomial tree, we have respectively in the one-step trinomial:



Thus, in order by finding the intrinsic value in the last node and with the formula we tested in the binomial model we have:

$$\boxed{c = e^{-r\Delta t}(p_u c_u + p_m c_m + p_d c_d)} \qquad (2.19)$$

CHAPTER 3

TIAN MODIFIED BINOMIAL AND TRINOMIAL MODELS

As already mentioned, the binomial model simulates the movements of an asset. In this chapter we will show how the CRR binomial model, and the Boyle trinomial model are extended and replaced by the modified binomial model, and the modified trinomial models elaborated by Tian (1993), also known as BIN and TRIN1 & TRIN2, respectively.

3.1 Modified Binomial Tree Model

Cox, Ross, and Rubinstein initially elaborated the lattice methodology in 1979. They elaborated a discrete-time, binomial methodology for option valuation. The core of their strategy is the composition of a binomial lattice of share values, where they preserve the norm of the risk-neutral valuation world. With a specific choosing as regards binomial parameters, as for instance probabilities and jumps, they demonstrated the convergence of the CRR binomial model with the Black-Scholes model. The CRR process has been expanded consequently by a variety of investigators.

In this section we demonstrate a modified approach to the selection of lattice parameters that contain probabilities and jumps with respect to the binomial model. This general approach is feasible to be applied to any multi-dimensional lattice approach. It is identified that the elementary circumstances for a lattice model to converge to the Black-Scholes model do not provide a sole solution to the lattice parameters. Further constraint(s) on the lattice parameters is (are) required. In the modified methodology the lattice parameters are selected in such a way that the accuracy of approximation is enhanced. The subsequent presentation pertains to the approach applied for the modification of the CRR binomial model.

To begin with, suppose you want to model the price movement of a non-dividend stock over the period from t = 0 to t = T. It is assumed that in a risk-neutral world, the stock price follows the following stochastics process:

$$\boxed{\frac{dS_{(t)}}{S_{(t)}} = rdt + \sigma dz} \qquad (3.1)$$

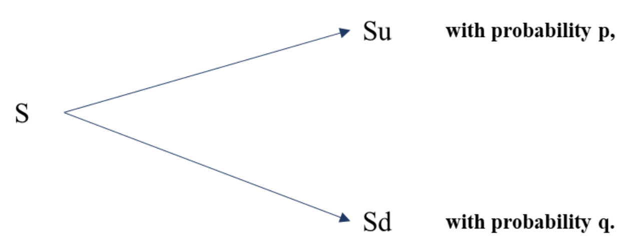
where the pair of r and σ are steady, are the immediate proportional deviation and volatility rates, respectively. A logarithmic alteration elucidates the above process to:

$$\boxed{d\log S_{(t)} = \left(r - \frac{\sigma^2}{2}\right)dt + \sigma dz} \qquad (3.2)$$

It trails behind promptly that the share value is lognormally distributed. In particular, the yield retention period $\log[S_{(t)} / S_0]$ is normally distributed with mean value $\left(r - \frac{\sigma^2}{2}\right)t$ and variance $\sigma^2 t$. For the most part, the mth non-central moment of the share value $S_{(t)}$ is stated by the succeeding formula:

$$\boxed{E[S_{(t)}{}^m \mid S_0] = S_0{}^m e^{[(mr+m(m-1)\frac{\sigma^2}{2})t]}} \qquad (3.3)$$

Based on the preceding continuous stochastic process we can elaborate a binomial estimation, as follows. The time to maturity (T) of an option on the share is divided into N equal sub-intervals of length $\Delta t = T / N$. During each period of time, say from $k\Delta t$ to $(k+1)\Delta t$, with $k = 0,\ 1, 2, \ldots,\ N-1$, stock price movements are assumed to follow a binomial process, sometimes referred to as the two-jump process. The share value jumps from its initial value, $S_{(k\Delta t)}$, either upward to $uS_{(k\Delta t)}$ or downward to $dS_{(k\Delta t)}$, where $0 < d < 1 < u$. This binomial branching process is shown in next figure, where p and q are the implicit probabilities in the risk neutral world.



This binomial lattice in overall has four input values, u, d, p , and q. These input values clearly determine the progression of share prices, which in turn defines a unique value of an option on the share. Nevertheless, these parameters cannot be selected arbitrarily, because the option price gained may not converge towards the corresponding limit. It is a broadly established fact that the price of an option is the discounted expectation of the final payoff according to the appropriate martingale measure. If a discrete-time process (e.g. a binomial process) converges against the continuous-time process, which is tracked by the stock price, i.e., a Geometric Brownian Motion, then the anticipations from the discrete-time case will converge to the continuous-time one.

Thereby, it is guaranteed that the option value gained from a discrete-time model is converged to the one from a continuous-time model.

Thus, the binomial parameters ought to be decided on such a way that the convergence of the share value gained from the discrete-time distribution to the one from a lognormal distribution in continuous-time is accomplished. According to Lindeberg's Central Limit Theorem, the subsequent statements are adequate to confirm this convergence:

(i) the possibilities [p and 1-p (=q)] are positive withinside the threshold between zero and one but not matched to either zero or one,
(ii) the possibilities aggregate to 1,
(iii) jumps (u and d) are unrelated to the share value level,
(iv) binomial distribution's mean is the same as the one of the lognormal distribution,
(v) binomial distribution's variance is the same as the one of the lognormal distribution.

The mathematical illustration of these constraints is:

$$\boxed{p+q=1}, \quad 0<p,q<1 \qquad (3.4)$$

$$\boxed{pu+qd=M} \qquad (3.5)$$

$$\boxed{pu^2+qd^2=M^2V} \qquad (3.6)$$

where, $M=e^{r\Delta t}$, $V=e^{\sigma^2\Delta t}$, and $\Delta t=T/N$. There are 4 undefined parameters, u, d, p , and q withinside the preceding 3 equations. Hence, for a sole answer one extra equation is required. The range for this extra constraint is theoretically unlimited, and there is no obvious criterion to select among these unlimited options. The optimal choice of this equation is the one, with the intention to derive the suited convergence properties of the binomial estimation process. With ulterior motive the above intention, the subsequent constraint is suggested:

$$\boxed{pu^3+qd^3=M^3V^3} \qquad (3.7)$$

This circumstance guarantees that the 3rd moment of the discrete-time procedure is likewise accurate in step with to the continuous-time procedure. Since the binomial distribution is a skewed one, this circumstance is probably more realistic and bring about an even more correct binomial process.

Solving equations (3.4) – (3.7), an answer to the 4 binomial unknowns is emanated, and the ensuing binomial model constitute a modified version of the original CRR model, which is mentioned to henceforth as the BIN model:

$$\boxed{p=\frac{M-d}{u-d},\quad q=1-p=\frac{u-M}{u-d}} \qquad (3.8)$$

$$\boxed{u=\frac{MV}{2}\left[(V+1)+\sqrt{V^2+2V-3}\right]} \qquad (3.9)$$

$$\boxed{d=\frac{MV}{2}\left[(V+1)-\sqrt{V^2+2V-3}\right]} \qquad (3.10)$$

Based on (3.9) – (3.10) we have:

$$ud=\left(\frac{MV}{2}\left[(V+1)+\sqrt{V^2+2V-3}\right]\right)\left(\frac{MV}{2}\left[(V+1)-\sqrt{V^2+2V-3}\right]\right)\Rightarrow$$

$$ud=\left(\frac{(MV)^2}{4}\left[(V+1)^2-\sqrt{V^2+2V-3}^{\,2}\right]\right)\Rightarrow$$

$$ud=\left(\frac{(MV)^2}{4}\left[V^2+2V+1-V^2-2V+3\right]\right)\Rightarrow$$

$$ud=\left(\frac{(MV)^2}{4}4\right)\Rightarrow\boxed{ud=(MV)^2} \qquad (3.11)$$

Equation (3.11) indicates that the binomial lattice grows approximately at the rate ($r + \sigma^2$). In order to estimate the precise cost of an option we can utilize a rearward recurring process. Operating rearwards from the maturity of the option, we are capable to estimate the value of the option at time $k\Delta t$ through the discounted anticipated value of the option at time $(k+1)\Delta t$. For a call option on a non-dividend paying share, such a rearward recursion can be done with the aid of using the subsequent recursive valuation equation:

$$\boxed{C(k\Delta t)=\frac{\left[pC_u\left((k+1)\Delta t\right)+qC_d\left((k+1)\Delta t\right)\right]}{e^{r\Delta t}}} \qquad (3.12)$$

For alternative options, principally American options, some slight adjustments to the preceding recursion equation are required.

The BIN design diversifies from the CRR binomial design in the succeeding two features. Initially, withinside the CRR version the variance of the share value is solely accurate in the threshold as the time step Δt approaches zero. In the BIN version, each of the mean and variance are accurate for any given Δt. Secondary, as a substitute of selecting the proper third moment (consequently accurate skewness), Cox, Ross, and Rubinstein (1979) decided on the subsequent circumstance for modeling plainness:

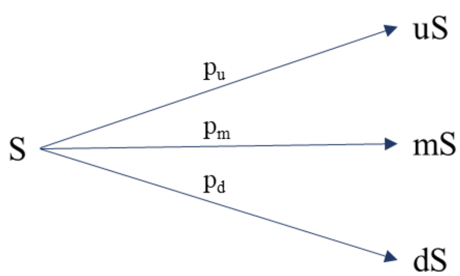
$$\boxed{ud=1} \qquad (3.13)$$

3.2 Modified Trinomial Tree Model

Boyle used the CRR model to refine it and proposed a trinomial option pricing model, also known as the triple jump model, in which the value of the share can either rise, fall, or remain the same over a given period. This procedure resulted in an entire pack of trinomial lattice parameters such as probabilities and three jumps. In addition, he examined the numerical precision of his model in comparison to that of the CRR binomial model. In a later effort, Boyle demonstrated how a five-jump, three-dimensional lattice can be developed to value options on two underlying securities, back in 1988.

The trinomial approximation technique is a clear-cut generalization of its binomial equivalent. Surmise that you desire to appraise a call option on a share which pays no dividend, matures at time t = T and it could be exercised at a striking price of K. For instance, suppose that the share value tracks the same stochastic dispersal procedure as expressed in the anterior section. Therefore, the trinomial approximation regarding the continuous-time procedure of the share value may be elaborated as it is depicted hereafter.

The time to maturity T of an option is separated into N equal sub-interims of length $\Delta t = T/N$. Throughout every single time span, for instance from $k\Delta t$ to $(k+1)\Delta t$ (with $k = 0, 1, 2, \ldots, N-1$), it is presumed that the value shifts of the share follow a trinomial procedure, e.g., a three-jump procedure. The share value can either shift from its opening price S(kΔt), up to uS(kΔt), down to dS(kΔt), or remain stable at mS(kΔt). This kind of trinomial process for a share value is presented in the following figure, where u > m > d > 0, u > 1, and d < 1.



$$\mathbf{p_u + p_m + p_d = 1}$$

This trinomial frame has six parameters, **u**, **m**, **d**, $\mathbf{p_u}$, $\mathbf{p_m}$, and $\mathbf{p_d}$, which is comprised of three jump parameters and three probability parameters. These parameters clearly define the progress of share values, which in turn defines a sole price for an option on the share. Once more, as within the binomial circumstance, they cannot be selected arbitrarily. They must be bounded in such a way that the convergence of the trinomial procedure to the lognormal distribution of the share value in continuous time is accomplished. The constraints that are adequate to ensure this convergence are identical to those in the binomial illustration. The fundamental constraints, plus the additional ones, which are required are depicted as needed.

$$\boxed{p_u + p_m + p_d = 1} \quad 0 < p_u, p_m, p_d < 1 \qquad (3.14)$$

$$\boxed{p_u u + p_m m + p_d d = M} \qquad (3.15)$$

$$\boxed{p_u u^2 + p_m m^2 + p_d d^2 = M^2 V} \qquad (3.16)$$

$$\boxed{ud = m^2} \qquad (3.17)$$

Thus, in this section we demonstrate a modified approach to the selection of lattice parameters that contain probabilities and jumps with respect to the trinomial model. This general approach is feasible to be applied to any multi-dimensional lattice approach. It is identified that the elementary circumstances for a lattice model to converge to the Black-Scholes model do not provide a sole solution to the lattice parameters. Further constraint(s) on the lattice parameters is (are) required. In the modified methodology the lattice parameters are selected in such a way that the accuracy of approximation is enhanced. The subsequent presentation pertains to the approach applied for the modification of the Boyle trinomial model.

3.2.1 The First Modified Trinomial Model

Remember that in the Boyle trinomial version states that "greatest findings were gained when the probabilities were approximately equal". In the modified trinomial lattice where m can differ from 1, it is possible to make these probabilities equivalent, so we have:

$$\boxed{p_u = p_m = p_d} \qquad (3.18)$$

Once we solve the system of equations (3.14) – (3.18), we come up to a modified trinomial model, which is mentioned here as the "TRINl" model:

$$\boxed{p_u = p_m = p_d = \frac{1}{3}} \qquad (3.19)$$

$$\boxed{u = K + \sqrt{K^2 - m^2}} \qquad (3.20)$$

$$\boxed{d = K - \sqrt{K^2 - m^2}} \qquad (3.21)$$

$$\boxed{m = \frac{M(3-V)}{2}} \qquad (3.22)$$

Where, $K = M\frac{(V+3)}{4}$.

3.2.2 The Second Modified Trinomial Model

To elaborate the second modified trinomial model, the subsequent two constraints are implemented:

$$\boxed{p_u u^3 + p_m m^3 + p_d d^3 = M^3 V^3} \qquad (3.23)$$

$$\boxed{p_u u^4 + p_m m^4 + p_d d^4 = M^4 V^6} \qquad (3.24)$$

These two equations ensure that the third and fourth moments of the trinomial distribution are accurate based on their counterparts of the continuous distribution. By solving the system of equations (3.14)-(3.17), (3.23), and (3.24) an answer detected is the following:

$$\boxed{p_u = \frac{md - M\left(m+d\right) + M^2 V}{\left(u-d\right)\left(u-m\right)}} \qquad (3.25)$$

$$\boxed{p_m = \frac{M\left(u+d\right) - ud - M^2 V}{\left(u-m\right)\left(m-d\right)}} \qquad (3.26)$$

$$\boxed{p_d = \frac{um - M\left(u+m\right) + M^2 V}{\left(u-d\right)\left(m-d\right)}} \qquad (3.27)$$

$$\boxed{u = K + \sqrt{K^2 - m^2}} \qquad (3.28)$$

$$\boxed{d = K - \sqrt{K^2 - m^2}} \qquad (3.29)$$

$$\boxed{m = MV^2 \quad , \quad K = \frac{M}{2}(V^4 + V^3)} \qquad (3.30)$$

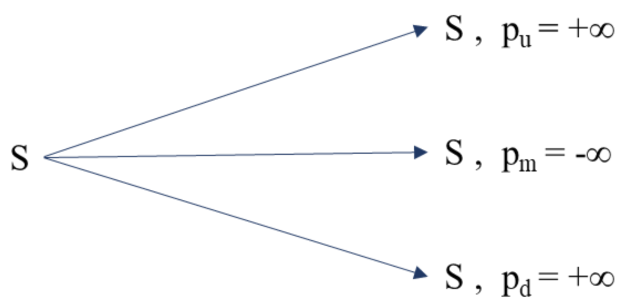
This modified trinomial model is mentioned as the “TRIN2” model.

The modified trinomial models are instinctively more attractive than the Boyle model. Remember that Boyle (1986) discovered the “optimal” solution through a trial-and-error experiment. While, here we have a systematic method to solve the problem. Specifically, the TRINl model permits balancing the odds (equal probabilities).The only way Boyle could accomplish it, was only “approximately”. The cause for this is that m may differ from 1 in the modified trinomial lattice. On the contrary, only the mean value and variance of the discrete-time procedure are accurate in the Boyle version, whilst the third and fourth moments (skewness and kurtosis) are also accurate in the TRIN2 version. Because of this statement, the average expectation is that the two modified versions are at minimum as accurate as the Boyle model.

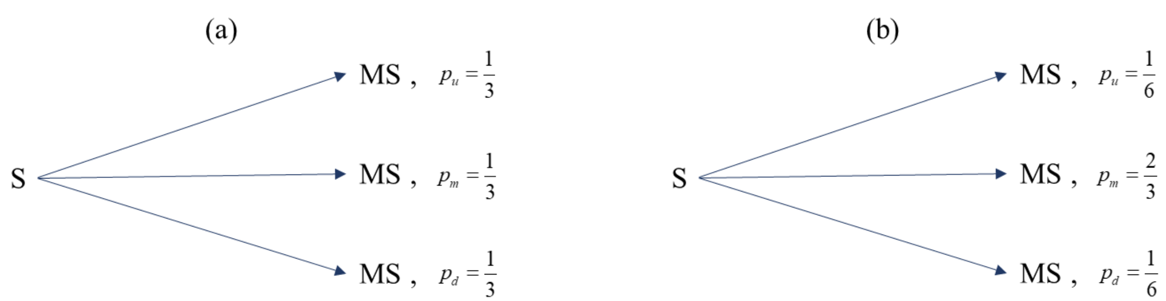
In addition, both modified versions of the trinomial model behave correctly at the limit as $\sigma \to 0$, whilst the Boyle model does not. Contemplate the price of a call option on a share that does not pay dividends. A little algebra demonstrates that as $\sigma \to 0$ the Black-Scholes formula is downsized to:

$$\boxed{C = \max\left\{S_0 - Ke^{-rT}, 0\right\}} \qquad (3.31)$$

Hereby, an approximation process should therefore lead to call prices which also approximate this limit. Nevertheless, the Boyle version does not converge at all in the case where $\sigma \to 0$. It is frivolous to demonstrate that when $\sigma \to 0$, the Boyle trinomial lattice is downscaled to the one shown in next figure. Undoubtedly, the Boyle lattice frame is inaccurate in this thresholding situation.



All the same, the modified versions do converge to the accurate solution when $\sigma \to 0$, as indicated in the equation (3.31), and the lattice frame for the two modified models is downscaled to the one shown in figure (a) and the one shown in figure (b), respectively.



Chapter 4

NUMERICAL ACCURACY AND EMPIRICAL STUDY

In this chapter we will apply at first numerical analysis, in order to study the accuracy of the models we have mention until now under different precision levels. After that, by using the market data we have chosen we perform an empirical study, so as to determine the ability of the models, which describes the price of the options and examines if the results coming out the numerical accuracy analysis are confirmed.

4.1 Numerical Accuracy

The mathematical precision and union properties of the altered estimate strategies, when contrasted with the CRR and Boyle models are analyzed in this segment. By and large five grid techniques are examined and analyzed. Mathematical recreations are performed in Matlab to figure the worth of a stock option that do not pay dividends utilizing the information boundary esteems given as following:

S0	K	r	sigma	T
200	185, 200, 215	4%	25%	6 months

Table 4.1 depicts the worth of the European call options obtained from the five programs in the network using time intervals of 5, 10, 20, …, 100. Obviously, the value of the call option obtained from all methods soon will converge to the price of Black and Scholes. Using 100 time-steps, the prices generated by all methods are within 3 cents of the Black and Scholes price, and the Black and Scholes price ranges from \$ 24.7643 to \$ 9.7205. Similar results were obtained from European put option studies using the same option parameter values and are recapitulated in Table 4.2.

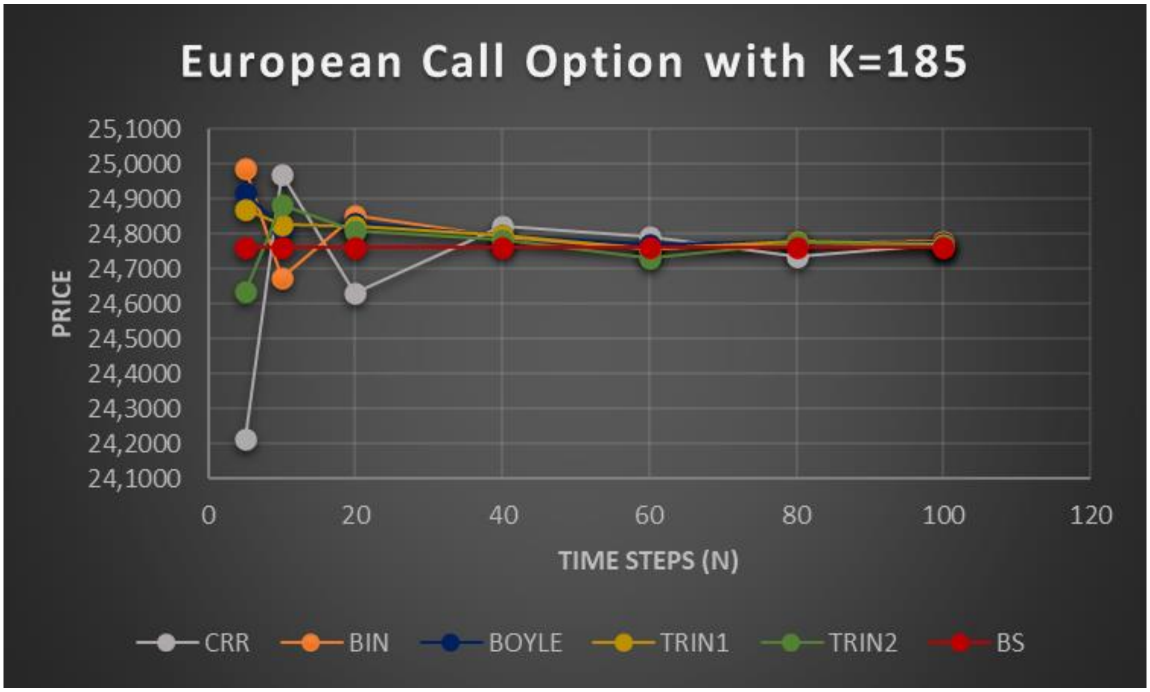
Time Steps (N)	CRR	BIN	BOYLE	TRIN1	TRIN2	BS
Exercise Price, K=185						
5	24,2134	24,9895	24,9190	24,8706	24,6368	24,7643
10	24,9722	24,6763	24,8206	24,8303	24,8837	24,7643
20	24,6317	24,8551	24,8343	24,8230	24,8114	24,7643
40	24,8248	24,7936	24,7959	24,7989	24,7837	24,7643
60	24,7949	24,7590	24,7737	24,7607	24,7318	24,7643
80	24,7348	24,7740	24,7786	24,7816	24,7754	24,7643
100	24,7709	24,7804	24,7773	24,7714	24,7665	24,7643
Exercise Price, K=200						
5	16,7084	16,0900	15,9298	15,9762	16,3427	16,0160
10	15,6701	16,3360	15,9651	16,0094	15,9332	16,0160
20	15,8417	16,1486	15,9894	16,0237	16,0098	16,0160
40	15,9286	15,9768	16,0024	16,0273	16,0382	16,0160
60	15,9576	16,0036	16,0069	16,0271	15,9854	16,0160
80	15,9722	16,0416	16,0091	16,0264	16,0354	16,0160
100	15,9810	16,0474	16,0105	16,0257	16,0256	16,0160
Exercise Price, K=215						
5	9,2035	10,3525	9,9354	9,9903	9,2919	9,7205
10	9,9851	9,7823	9,7140	9,6433	9,5708	9,7205
20	9,6416	9,8128	9,8057	9,8004	9,6810	9,7205
40	9,7670	9,7954	9,7333	9,7025	9,7213	9,7205
60	9,7695	9,7773	9,7457	9,7480	9,7296	9,7205
80	9,7224	9,7635	9,7165	9,7298	9,7318	9,7205
100	9,6943	9,7528	9,7366	9,7239	9,7322	9,7205

Table 4.1 – European Call prices

Time Steps (N)	CRR	BIN	BOYLE	TRIN1	TRIN2	BS
Exercise Price, K=185						
5	5,5502	6,3263	6,2558	6,2074	5,9735	6,1011
10	6,3089	6,0131	6,1574	6,1670	6,2205	6,1011
20	5,9685	6,1919	6,1710	6,1598	6,1481	6,1011
40	6,1616	6,1303	6,1326	6,1356	6,1205	6,1011
60	6,1317	6,0957	6,1105	6,0975	6,0685	6,1011
80	6,0716	6,1108	6,1154	6,1183	6,1122	6,1011
100	6,1076	6,1171	6,1140	6,1081	6,1033	6,1011
Exercise Price, K=200						
5	12,7482	12,1297	11,9696	12,0160	12,3824	12,0557
10	11,7098	12,3758	12,0048	12,0491	11,9730	12,0557
20	11,8815	12,1884	12,0291	12,0634	12,0496	12,0557
40	11,9683	12,0166	12,0421	12,0671	12,0779	12,0557
60	11,9974	12,0433	12,0466	12,0668	12,0252	12,0557
80	12,0119	12,0814	12,0489	12,0661	12,0752	12,0557
100	12,0207	12,0872	12,0502	12,0654	12,0653	12,0557
Exercise Price, K=215						
5	19,9462	21,0952	20,6781	20,7330	20,0346	20,4632
10	20,7279	20,5250	20,4567	20,3860	20,3135	20,4632
20	20,3843	20,5555	20,5484	20,5432	20,4237	20,4632
40	20,5097	20,5381	20,4760	20,4452	20,464	20,4632
60	20,5122	20,5200	20,4884	20,4907	20,4723	20,4632
80	20,4652	20,5062	20,4592	20,4725	20,4745	20,4632
100	20,4370	20,4955	20,4793	20,4666	20,4749	20,4632

Table 4.2 – European Put prices

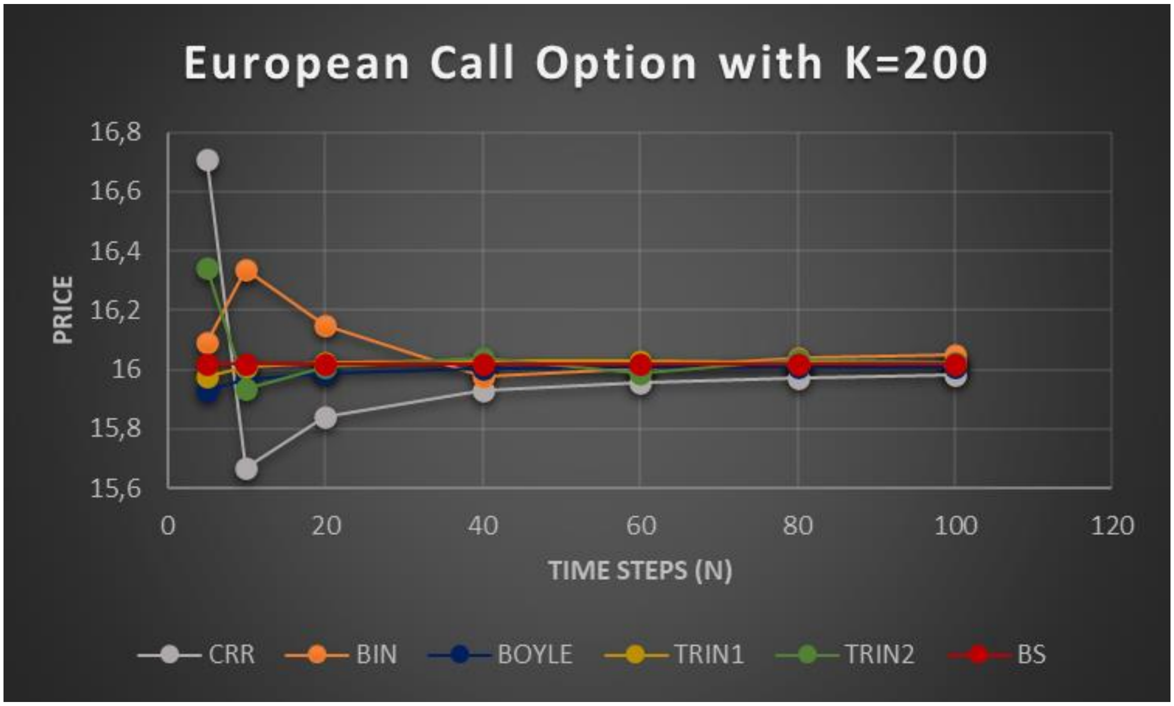
<u>European Call Option Prices with Strike Price=185</u>



Graph 4.1

- We can observe that, Tian Trin1 and Boyle almost concur with Black and Scholes from the beginning, where Time steps are equal to 5.
- From N=40 and thereon all models are close enough to Black and Scholes line.

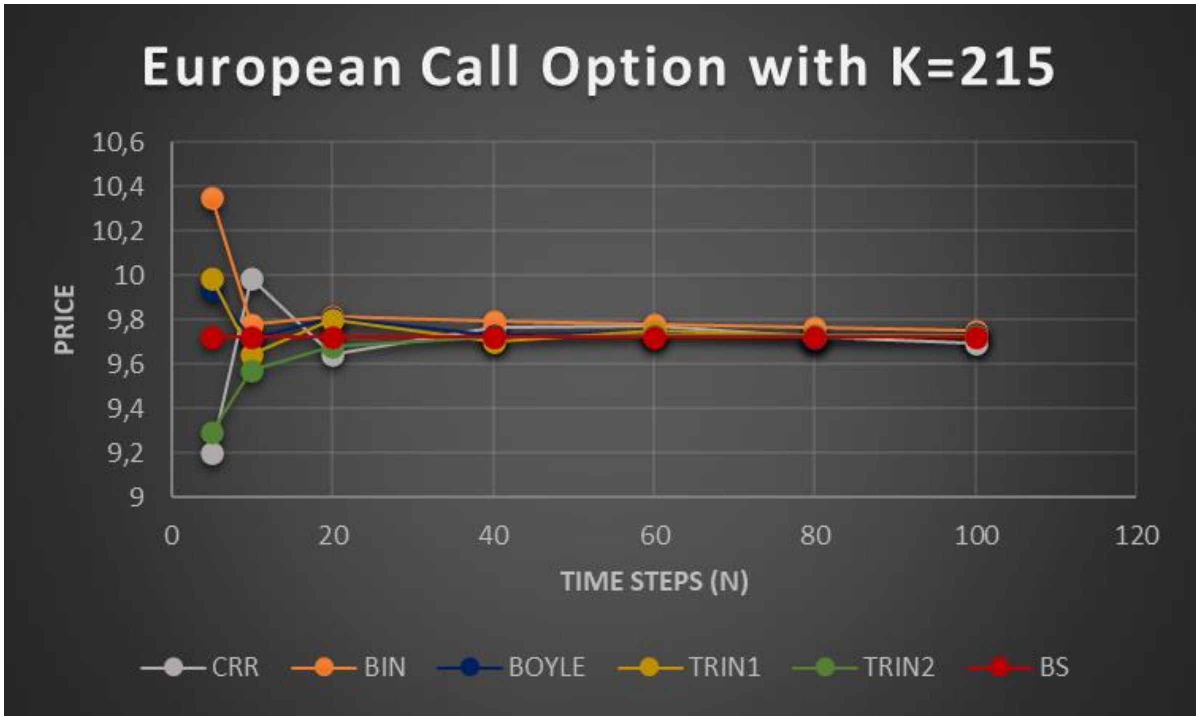
European Call Option Prices with Strike Price=200



Graph 4.2

- We can observe that, Tian Trin1 and Boyle almost concur with Black and Scholes from the beginning, where Time steps are equal to 5.
- CRR seems to be the least accurate model.

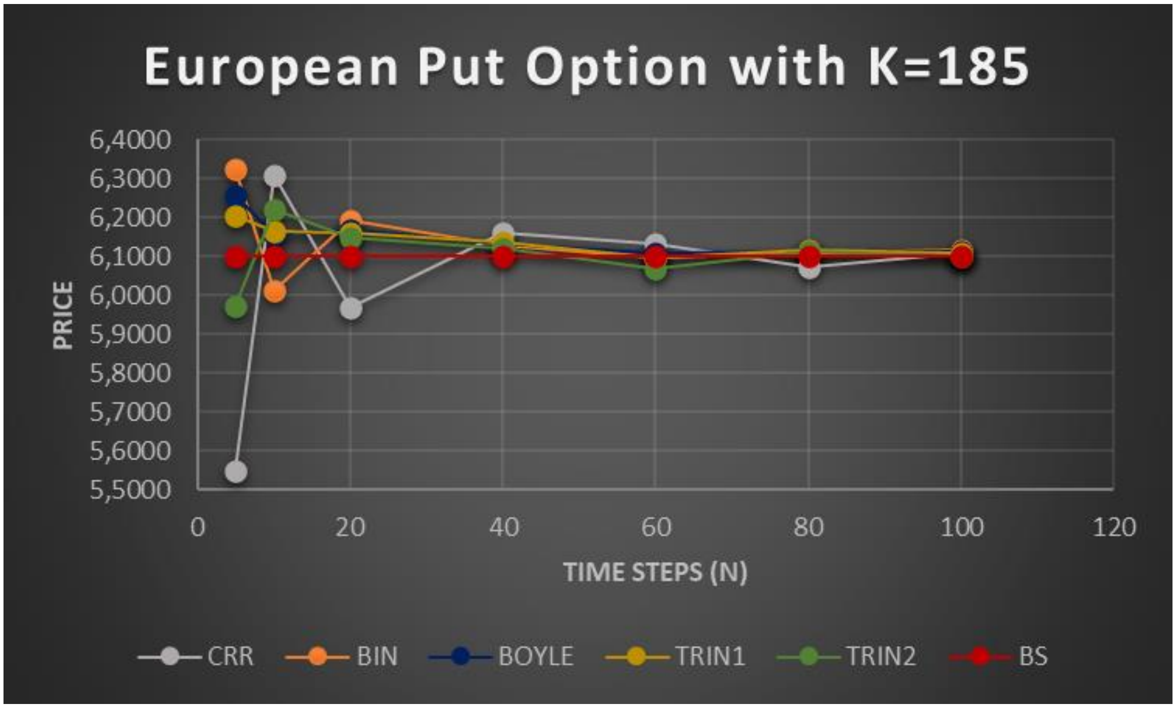
European Call Option Prices with Strike Price=215



Graph 4.3

- We can observe that, Tian Trin1 and Boyle almost concur with Black and Scholes from the beginning, where Time steps are equal to 5.
- CRR and Tian Bin seem to be the least accurate models.
- From N=40 and thereon all models are close enough to the Black and Scholes line.

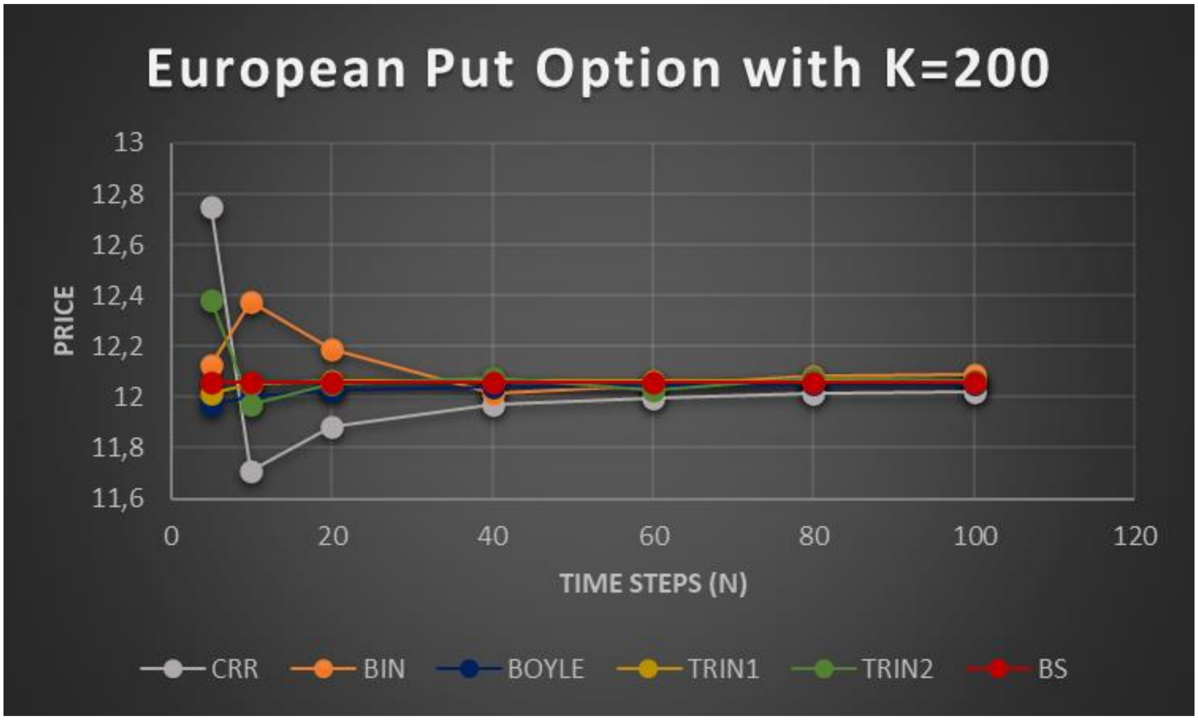
European Put Option Prices with Strike Price=185



Graph 4.4

- We can observe that, Tian Trin1 and Boyle almost concur with Black and Scholes from almost the beginning, where Time steps are equal to 10.
- As time steps increased, accuracy is improved too.

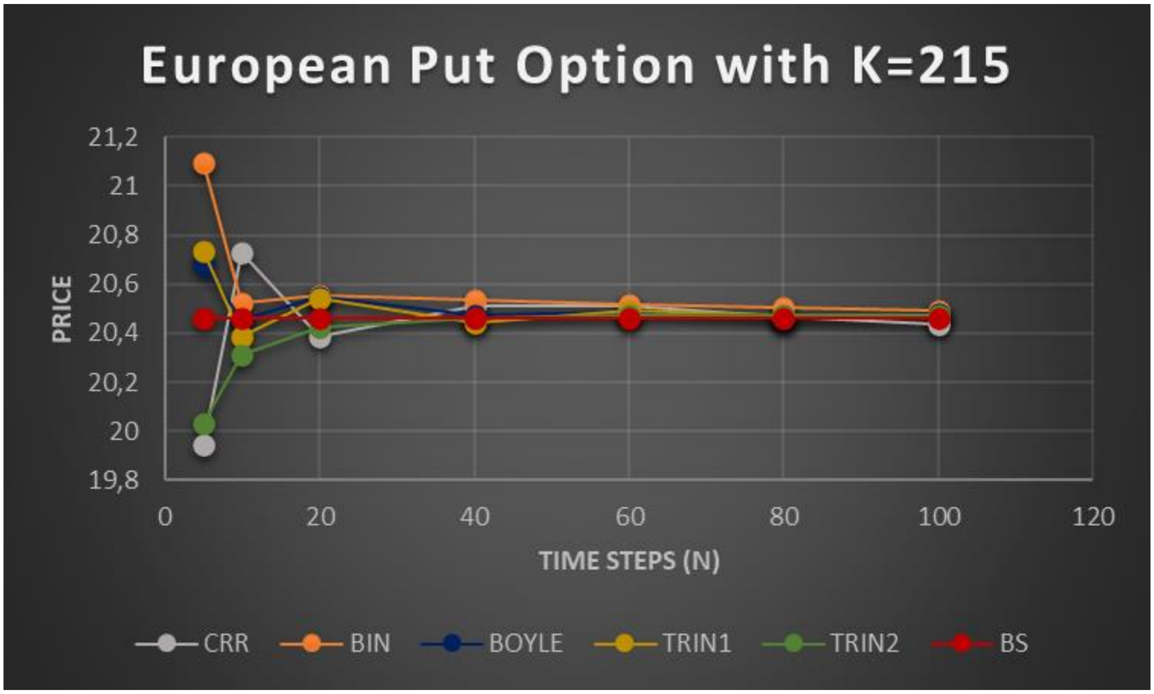
<u>European Put Option Prices with Strike Price=200</u>



Graph 4.5

- We can observe that, Tian Trin1 and Boyle almost concur with Black and Scholes from the beginning, where Time steps are equal to 5.
- From N=40 and thereon all models are close to the Black and Scholes line.

European Put Option Prices with Strike Price=215



Graph 4.6

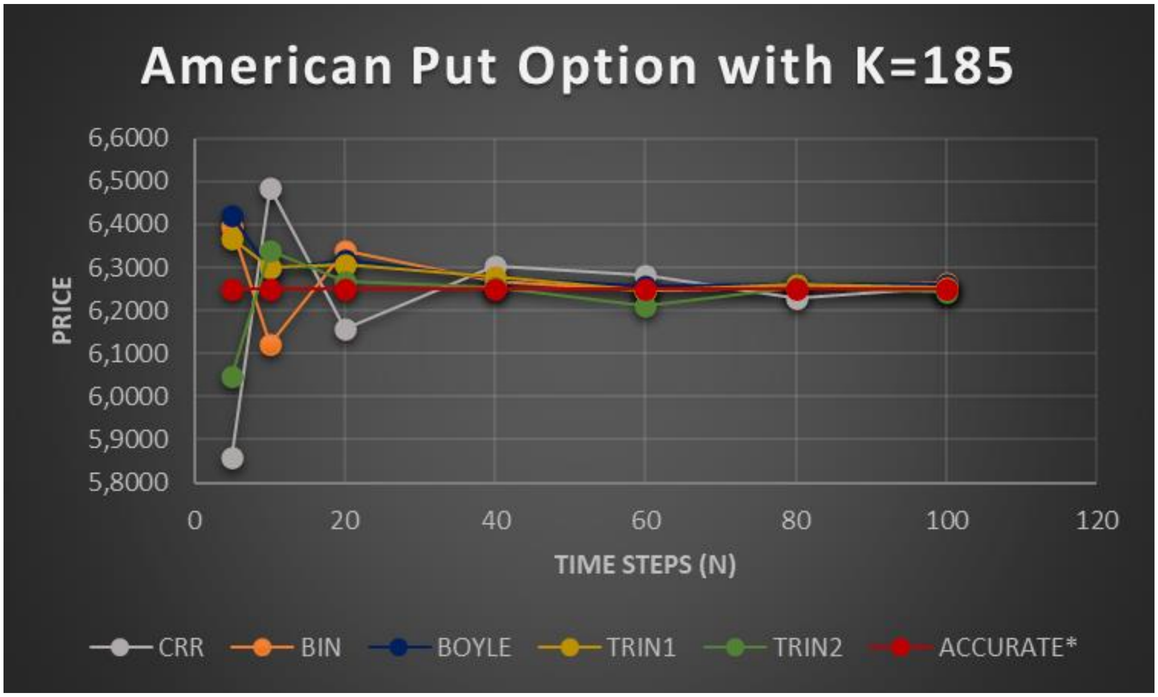
- We can observe that, Tian Trin1 and Boyle almost concur with Black and Scholes from almost the beginning, where Time steps are equal to 10.
- CRR and Tian Bin seem to be the least accurate models.

The valuation of an American put option on a stock that does not pay dividends is very different from the valuation of an American call option on the same stock. Exercising American put options with a positive likelihood before expiration may be the best option. Therefore, there is no shut structured answer to the value of American put options. In order to examine the convergence characteristics of various lattice approaches applied to American put options, the 400-step Tian Trin1 trinomial program is used to calculate the precise price of American put options. The outcomes are listed in Table 4.3. Comparative combination properties are displayed by these outcomes. Using 100 time-steps, the prices reported by all methods are within 2 cents of the "accurate" price of $21,2049 to $6.2493. However, in order to further study the comparative accuracy of the five lattice processes, including three modified versions and the CRR and Boyle processes, it is necessary to accurately point out the meaning of "precision". The precision of a guess strategy is estimated by the base number of steps expected to accomplish a given accuracy level. For example, in case it is necessitated that the guess mistake be under 5 cents, and in case it is discovered that estimate technique (X) needs no less than 70 stages to accomplish a particularly required precision, contrasted with 50 stages for guess strategy (Y), then, at that point technique (Y) is considered more exact than (X).

Time Steps (N)	CRR	BIN	BOYLE	TRIN1	TRIN2	ACCURATE*
Exercise Price, K=185						
5	5,8595	6,3973	6,4214	6,3690	6,0478	6,2493
10	6,4834	6,1234	6,2947	6,3001	6,3376	6,2493
20	6,1591	6,3390	6,3176	6,3069	6,2677	6,2493
40	6,3052	6,2689	6,2764	6,2788	6,2522	6,2493
60	6,2825	6,2476	6,2591	6,2472	6,2101	6,2493
80	6,2279	6,2533	6,2601	6,2624	6,2525	6,2493
100	6,2529	6,2628	6,2599	6,2546	6,2432	6,2493
Exercise Price, K=200						
5	13,0826	12,5865	12,4185	12,4153	12,6284	12,4137
10	12,2087	12,6940	12,4060	12,4112	12,2823	12,4137
20	12,3110	12,5449	12,4089	12,4199	12,3605	12,4137
40	12,3606	12,3926	12,4099	12,4221	12,4202	12,4137
60	12,3767	12,3973	12,4099	12,4209	12,3729	12,4137
80	12,3850	12,4312	12,4098	12,4202	12,4182	12,4137
100	12,3900	12,4378	12,4097	12,4194	12,4135	12,4137
Exercise Price, K=215						
5	20,8749	21,6829	21,3432	21,3740	20,7194	21,2049
10	21,4292	21,3290	21,2383	21,1769	20,9800	21,2049
20	21,1536	21,3055	21,2787	21,2755	21,1020	21,2049
40	21,2617	21,2721	21,2236	21,1972	21,1730	21,2049
60	21,2480	21,2510	21,2258	21,2277	21,1881	21,2049
80	21,2076	21,2380	21,2054	21,2113	21,2003	21,2049
100	21,1930	21,2284	21,2180	21,2087	21,2015	21,2049

Table 4.3 – American Put prices

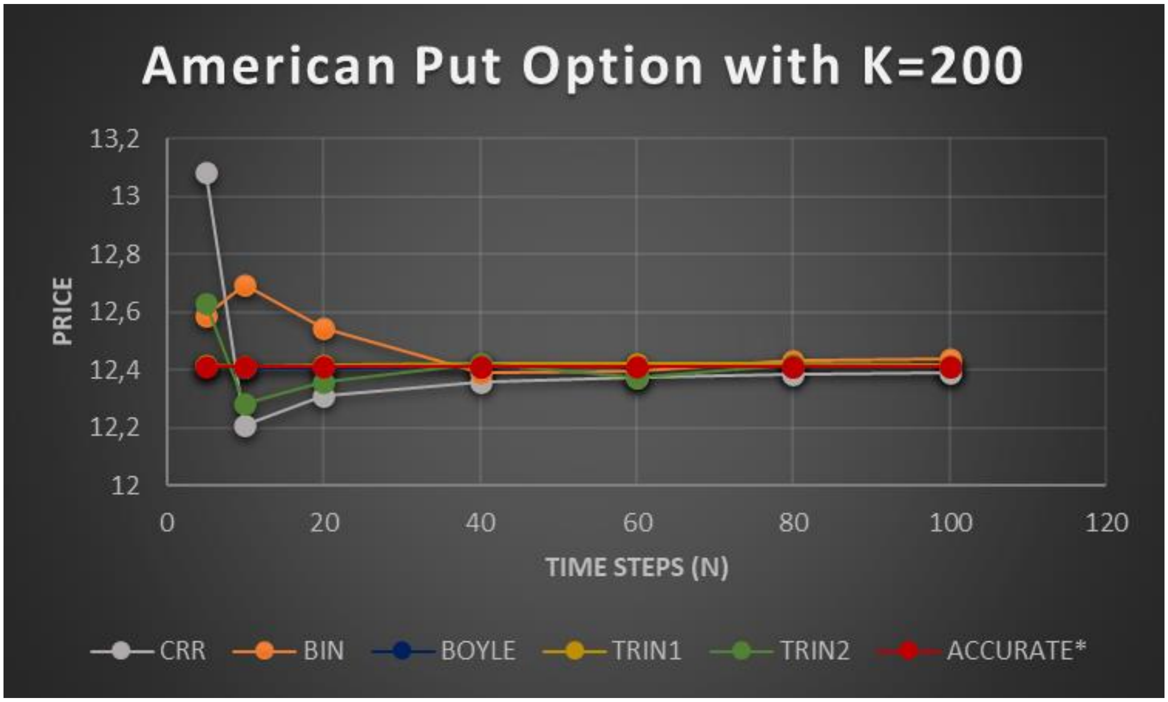
<u>American Put Option Prices with Strike Price=185</u>



Graph 4.7

- We can observe that, Tian Trin1 and Boyle almost concur with Black and Scholes from almost the beginning, where Time steps are equal to 10.
- CRR and Tian Trin2 seem to be the least accurate models.

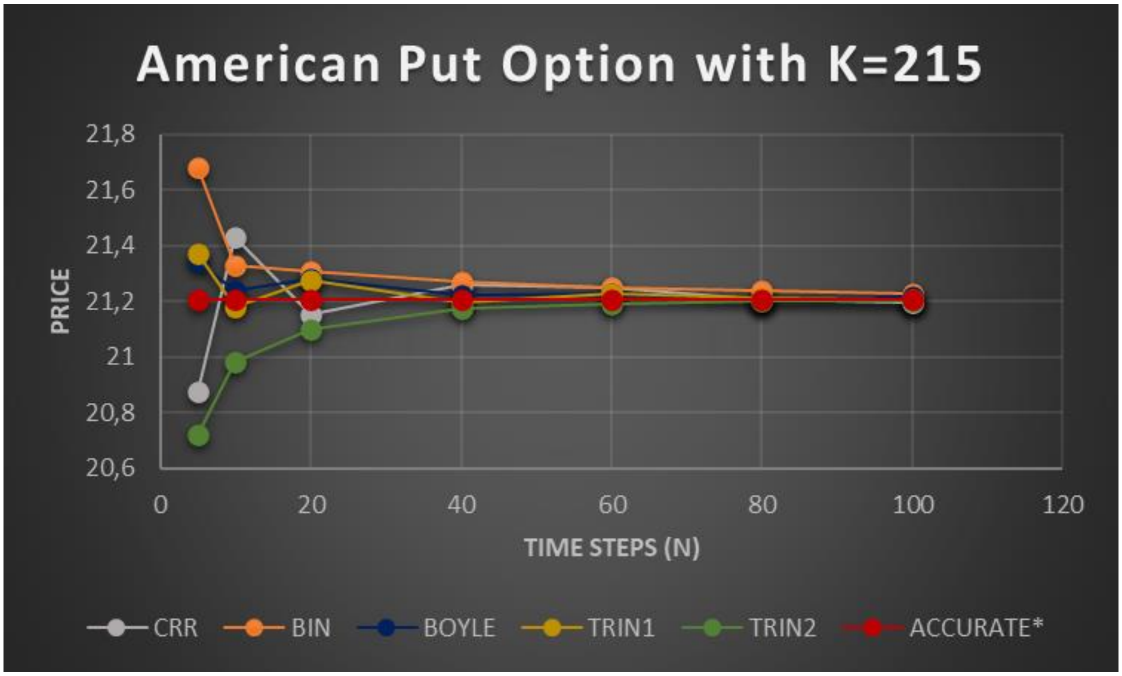
<u>American Put Option Prices with Strike Price=200</u>



Graph 4.8

- We can observe that, Tian Trin1 and Boyle concur with Black and Scholes from the beginning, where Time steps are equal to 5.
- CRR seems to be the least accurate model.

American Put Option Prices with Strike Price=215



Graph 4.9

- We can observe that, Tian Trin1 and Boyle almost concur with Black and Scholes from the beginning, where Time steps are equal to 5.
- From N=40 and thereon all models are close to the Black and Scholes line.

Accuracy examination

Officially, the precision of an estimate technique might be characterized as follows: Consider a boundless sequence {P(n)}, which merges to a positive breaking point P^*. Given an accuracy level, i.e., a small positive number ε, characterize a positive integer N_ε,

$$N_\varepsilon = Min\left\{ N : \left| \frac{P(n) - P^*}{P^*} \right| < \varepsilon, \qquad for\ all\ n \geq N \right\} \qquad (4.1)$$

This number N, is alluded to as the minimum convergence step (MCS) comparative with ε. In case P^* is the exact cost of an options, and P(n) is the cost of the choice acquired from a n-step estimate technique, then, at that point the MCS (i.e., N,) of the value arrangement estimates the exactness of the guess strategy at the ε accuracy level. The more modest the MCS, the more precise the method.

This meaning of exactness accentuates the significance of the steadiness of assembly. It distinguishes the base number of steps that are expected to guarantee that the overall estimation blunder be not exactly the necessary accuracy level for every resulting step. The more modest the MCS, the more exact the estimation. A model might assist with representing this idea. Consider two groupings: $\left\{\alpha_n = 1 - \frac{1}{2^n}\right\}$ *and* $\left\{b_n = 1 - \frac{1}{n}\right\}$. Both groupings converge to 1. Unmistakably the grouping $\{a_n\}$ meets to 1 a lot quicker than the other grouping. To apply the idea of MCS to those two groupings, consider ε = 0.05. The MCS is 5 for the grouping $\{a_n\}$ and 21 for the grouping $\{b_n\}$.

At this juncture, the idea of least assembly step is applied to research the near exactness of the five cross section techniques considered previously.' The option boundary esteems displayed in Table 4.1 are utilized for the investigation underneath. The worth of an option is determined for every grid technique utilizing time steps going from 1, 2, 3, … , 400. By contrasting these option prices and the exact worth, the MCS is then gotten given an exactness level. Call options on a share that pays no dividend are analyzed first. A nearby structure answer for the worth of such an option is accessible and it is given by the notable Black-Scholes equation. Table 4.4 reports the MCS esteems at three exactness levels, 5%, 1% and 0.5%.

European call option on share that pays no dividend

Exercise Price (K)	Numerical Method				
	CRR	BIN	BOYLE	TRIN1	TRIN2
Precision level, e=5%					
185	2	1	2	2	2
200	5	2	2	1	1
215	7	6	4	4	5
Average	4,7	3,0	2,7	2,3	2,7
Precision level, e=1%					
185	9	9	5	5	5
200	22	19	3	3	16
215	30	18	7	7	13
Average	20,3	15,3	5,0	5,0	11,3
Precision level, e=0,5%					
185	22	18	6	5	16
200	43	25	7	3	18
215	61	71	26	29	19
Average	42,0	38,0	13,0	12,3	17,7

Table 4.4 – The accuracy of the proposed methods as compared to the CRR binomial and the Boyle trinomial – Call option on share that pays no dividend

At the 5% exactness level, all methods are extremely precise. Tian Trin1 presents the lowest average MCS of 2.3, while both CRR and Tian Bin present the highest average MCS of 4.7 and 3.0, individually.

At the 1% exactness level, Boyle and Tian Trin1 are the most accurate methods and present the lowest average MCS of 5.0. CRR is the least accurate and presents an average MCS of 20.3, while Tian Bin is more accurate with an average MCS of 15.3.

At the 0.5% exactness level, Tian Trin1 is the most accurate and presents the lowest average MCS of 12.3. Boyle is close enough with an identical average MCS of 13.0, while Tian Trin2 is less accurate with an average MCS of 17.7. However, Tian Trin2 in the OTM case with a strike price at 215 is the most accurate with a MCS of 19 against Tian Trin1 which although overall is the best model, in the OTM has a MCS of 29. CRR is the least accurate and presents the highest average MCS of 42.0, compared to a value of 38.0 presented by the Tian Bin method.

Summing up the outcomes at all three exactness levels, Boyle and Tian Trin1 are the most precise and very little discrepancies are seen between them. CRR is the most un-precise strategy and is fundamentally less exact than Tian Bin. It is additionally certain that all trinomial techniques are substantially more precise than binomial strategies.

<u>European put option on share that pays no dividend</u>

Exercise Price (K)	Numerical Method				
	CRR	BIN	BOYLE	TRIN1	TRIN2
Precision level, e=5%					
185	9	9	5	4	5
200	6	2	2	2	4
215	2	4	2	2	3
Average	5,7	5,0	3,0	2,7	4,0
Precision level, e=1%					
185	45	48	22	20	34
200	29	21	5	3	17
215	11	12	6	6	9
Average	28,3	27,0	11,0	9,7	20,0
Precision level, e=0,5%					
185	88	94	50	47	92
200	59	52	9	5	18
215	28	18	7	7	13
Average	58,3	54,7	22,0	19,7	41,0

Table 4.5 – The accuracy of the proposed methods as compared to the CRR binomial and the Boyle trinomial – Put option on share that pays no dividend

At the 5% exactness level, Tian Trin1 is the most precise, with the least presented average MCS of 2.7, while Boyle and Tian Trin2 are marginally less precise presenting an average MCS of 3.0 and 4.0, respectively. CRR is the least accurate and presents the highest average MCS of 5.7, while Tian Bin is more accurate presenting an average MCS of 5.0.

At the 1% exactness level, Tian Trin1 is again the most accurate, with the lowest reported average MCS of 9.7, while Boyle's accuracy is slightly lower with an average MCS of 11.0. The accuracy of Tian Trin2 is much lower with an average MCS of 20.0. CRR is the least accurate with the highest average MCS of 28.3, while Tian Bin's average MCS of 27,0, which is not much better.

At the 0.5% exactness level, Tian Trin1 is still the most accurate with the lowest average MCS of 19.7, although Boyle is only slightly worse with an average MCS of 22.0. On the other hand, the precision of Tin Trin2 is much lower with an average MCS of 41.0. CRR is the least accurate with the highest average MCS of 58.3, while Tian Bin's performance is better, with an average MCS of 54.7.

Summing up the aftereffects of the European put options, Boyle and Tian Trin1 are the most precise with practically indistinguishable MCSs, while CRR is the most un-exact with the most elevated averages MCSs. Tian Bin and Tian Trin2 are mediocre.

<u>American put option on share that pays no dividend</u>

Exercise Price (K)	Numerical Method				
	CRR	BIN	BOYLE	TRIN1	TRIN2
Precision level, e=5%					
185	9	9	4	4	5
200	6	2	2	1	1
215	3	2	2	1	3
Average	6,0	4,3	2,7	2,0	3,0
Precision level, e=1%					
185	32	33	21	20	35
200	26	21	2	2	17
215	11	8	5	5	11
Average	23,0	20,7	9,3	9,0	21,0
Precision level, e=0,5%					
185	61	94	25	23	92
200	52	27	3	3	20
215	26	19	6	7	20
Average	46,3	46,7	11,3	11,0	44,0

Table 4.6 – The accuracy of the proposed methods as compared to the CRR binomial and the Boyle trinomial – American put option on share that pays no dividend

At the 5% exactness level, all strategies are exceptionally precise. Tian Trin1 presents the lowest average MCS of 2.0, while both CRR and Tian Bin present the most elevated average MCS of 6.0 and 4.3, respectively.

At the 1% exactness level, Tian Trin1 and Boyle are the most exact strategies and present the lowest averages MCS of 9.0 and 9.3 , respectively. CRR is most un-precise and presents an average MCS of 23.0,while Tian Bin is more exact with an average MCS of 20.7.

At the 0.5% exactness level, Tian Trin1 is the most exact and presents the lowest average MCS of 11.0. Boyle is close enough with an indistinguishable average MCS of 11.3, while

Tian Trin2 is less exact with an average MCS of 44.0. CRR and Tian Bin are the most un-precise and present the most noteworthy average MCS of 46.3 and 46,7, respectively.

Summing up the outcomes at all three exactness levels, Boyle and Tian Trin1 are the most precise and very little discrepancies are noted between them. CRR is the most un-exact strategy and is significantly less exact than Tian Bin in the ATM and ITM cases per accuracy level e=0.5%. It is additionally evident that all trinomial techniques are substantially more exact than binomial strategies. Of the trinomial techniques, Tian Trin1 and Boyle are the most exact and report practically indistinguishable MCSs. Tian Trin2 is less precise than its trinomial partners.

In synopsis, apparently all trinomial strategies are more exact than binomial models. Albeit a N-step trinomial strategy includes a larger number of calculations than a N-step binomial method, the huge improvement in exactness appears to legitimize the extra calculations that are required on a money saving advantage premise. All in all, to accomplish a given degree of accuracy, the trinomial process appears to require less calculations than its binomial partner. For example, to get the cost of a call option on a share that pays no dividend permitting a blunder of 0.5%, a 12-step Tian Trin1 trinomial strategy is for the most part required which requires checking 240 hubs and assessing an articulation at every hub, while a 38-step Tian Bin binomial technique is by and large required which requires checking 742 hubs and assessing an articulation at every hub. In this manner, the trinomial method seems, by all accounts, to be more cost efficient than the binomial methodology. Besides, of the two binomial techniques, Tian Bin is more precise when applied to call options and American put options. At long last, of the trinomial techniques, Tian Trin1 and Boyle are indistinguishably precise while Tian Trin2 is less exact.

4.2 Empirical Study

In this section we will experientially approach the lattice models of CRR (1979) & Boyle (1988) and Tian (1993) elaborated in the preceding chapters and compare them in terms of their effectiveness. More specifically:

- Collect our data from the source mentioned below and process them appropriately in an excel file.
- Estimate the parameters which have been mentioned before and bring the most accurate pairing of the models' prices with the observed market prices.
- By estimating the parameters, we will check the forecasting ability of the models.

4.2.1 Empirical Study Collected Data

For the purpose of conducting the empirical study, we drew upon data from Thompson Reuters for American-style market options, with the underlying asset being the Apple stock for the 09/07/2020 – 07/05/2021 period that was chosen for the various historical purchase options prices were available at the time.

Consequently, the limited availability of historical prices of market options was a deterrent to the conduct of this empirical study. Even so, the Apple stock was preferred after meeting the following criteria:

- It is listed on the American stock exchange.
- Its stock has a significant market value.
- The volume of securities traded on daily basis is satisfactory.

We consider that Apple stock's dividend distribution policy has no significant impact on the price of the stock. It should be mentioned that the lattice models we demonstrate do not include a dividend, although they can be expanded appropriately.

Furthermore, it is more than crucial to point out that the American type of market options in the underlying asset of a share that pays no dividend, function in an identical way as the European type of market options, regarding the terms of early exercise.

Videlicet, this kind of options is not wised to be exercised before the maturity. Nevertheless, in case of the expectation that the stock is going to pay a dividend, it may be best to exercise the option before maturity. This could be explained by considering that when the dividend is given, the share price shifts downwards and consequently the purchase option will be less appealing since its value has decreased.

In order to prove that, suppose we possess an American type call option, which at some point of time (t_1) before maturity (T) is deep in the money, so the price of the stock is $S_1 > S_0$, where S_0 is the stock price at time zero when we purchased the call option. Therefore, the value of the option at t_1 will be C_1. We know that $C_1 \geq c_1$, where c_1 is the value of the respective European type call option. Additionally, we are familiar with the following condition:

$$C_1 \geq c_1 \geq S_{t_1} - Ke^{-r(T-t_1)} > S_{t_1} - K$$

As a result, at that point in time $(t_1 < T)$, the holder of the option owns an asset with a larger value than the one that its early exercise would bring him.

For the purpose of the specific study, since we assume that the dividend has a negligible impact on the share value, we will treat the American type call option of American type as a European type call option. A European type option was not preferred in the first place, as the market considers them platitude and either do not exist or are not traded and consequently we do not have an adequate quantity of historical data.

For the prosecution of the experiential examination, we needed historical values of the following data:

- The values of the call options for different maturities and for different exercise prices.
- The exercise prices for the above call options.
- The daily stock price of the Apple stock.
- The implied volatility of every call option.
- The yield on the American 10 years Treasury Note.

The data we used refer only to call options to conserve time for the execution of the empirical study. Specifically, 12 call options were used for each day and the data implied was for 4 specific maturities and 3 exercise prices. The options tickers acquired from Thomson Reuters Database are depicted bellow:

Tickers		
AAPL US 6/18/21 C120 Equity	AAPL US 6/18/21 C130 Equity	AAPL US 6/18/21 C140 Equity
AAPL US 9/17/21 C120 Equity	AAPL US 9/17/21 C130 Equity	AAPL US 9/17/21 C140 Equity
AAPL US 1/21/22 C120 Equity	AAPL US 1/21/22 C130 Equity	AAPL US 1/21/22 C140 Equity
AAPL US 6/17/22 C120 Equity	AAPL US 6/17/22 C130 Equity	AAPL US 6/17/22 C140 Equity

4.2.2 Parameters Estimation

Broadly speaking, from the observations of the sample, we can compute the point estimate of the parameter of a random variable. The point estimate of a parameter is the statistic that we compute from the sample. It is a value calculated based on the sample data and represents the actual value of the population relative parameter.

In the models we are investigating, the volatility σ of the share price or the return on the share is an unknown parameter, so this parameter must be evaluated. For this process we use the iterative algorithm Levenberg Marquardt (1944 and 1963), that assumes that we have a model and a set of parameters to estimate.

The algorithm detects those values of the parameters for which the squares of the differences between the values of the options derived from the model (theoretical values) and the values of the market options (actual values) have the lowest possible value. The procedure of minimizing square errors is illustrated by the following relationship:

$$\boxed{\theta = \arg\min \sum_{i=0}^{N} \left(f_i^{market} - f_i^{model} \right)^2} \qquad (4.2)$$

In which state θ is the estimated parameter for σ and N is the number of days the sample observations were made (in sample).

The Levenberg-Marquardt algorithm was developed in the early 1960s to solve nonlinear least squares problems (least square curve fitting problems). Fewer least squares problems occur when a parameter α is placed in a data sample by minimizing the sum of the squares of the errors between the sample data and the function. If the fitting function in the parameters is non-linear, the least squares problem is non-linear. The nonlinear least squares method repeatedly reduces the sum of squares of the errors between the function and the data sample through a series of parameter value updates.

Suppose we have N observations y_i, where $i = 1, 2, ..., N$ and a function : $g : R^n \to R$ with parameters $x_1, x_2, ..., x_n$, where $n \leq N$. In this case y_i are the allocation prices as we collected them from the market. Therefore, we calculate the model values $g(x) = y_i$ and then compute the residuals $n(x) = y_i - y_i$. Therefore, we have calculated a N dimensions vector which contains the residuals $R = (r_1, r_2, ..., r_n)^T$. So, the following minimization problem needs to be solved:

$$\boxed{\min_x f(x) = \frac{1}{2} \sum_{i=1}^{N} r_i(x)^2 = \frac{1}{2} R(x)^T R(x)} \qquad (4.3)$$

Levenberg and Marquardt (1944 and 1963) in order to find the answer to the previous problem, proposed to use an iterative algorithm that combines the Newton-Gauss method and the Steepest Descent method (expansion of the Laplace method to approximate an integral).

In his article (1944) Levenberg suggests the calculation of a d_k search direction, as a solution to the Newton-Gauss equation:

$$\boxed{\left(R^{'}(x^{k})R^{'}(x^{k})+\lambda_{k}I\right)d_{k}=-\left(R^{'}(x^{k})^{T}R^{'}(x^{k})\right)} \qquad (4.4)$$

Where, I is the unitary matrix and λ_k is an amortization parameter with $\lambda_k > 0$. The table on the left of the equation is defined positively, so the solution d_k gives a fair direction for the function f for all positive amortization parameters.

For small λ_k, the iterative algorithm Levenberg - Marquardt (1944 & 1963) is reminiscent of the iterative Newton-Gauss method and shows a quadratic rate of convergence for x^k values that are close to x^*.

For repetitions that are not optimal, the amortization parameter is too large, and the d_k search direction is approximately:

$$\boxed{d_{k}=\frac{1}{\lambda_{k}}R^{'}(x^{k})^{T}R^{'}(x^{k})} \qquad (4.5)$$

The above relationship is a step of the Steepest Descent method. The selection of the amortization parameter directly affects the stability of the procedure. The most common choice is:

$$\boxed{\lambda_{0}=\tau\max_{i}\left\{D_{0}(i,i)\right\}_{i=1,2,\dots,n}} \qquad (4.6)$$

where τ is the parameter related to the initial prognostication for the estimated parameter.

In our analysis we find the prices of 12 call options for a period starting at 09.07.2020 and ending at 23.04.2021 and we apply the iterative algorithm Levenberg - Marquardt (1944 and 1963) using the programming language MATLAB and more specifically the command Lsqnonlin. We give an initial value to the parameters that we want to estimate. The value we think is most likely is usually used so that when we run the algorithm on the model, we get a theoretical value with the least possible error relative to the actual value.

Therefore, we first estimate the unknown parameters for all call options (in sample). Then we calculate the theoretical prices of the call options that we have according to the models and the estimated parameters as they turned out for each day. The theoretical values were calculated to represent the absolute error, which, when combined with the squared residuals, is useful for drawing conclusions and is expressed as follows:

$$\boxed{\textit{absolute error} = |\textit{market price} - \textit{model price}|} \qquad (4.7)$$

4.2.3 Efficiency study of the models (In the sample)

<u>CRR results</u>

In the CRR model there is one parameter to be estimated, the stock price volatility σ. The accompanying measurements are given in Table 4.7. The average value for the daily estimates we made for σ is 0.45423, with a standard deviation of 0.27965. The minimum and maximum of the parameter sigma are 0.28705 and 1.69545, respectively.

<u>Tian Bin results</u>

In the Tian Bin model there is one parameter to be estimated, the stock price volatility σ. The accompanying measurements are given in Table 4.7. The average value for the daily estimates we made for σ is 0.45512, with a standard deviation of 0.28238. The minimum and maximum of the parameter sigma are 0.28727 and 1.70699, respectively.

<u>Boyle results</u>

In the Boyle model there are two parameters to be considered, the stock price volatility σ and the parameter λ used to bounce u of the stock price. The accompanying measurements are given in Table 4.7. The average value for the daily estimates we made for σ and λ are 0.45380 and 1.48310 respectively, with a standard deviation of 0.27736 and 0.48672 respectively. The minimum and maximum of the two parameters are 0.28692 and 0.99353 for the minimum, while 1.68156 and 3.5 for the maximum, for each parameter.

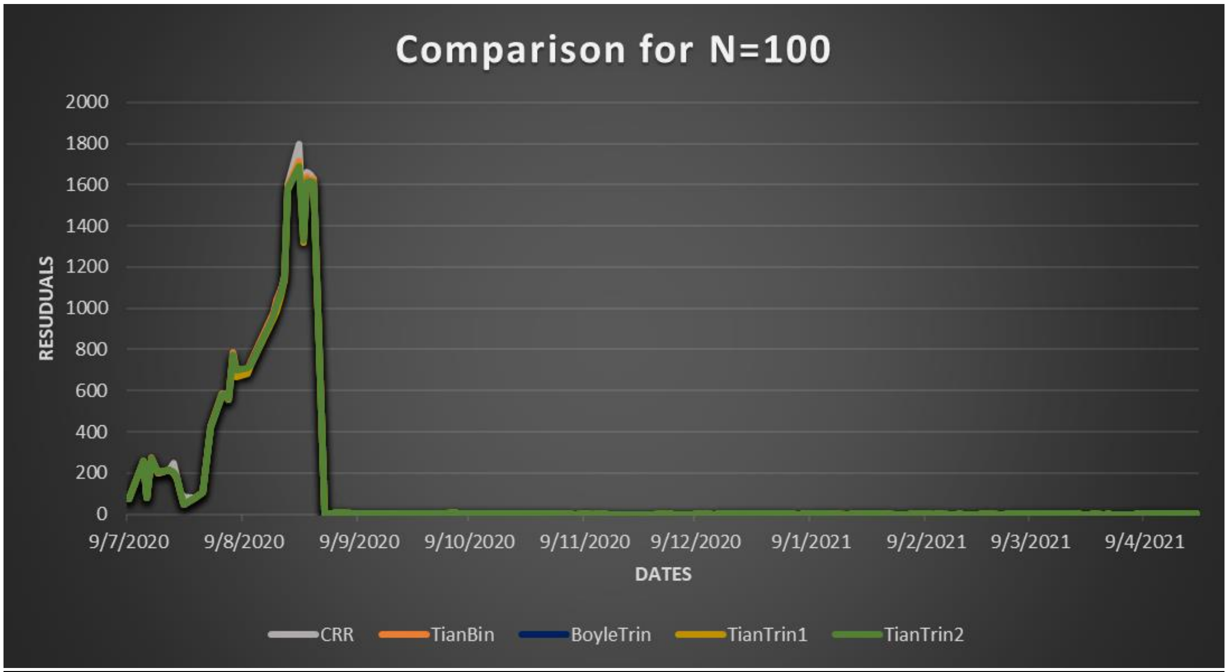
<u>Tian Trin1 results</u>

In the Tian Trin1 model there is one parameter to be estimated, the stock price volatility σ. The accompanying measurements are given in Table 4.7. The average value for the daily estimates we made for σ is 0.45317, with a standard deviation of 0.27624. The minimum and maximum of the parameter sigma are 0.28702 and 1.67333, respectively.

<u>Tian Trin2 results</u>

In the Tian Trin2 model there is one parameter to be estimated, the stock price volatility σ. The accompanying measurements are given in Table 4.7. The average value for the daily estimates we made for σ is 0.45449, with a standard deviation of 0.27940. The minimum and maximum of the parameter sigma are 0.28731 and 1.67110, respectively.

<u>Total model results</u>



Graph 4.10

- We observe that residuals are negligible, except for the period between 09/08/2020 to 09/09/2020.
- All model residuals follow an almost identical path for N=100.

Parameter estimation results											
	CRR		Tian Bin		Boyle Trin			Tian Trin1		Tian Trin 2	
N=100	Sigma	Residuals	Sigma	Residuals	Sigma	Lamda	Residuals	Sigma	Residuals	Sigma	Residuals
Average	0,45423	110,53623	0,45512	109,58794	0,45380	1,48310	107,84297	0,45317	107,59111	0,45449	108,64079
Standard deviation	0,27965	338,00829	0,28238	333,88695	0,27736	0,48672	329,16719	0,27624	328,54055	0,27940	330,71367
Minimum	0,28705	0,04423	0,28727	0,06705	0,28692	0,99353	0,04319	0,28702	0,05699	0,28731	0,04763
Maximum	1,69545	1797,32847	1,70699	1713,78349	1,68156	3,50000	1667,79431	1,67333	1699,11466	1,67110	1690,69039

Table 4.7

Regarding the call options, we have utilized and for the specific time of period we have collected our data, based on the Table 4.7, we observe that residuals do not differ significantly because of the expansion of the quantity of time steps N. Having considered the 12 options for valuation, whose market prices are affected by different parameters that our models doesn't consider, like the interest for an option at a specific time, it makes sense to have such discrepancies. Also, we should point out that the results from the numerical accuracy are confirmed here, as we can state that Boyle and Tian Trin1 are the best models with residuals being 107,84297 and 107,59111 for N=100. Tian Trin2 is less accurate with residuals being 108,64079 for N=100. The least accurate model is CRR with residuals being 110,53623 for N=100. Summarizing the results for 100 N-steps level, Boyle and Tian Trin1 are the most accurate and very little discrepancies are observed between them. CRR is the least accurate method. It is also clear that all trinomial methods are much more accurate than binomial methods.

4.2.4 Efficiency study of the models (Out of sample)

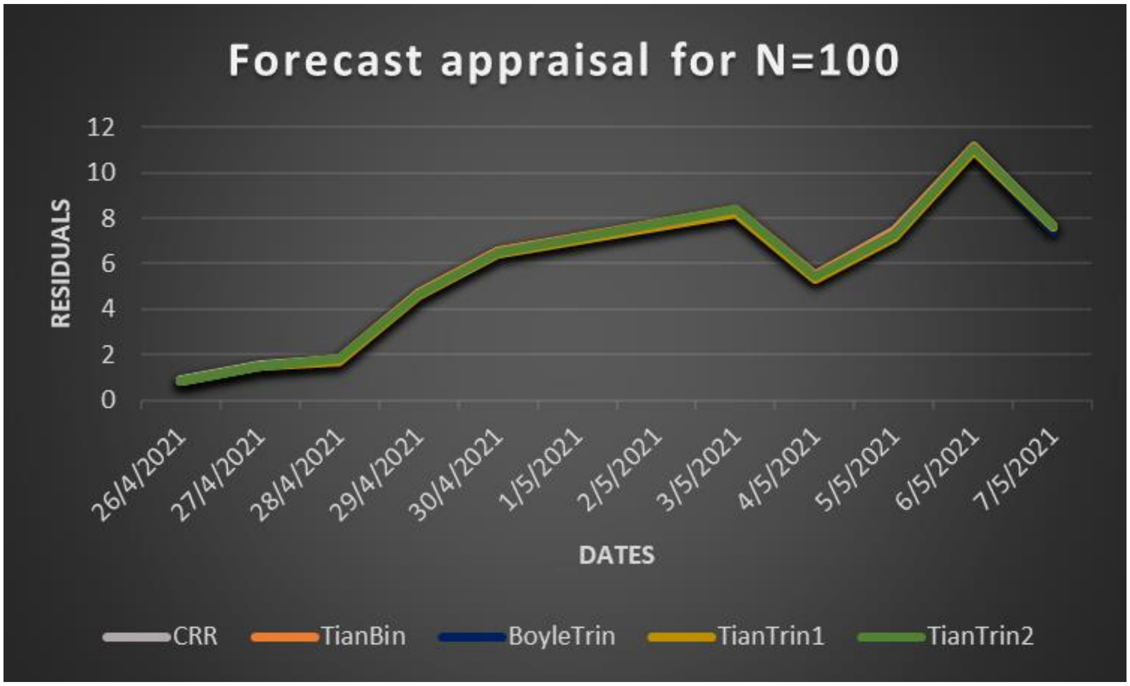
After completing the estimation of the parameters of our models, the assessment of their forecasting capacity follows. That is, we will use the average values of the parameters we estimated from 21/04/2021 to 23/04/2021 (estimates of the last 3 days in Sample) to calculate the theoretical value of the call options according to our models for the period from 26/04/2021 to 07/05/2021 (Out of Sample) with a time-step of N=100.

The theoretical values of the call options were used to calculate the absolute forecast error, in order to obtain the square residuals. The absolute error is expressed as follows:

$$\boxed{absolute\ error = \left| market\ price - \mathrm{mod}\,el\ \ forecast\ price \right|} \qquad (4.8)$$

The fact that the specific models manage to describe the contract price in the sample does not mean that this should also apply to prices outside the valuation sample. For this reason, we will control their behavior at out of sample prices.

Forecasting capability for all options’ maturities, N=100



Graph 4.11

- We observe that there are very small differences between the models. Therefore, we are going to investigate where those differences came from.
- The residuals between the prices we calculated are small in the first days, following the price of the residuals of the last valuation day (In Sample) and gradually increasing.

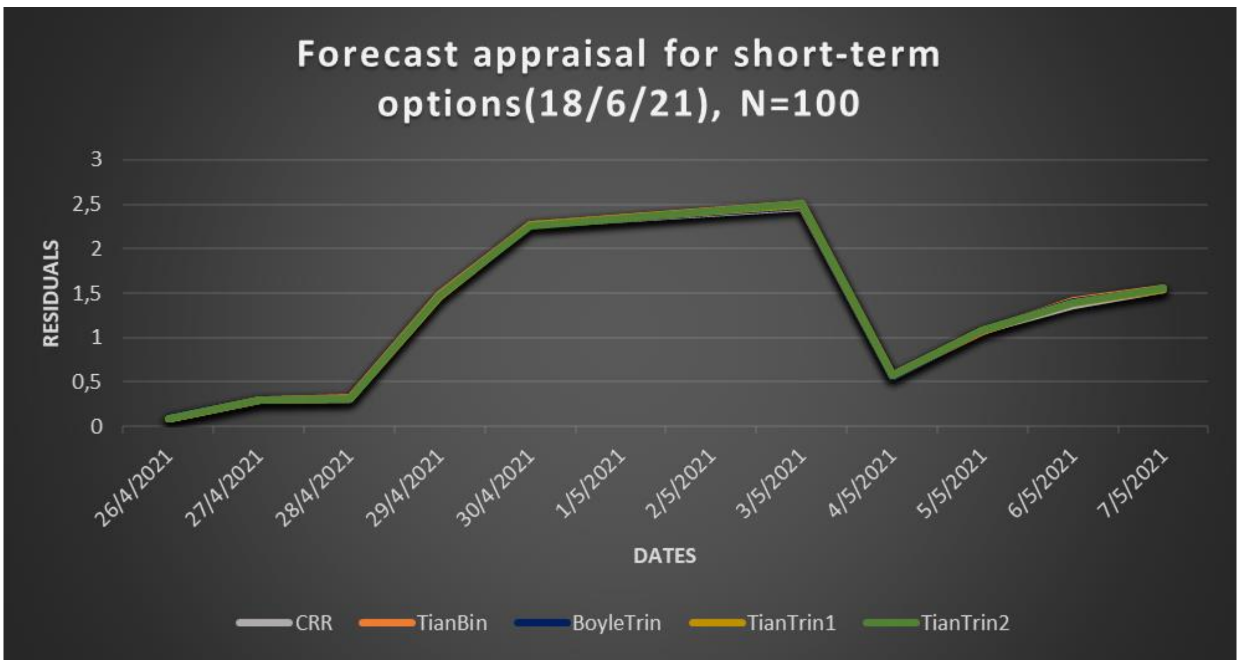
Forecasting Ability					
	CRR	TianBin	BoyleTrin	TianTrin1	TianTrin2
N=100	Residuals	Residuals	Residuals	Residuals	Residuals
26/4/2021	0,91648	0,85697	0,83383	0,83544	0,85838
27/4/2021	1,57374	1,50388	1,48842	1,49445	1,50696
28/4/2021	1,83775	1,79981	1,74231	1,67438	1,80181
29/4/2021	4,68932	4,70348	4,60280	4,61506	4,65240
30/4/2021	6,55025	6,55531	6,45457	6,45619	6,52103
3/5/2021	8,27857	8,38329	8,18511	8,21274	8,37200
4/5/2021	5,49263	5,51965	5,35562	5,31873	5,44539
5/5/2021	7,46097	7,44161	7,25848	7,12760	7,31188
6/5/2021	11,14453	11,15914	10,96190	10,93996	11,10249
7/5/2021	7,68806	7,63319	7,42020	7,59234	7,69076
Average	5,56323	5,55563	5,43032	5,42669	5,52631

Table 4.8

Summarizing the results that are presented at table 4.8, Tian Trin1 and Boyle are the most accurate and very little differences are observed between them, with average residuals of 5.42669 and 5.43032, respectively. CRR is the least accurate method with an average residual of 5.56323 and is less accurate than BIN which has average residuals of 5.55563 . It is also clear that all trinomial methods are much more accurate than binomial methods.

Afterwards we examine the forecasting capability by segregating the options in three different periods, the short-term for the ones that are expiring at 18/06/2021, the mid-term for the ones that are expiring at 17/09/2021 and at 21/01/2022 and the long-term for the options that are expiring at 17/06/2022.

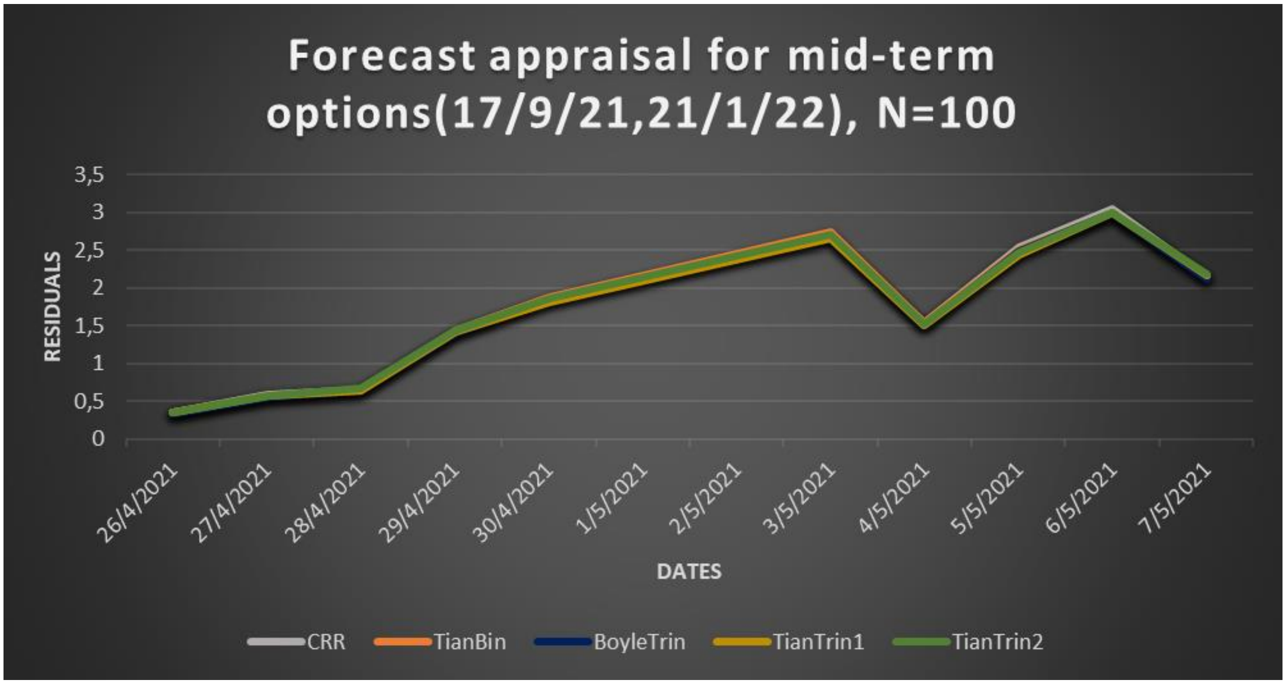
<u>Forecasting capability for short-term options (expiring at 18/06/2021), N=100</u>



Graph 4.12

- The one-month options seem to follow the same route.
- The residuals between the prices we calculated are small in the first days, following the price of the residuals of the last valuation day (In Sample) and gradually increasing.

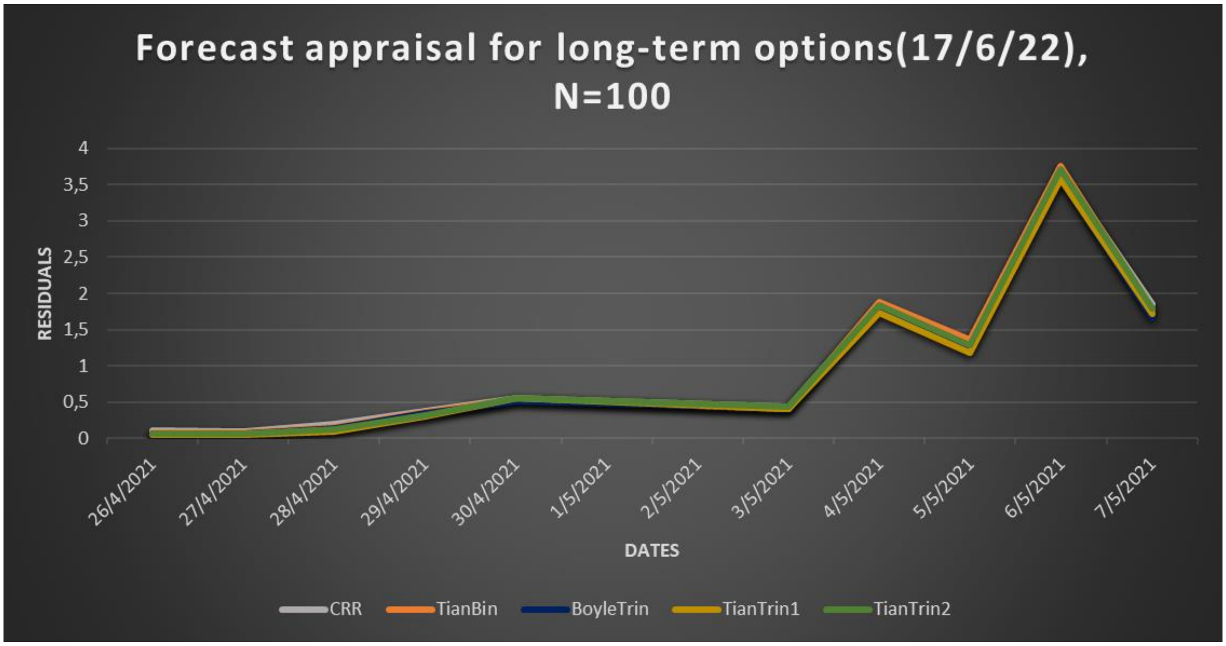
<u>Forecasting capability for mid-term options (expiring at 17/09/2021 & 21/01/2022), N=100</u>



Graph 4.13

- The mid-term options as we approach the 10th day appear the same behavior and it is impossible to draw a conclusion based on the graph.
- The residuals between the prices we calculated are small in the first days, following the price of the residuals of the last valuation day (In Sample) and gradually increasing.

<u>Forecasting capability for long-term options (expiring at 17/06/2022), N=100</u>



Graph 4.14

- The twelve-month options seemed to have differences between the different forecasting models, however those discrepancies are also low.
- The residuals between the prices we calculated are small in the first days, following the price of the residuals of the last valuation day (In Sample) and gradually increasing.

Forecasting Ability			
	Short-term options	Mid-term options	Long-term options
N=100	Average Residuals	Average Residuals	Average Residuals
CRR	1,14407	1,68552	1,04813
BIN	1,15633	1,67972	1,03986
BOYLE	1,14755	1,65138	0,98002
TRIN1	1,15080	1,65323	0,96943
TRIN2	1,15318	1,67668	1,01977

Table 4.9

In order to draw clear conclusions, it is necessary to look at our numerical results that are depicted at table 4.9.

Regarding short-term options we observe that the model with the best forecasting ability is the CRR with average residuals of 1.14407, while the Tian Bin model is the one with the worst predictability with average residuals being 1.15633.

As concerns the mid-term options, Boyle seems to have the best accuracy with average residuals being 1.65138, while Tian Trin1 is lightly less accurate with average residuals of 1.65323. Whereas the CRR is the least accurate with average residuals being 1.68552.

Finally, regarding long-term options, Tian Trin1 is the most accurate with the lowest average residuals of 0.96943, although Boyle is slightly worse with an average residual of 0.98002. CRR is the least accurate with the highest average residual of 1.04813, while Tian Bin performs better with an average residual of 1.03986.

Summarizing the results at all three maturities, Tian Trin1 and Boyle are the most accurate and very little differences are observed between them. CRR is the least accurate method and is less accurate than BIN. It is also clear that all trinomial methods are much more accurate than binomial methods, apart from the short-term period accuracy level, where the result should not concern us, because compared to the long-term situation, there is less uncertainty in the short term. Over time, a factor that contains uncertainty is mainly volatility σ. In Boyle's (1986) model and Tian's models, volatility is used repeatedly when calculating jump probability, greatly affecting the theoretical value of options.

CHAPTER 5

CONCLUSIONS

Options are one of the most popular securities transactions on the market. In fact, more and more companies hope to offer more mature financial products to the investing public and are looking for better valuation methods for this purpose. Many models have been used for this purpose. We mentioned five of them, the Cox Ross Rubinstein model (1979), the Boyle Trinomial model (1986), the Tian Modified Binomial model (1993), the Tian Modified Trinomial 1 model (1993) and the Tian Modified Trinomial 2 model (1993).

Therefore, we proceeded to the numerical accuracy analysis for the lattice models of CRR (1979), Boyle (1986) and Tian (1993) for European-type call options, European-type put options and American-type put options. Recognizing that as long as the first two moments of the discrete-time process are correct according to the continuous-time process, that is, geometric Brownian motion, the convergence of the discrete-time lattice model and the continuous-time Black & Scholes model can be ensured. In principle, these two conditions do not provide a single solution for the parameters of the network, including probability and jump. The choice of additional conditions will lead to different parameter values, which in turn leads to different lattice procedures. In this study, the main purpose behind these choices for the modified procedures is that the accuracy of the approximation process is higher than the one in the previous models. Numerical results revealed that the modified binomial model Tian Bin is more accurate than the CRR binomial model when applied to call options and put options on stocks that do not pay dividends. For all three trinomial processes, Tian Trin1 and Boyle have almost the same accuracy, while Tian Trin2 has lower accuracy. In addition, all trinomial methods tested in this study seem to be more accurate than binomial methods. It seems that the best trinomial process is better than the best binomial process based on cost-exactness basis.

The empirical examination that followed assessed an American call option with all five different models. However, because the stock we picked as the underlying title gives a dividend which has a negligible effect on its price, we considered that the stock is treated as though it doesn't give any dividends at all, and accordingly the option functions as a European type call option. Therefore, we evaluated the parameters of each model and determined the order in which the models are ranked based on the evaluation errors. Thus, we confirmed the conclusion we came up, at the preceding numerical accuracy, as the empirical analysis also revealed that Tian Bin is more accurate than the CRR binomial model, Tian Trin1 and Boyle have almost the same accuracy, while Tian Trin2 has lower accuracy. Additionally, all trinomial methods tested in this study seem to be more accurate than binomial methods. Finally, we tested their predictive power to see how the models behave at optional and out-sample prices. That is, how much more accurate can the behavior of our models be in market forecasts and at a great level confirmed the conclusion we previously reached at the numerical accuracy analysis and during the estimation of parameters at the empirical study.

REFERENCES

1. Ahn J. and Song M., Convergence of the trinomial tree method for pricing European/American options, Appl. Math. Comput. 189, 575–582, (2007).
2. Boyle P., "A lattice framework for option pricing with two state variables," The Journal of Financial and Quantitative Analysis, vol. 23, no. 1, pp. 1–12, 1988
3. Boyle P., "Option valuation using a three jump process," International Options Journal, vol. 3, pp. 7–12, (1986).
4. Boyle, P.P., A lattice framework for option pricing with two state variables. Journal of Financial and Quantitative Analysis, 35, 1-12., (1998).
5. Broadie, M., Detemple, J., American option valuation: new bounds, approximations, and a comparison of existing methods. The Review of Financial Studies, volume 9, issue 4, p. 1211 – 1250, (1996).
6. Chang- Palmer, Smooth convergence in the binomial model., (2005).
7. Cox, J. C., Ross, S. A., &Rubinstein, M. Option pricing: A simplified approach. Journal of Financial Economics, 7, 229-264., (1979).
8. Crépey S., Calibration of the local volatility in a trinomial tree using Tikhonov regularization, Inverse Problems 19, pp. 91–127, (2003).
9. Derman E., Kani I. and Chriss N., Implied trinomial trees of the volatility smile, J. Derivative 3, pp. 7–22, (1996).
10. Diener, F., Diener, M., Asymptotic of the Price Oscillations of a European Call Option in a Tree Model-Francine Mathematical Finance, Vol. 14, No. 2, 271-293, (2004).
11. Han X., "Pricing American options based on trinomial tree model," Journal of Times finance, vol. 1, no. 505, p. 214, (2013).
12. Heston, S., & Zhou, G. F., On rate of convergence of discrete-time contingent claims. Working paper, Olin School of Business, Washington University, (1997).
13. Hull, J. C., & White, A., Valuing derivative securities using the explicit finite difference method. Journal of Financial and Quantitative Analysis, 25, 87-100. (1990).
14. Jarrow, R., Rudd, A., Option pricing. Journal of Banking & Finance, vol. 10, issue 1, 157-161., (1986).
15. Joshi, M.S., Achieving Higher Order Convergence for the Prices of European Options in Binomial Trees, University of Melbourne, Centre for Actuarial Studies, (2007).
16. Kou, S.G., On pricing of discrete barrier options. Statistica Sinica 13, 955–964., (2003).
17. Kunito, N. and Ikeda, M. Pricing options with curved boundaries. Math. Finance 2, 275-298., (1992).
18. Leisen, D., & Reimer, M., Binomial models for option valuation: Examining and improving convergence. Discussion paper No. B-309, Rheinische Friedrich-Wilhelms - Universitat Bonn., (1995).

19. Tian Yisong, “A Modified Lattice Approach to Option Pricing”, The Journal of Futures Markets, Vol. 13, pp. 563-577, (1993).
20. Walsh J., The rate of convergence of the binomial tree scheme. Finance and Stochastics, 7, 337–361, (2003).
21. Xiong B. Z. , “A trinomial option pricing model based on Bayesian Markov chain Monte Carlo method,” Journal of Jiaxing University, vol. 24, no. 6, pp. 48–53, (2012).
22. Yuen F. L. and Yang H., “Option pricing with regime switching by trinomial tree method,” Journal of Computational and Applied Mathematics, vol. 233, no. 8, pp. 1821–1833, (2010).

Other notes

1. Englezos Nikolaos, Special Topics of Quantitative Methods in Finance, March 2021.
2. Michail Antropelos, Financial Derivatives and Applications, October 2020.

Books

1. Hull, John C. Options, Futures, and Other Derivatives. 9th edition, Chapter 13.
2. Paolo Brandimarte, Numerical Methods in Finance and Economics: A Matlab-Based Introduction, 2nd Edition, John Wiley & Sons, New York, 2006

Websites

1. mathworks.com
2. investopedia.com

MATLAB ALGORITHMS

1. **Calculation of European Option Payoff**

```
function PayoffValue = calcPayoff(S0, K, CallorPut)
if strcmp(CallorPut, 'put') || strcmp(CallorPut, 'Put')
   PayoffValue = max(K - S0, 0);
else
   PayoffValue = max(S0 - K, 0);
end
end
```

2. **Creation of Binomial/ Trinomial Lattice**

```
function DataStructure = buildLattice(N, BinTrinLattice)
DataStructure = '';
if strcmp(BinTrinLattice, 'trin') || strcmp(BinTrinLattice, 'Trin')
   for i = 1:N
      DataStructure{i} = nan(1,2*i-1);
   end
else
   for i = 1:N
      DataStructure{i} = nan(1,i);
   end
end
end
```

3. **Black-Scholes-Merton (BSM) European Call - Put Option**

```
function OptionPrice = ...
   priceOptionBSM(S0, K, r, T, sigma, CallorPut)
X1=1/(sigma*sqrt(T))*(log(S0/K)...
      + (r +1/2*sigma^2)*T);
X2=X1-sigma*sqrt(T);
OptionPrice=S0*normcdf(X1)...
   -K*exp(-r*T)*normcdf(X2);
if strcmp(CallorPut,'put') || strcmp(CallorPut,'Put')
   % Put-Call Parity
   OptionPrice = K*exp(-r*T)-S0...
            +OptionPrice;
end
end
```

4. Binomial Cox-Ross-Rubinstein (CRR) European Call - Put Option

```
function [OptionPrice, BinLatticeS0, BinLatticeOption] = ...
   priceOptionBinCRR(S0, K, r, T, N, sigma, CallorPut)
dt=T/N;
M=exp(r*dt);
u=exp(sigma*sqrt(dt));
d=1/u;
p=(M-d)/(u-d);
%% Loop over each node of S0 price tree
BinLatticeS0=buildLattice(N+1,'bin');
BinLatticeS0{1}(1)=S0;
for i=2:N+1
   BinLatticeS0{i}(1)=BinLatticeS0{i-1}(1)*u;
   BinLatticeS0{i}(2)=BinLatticeS0{i-1}(1)*d;
   x=length(BinLatticeS0{i});
   if  x>2
      for j=3:x
         BinLatticeS0{i}(j)=BinLatticeS0{i-1}(j-1)*d;
      end
   end
   clear x
end
%% Calculate the value at expiry
BinLatticeOption = buildLattice(N+1, 'bin');
for j=1:length(BinLatticeOption{end})
   BinLatticeOption{end}(j)=calcPayoff(BinLatticeS0{end}(j), ...
      K, CallorPut);
end
%% Loop backwards to get values at the earlier times
for i=N:-1:1
   for j=1:length(BinLatticeOption{i})
      BinLatticeOption{i}(j)= ...
         M^(-1)*(p*BinLatticeOption{i+1}(j) ...
            +(1-p)*BinLatticeOption{i+1}(j+1));
   end
end
OptionPrice=BinLatticeOption{1}(1);
end
```

5. Trinomial Boyle European Call - Put Option

```
function [OptionPrice, TrinLatticeS0, TrinLatticeOption] = ...
                  priceOptionTrinBoyle(S0, K, r, T, N, sigma, CallorPut)
dt=T/N;
M=exp(r*dt);
lamda=1.2;
V=M^2*(exp(sigma^2*dt)-1);
u=exp(lamda*sigma*sqrt(dt));
m=1;
d=exp(-lamda*sigma*sqrt(dt));
pu=((V+M^2-M)*u-(M-1))/((u-1)*(u^2-1));
pd=((V+M^2-M)*u^2-(M-1)*u^3)/((u-1)*(u^2-1));
pm=1-pu-pd;
%% Loop over each node of S0 price tree
TrinLatticeS0=buildLattice(N+1,'trin');
TrinLatticeS0{1}(1)=S0;
for i=2:N+1
    TrinLatticeS0{i}(1)=TrinLatticeS0{i-1}(1)*u;
    TrinLatticeS0{i}(2)=TrinLatticeS0{i-1}(1)*m;
    TrinLatticeS0{i}(3)=TrinLatticeS0{i-1}(1)*d;
    x=length(TrinLatticeS0{i});
    if  x>3
        for j=4:x
            TrinLatticeS0{i}(j)= TrinLatticeS0{i-1}(j-2)*d;
        end
    end
    clear x
end
%% Calculate the value at expiry
TrinLatticeOption=buildLattice(N+1,'trin');
for j=1:length(TrinLatticeOption{end})
    TrinLatticeOption{end}(j)=calcPayoff(TrinLatticeS0{end}(j), ...
        K, CallorPut);
end
%% Loop backwards to get values at the earlier times
for i=N:-1:1
    for j=1:length(TrinLatticeOption{i})
        TrinLatticeOption{i}(j)= ...
            M^-1*(pu*TrinLatticeOption{i+1}(j) ...
                +pm*TrinLatticeOption{i+1}(j+1) ...
                +pd*TrinLatticeOption{i+1}(j+2));
    end
```

```
end
OptionPrice=TrinLatticeOption{1}(1);
end
```

6. Modified Binomial Tian European Call - Put Option

```
function [OptionPrice, BinLatticeS0, BinLatticeOption] = ...
   priceOptionBinTian(S0, K, r, T, N, sigma, CallorPut)
dt=T/N;
M=exp(r*dt);
V=exp(sigma^2*dt);
u=M*V/2*(V+1+sqrt(V^2+2*V-3));
d=M*V/2*(V+1-sqrt(V^2+2*V-3));
p=(M-d)/(u-d);
%%
% Loop over each node of S0 price tree
BinLatticeS0=buildLattice(N+1,'bin');
BinLatticeS0{1}(1)=S0;
for i=2:N+1
   BinLatticeS0{i}(1)=BinLatticeS0{i-1}(1)*u;
   BinLatticeS0{i}(2)=BinLatticeS0{i-1}(1)*d;
   x=length(BinLatticeS0{i});
   if  x>2
      for j=3:x
         BinLatticeS0{i}(j)=BinLatticeS0{i-1}(j-1)*d;
      end
   end
   clear x
end
%% Calculate the value at expiry
BinLatticeOption = buildLattice(N+1, 'bin');
for j=1:length(BinLatticeOption{end})
   BinLatticeOption{end}(j)=calcPayoff(BinLatticeS0{end}(j), ...
      K, CallorPut);
end
%% Loop backwards to get values at the earlier times
for i = N:-1:1
   for j=1:length(BinLatticeOption{i})
      BinLatticeOption{i}(j)= ...
         M^-1*(p*BinLatticeOption{i+1}(j) ...
            +(1-p)*BinLatticeOption{i+1}(j+1));
   end
```

```
end
OptionPrice=BinLatticeOption{1}(1);
end
```

7. Modified Tian Trin1 European Call - Put Option

```
function [OptionPrice, TrinLatticeS0, TrinLatticeOption] = ...
            priceOptionTrin1Tian(S0, K, r, T, N, sigma, CallorPut)
dt=T/N;
M=exp(r*dt);
V=exp(sigma^2*dt);
m=M*(3-V)/2;
KA=M*(V+3)/4;
u=KA+sqrt(KA^2-m^2);
d=KA-sqrt(KA^2-m^2);
pu=1/3;
pm=1/3;
pd=1/3;
%% Loop over each node of S0 price tree
TrinLatticeS0=buildLattice(N+1,'trin');
TrinLatticeS0{1}(1)=S0;
for i=2:N+1
   TrinLatticeS0{i}(1)=TrinLatticeS0{i-1}(1)*u;
   TrinLatticeS0{i}(2)=TrinLatticeS0{i-1}(1)*m;
   TrinLatticeS0{i}(3)=TrinLatticeS0{i-1}(1)*d;
   x=length(TrinLatticeS0{i});
   if  x>3
      for j=4:x
         TrinLatticeS0{i}(j)=TrinLatticeS0{i-1}(j-2)*d;
      end
   end
   clear x
end
%% Calculate the value at expiry
TrinLatticeOption=buildLattice(N+1,'trin');
for j=1:length(TrinLatticeOption{end})
   TrinLatticeOption{end}(j)=calcPayoff(TrinLatticeS0{end}(j), ...
      K, CallorPut);
end
%% Loop backwards to get values at the earlier times
```

```
for i=N:-1:1
   for j=1:length(TrinLatticeOption{i})
      TrinLatticeOption{i}(j)= ...
         M^-1*(pu*TrinLatticeOption{i+1}(j) ...
            +pm*TrinLatticeOption{i+1}(j+1) ...
            +pd*TrinLatticeOption{i+1}(j+2));
   end
end
OptionPrice=TrinLatticeOption{1}(1);
end
```

8. Modified Tian Trin2 European Call - Put Option

```
function [OptionPrice, TrinLatticeS0, TrinLatticeOption] = ...
            priceOptionTrin2Tian(S0, K, r, T, N, sigma, CallorPut)
dt=T/N;
M=exp(r*dt);
V=exp(sigma^2*dt);
m=M*V^2;
KA=M/2*(V^4+V^3);
u=KA+sqrt(KA^2-m^2);
d=KA-sqrt(KA^2-m^2);
pu=(m*d-M*(m+d)+M^2*V)/((u-d)*(u-m));
pm=(M*(u+d)-u*d-M^2*V)/((u-m)*(m-d));
pd=(u*m-M*(u+m)+M^2*V)/((u-d)*(m-d));
%% Loop over each node of S0 price tree
TrinLatticeS0=buildLattice(N+1,'trin');
TrinLatticeS0{1}(1)=S0;
for i=2:N+1
   TrinLatticeS0{i}(1)=TrinLatticeS0{i-1}(1)*u;
   TrinLatticeS0{i}(2)=TrinLatticeS0{i-1}(1)*m;
   TrinLatticeS0{i}(3)=TrinLatticeS0{i-1}(1)*d;
   x=length(TrinLatticeS0{i});
   if  x>3
      for j=4:x
         TrinLatticeS0{i}(j)=TrinLatticeS0{i-1}(j-2)*d;
      end
   end
   clear x
end
%% Calculate the value at expiry
```

```
TrinLatticeOption=buildLattice(N+1,'trin');
for j=1:length(TrinLatticeOption{end})
   TrinLatticeOption{end}(j)=calcPayoff(TrinLatticeS0{end}(j), ...
      K, CallorPut);
end
%% Loop backwards to get values at the earlier times
for i=N:-1:1
   for j=1:length(TrinLatticeOption{i})
      TrinLatticeOption{i}(j)= ...
         M^-1*(pu*TrinLatticeOption{i+1}(j) ...
            +pm*TrinLatticeOption{i+1}(j+1) ...
            +pd*TrinLatticeOption{i+1}(j+2));
   end
end
OptionPrice=TrinLatticeOption{1}(1);
end
```

9. CRR Bin American Put Option

```
function [OptionPrice, BinLatticeS0, BinLatticeOption] = ...
   CRRBinAmerPut(S0, K, r, T, N, sigma)
dt=T/N;
M=exp(r*dt);
u=exp(sigma*sqrt(dt));
d=1/u;
p=(M-d)/(u-d);
%% Loop over each node of S0 price tree
BinLatticeS0=buildLattice(N+1,'bin');
BinLatticeS0{1}(1)=S0;
for i=2:N+1
   BinLatticeS0{i}(1)=BinLatticeS0{i-1}(1)*u;
   BinLatticeS0{i}(2)=BinLatticeS0{i-1}(1)*d;
   x=length(BinLatticeS0{i});
   if  x>2
      for j=3:x
         BinLatticeS0{i}(j)=BinLatticeS0{i-1}(j-1)*d;
      end
   end
   clear x
end
%% Calculate the value at expiry
```

```
BinLatticeOption = buildLattice(N+1, 'bin');
for j=1:length(BinLatticeOption{end})
   BinLatticeOption{end}(j)=calcPayoff(BinLatticeS0{end}(j), ...
      K, 'Put');
end
%% Loop backwards to get values at the earlier times
for i=N:-1:1
   for j=1:length(BinLatticeOption{i})
      BinLatticeOption{i}(j)= ...
         max(M^(-1)*(p*BinLatticeOption{i+1}(j) ...
            +(1-p)*BinLatticeOption{i+1}(j+1)),...
            K-BinLatticeS0{i}(j));
   end
end
OptionPrice=BinLatticeOption{1}(1);
end
```

10. <u>Boyle Trin American Put Option</u>

```
function [OptionPrice, TrinLatticeS0, TrinLatticeOption] = ...
               BoyleTrinAmerPut(S0, K, r, T, N, sigma)
dt=T/N;
M=exp(r*dt);
lamda=1.2;
V=M^2*(exp(sigma^2*dt)-1);
u=exp(lamda*sigma*sqrt(dt));
m=1;
d=exp(-lamda*sigma*sqrt(dt));
pu=((V+M^2-M)*u-(M-1))/((u-1)*(u^2-1));
pd=((V+M^2-M)*u^2-(M-1)*u^3)/((u-1)*(u^2-1));
pm=1-pu-pd;
%% Loop over each node of S0 price tree
TrinLatticeS0=buildLattice(N+1,'trin');
TrinLatticeS0{1}(1)=S0;
for i=2:N+1
   TrinLatticeS0{i}(1)=TrinLatticeS0{i-1}(1)*u;
   TrinLatticeS0{i}(2)=TrinLatticeS0{i-1}(1)*m;
   TrinLatticeS0{i}(3)=TrinLatticeS0{i-1}(1)*d;
   x=length(TrinLatticeS0{i});
   if  x>3
      for j=4:x
         TrinLatticeS0{i}(j)= TrinLatticeS0{i-1}(j-2)*d;
```

```
        end
    end
    clear x
end
%% Calculate the value at expiry
TrinLatticeOption=buildLattice(N+1,'trin');
for j=1:length(TrinLatticeOption{end})
    TrinLatticeOption{end}(j)=calcPayoff(TrinLatticeS0{end}(j), ...
        K, 'Put');
end
%% Loop backwards to get values at the earlier times
for i=N:-1:1
    for j=1:length(TrinLatticeOption{i})
        TrinLatticeOption{i}(j)= ...
            max(M^-1*(pu*TrinLatticeOption{i+1}(j) ...
                +pm*TrinLatticeOption{i+1}(j+1) ...
                +pd*TrinLatticeOption{i+1}(j+2)),...
                K-TrinLatticeS0{i}(j));
    end
end
OptionPrice=TrinLatticeOption{1}(1);
end
```

11. Tian Bin American Put Option

```
function [OptionPrice, BinLatticeS0, BinLatticeOption] = ...
    TianBinAmerPut(S0, K, r, T, N, sigma)
dt=T/N;
M=exp(r*dt);
V=exp(sigma^2*dt);
u=M*V/2*(V+1+sqrt(V^2+2*V-3));
d=M*V/2*(V+1-sqrt(V^2+2*V-3));
p=(M-d)/(u-d);
%%
% Loop over each node of S0 price tree
BinLatticeS0=buildLattice(N+1,'bin');
BinLatticeS0{1}(1)=S0;
for i=2:N+1
    BinLatticeS0{i}(1)=BinLatticeS0{i-1}(1)*u;
    BinLatticeS0{i}(2)=BinLatticeS0{i-1}(1)*d;
    x=length(BinLatticeS0{i});
```

```
    if  x>2
        for j=3:x
            BinLatticeS0{i}(j)=BinLatticeS0{i-1}(j-1)*d;
        end
    end
    clear x
end
%% Calculate the value at expiry
BinLatticeOption = buildLattice(N+1, 'bin');
for j=1:length(BinLatticeOption{end})
    BinLatticeOption{end}(j)=calcPayoff(BinLatticeS0{end}(j), ...
        K, 'Put');
end
%% Loop backwards to get values at the earlier times
for i = N:-1:1
    for j=1:length(BinLatticeOption{i})
        BinLatticeOption{i}(j)= ...
            max(M^-1*(p*BinLatticeOption{i+1}(j) ...
                +(1-p)*BinLatticeOption{i+1}(j+1)),...
                K-BinLatticeS0{i}(j));
    end
end
OptionPrice=BinLatticeOption{1}(1);
end
```

12. Tian Trin1 American Put Option

```
function [OptionPrice, TrinLatticeS0, TrinLatticeOption] = ...
                TianTrin1AmerPut(S0, K, r, T, N, sigma)
dt=T/N;
M=exp(r*dt);
V=exp(sigma^2*dt);
m=M*(3-V)/2;
KA=M*(V+3)/4;
u=KA+sqrt(KA^2-m^2);
d=KA-sqrt(KA^2-m^2);
pu=1/3;
pm=1/3;
pd=1/3;
%% Loop over each node of S0 price tree
```

```
TrinLatticeS0=buildLattice(N+1,'trin');
TrinLatticeS0{1}(1)=S0;
for i=2:N+1
   TrinLatticeS0{i}(1)=TrinLatticeS0{i-1}(1)*u;
   TrinLatticeS0{i}(2)=TrinLatticeS0{i-1}(1)*m;
   TrinLatticeS0{i}(3)=TrinLatticeS0{i-1}(1)*d;
   x=length(TrinLatticeS0{i});
   if  x>3
      for j=4:x
         TrinLatticeS0{i}(j)=TrinLatticeS0{i-1}(j-2)*d;
      end
   end
   clear x
end
%% Calculate the value at expiry
TrinLatticeOption=buildLattice(N+1,'trin');
for j=1:length(TrinLatticeOption{end})
   TrinLatticeOption{end}(j)=calcPayoff(TrinLatticeS0{end}(j), ...
      K, 'Put');
end
%% Loop backwards to get values at the earlier times
for i=N:-1:1
   for j=1:length(TrinLatticeOption{i})
      TrinLatticeOption{i}(j)= ...
         max(M^-1*(pu*TrinLatticeOption{i+1}(j) ...
            +pm*TrinLatticeOption{i+1}(j+1) ...
            +pd*TrinLatticeOption{i+1}(j+2)),...
            K-TrinLatticeS0{i}(j));
   end
end
OptionPrice=TrinLatticeOption{1}(1);
end
```

13. <u>Tian Trin2 American Put Option</u>

```
function [OptionPrice, TrinLatticeS0, TrinLatticeOption] = ...
            TianTrin2AmerPut(S0, K, r, T, N, sigma)
```

```
dt=T/N;
M=exp(r*dt);
V=exp(sigma^2*dt);
m=M*V^2;
KA=M/2*(V^4+V^3);
u=KA+sqrt(KA^2-m^2);
d=KA-sqrt(KA^2-m^2);
pu=(m*d-M*(m+d)+M^2*V)/((u-d)*(u-m));
pm=(M*(u+d)-u*d-M^2*V)/((u-m)*(m-d));
pd=(u*m-M*(u+m)+M^2*V)/((u-d)*(m-d));
%% Loop over each node of S0 price tree
TrinLatticeS0=buildLattice(N+1,'trin');
TrinLatticeS0{1}(1)=S0;
for i=2:N+1
    TrinLatticeS0{i}(1)=TrinLatticeS0{i-1}(1)*u;
    TrinLatticeS0{i}(2)=TrinLatticeS0{i-1}(1)*m;
    TrinLatticeS0{i}(3)=TrinLatticeS0{i-1}(1)*d;
    x=length(TrinLatticeS0{i});
    if  x>3
        for j=4:x
            TrinLatticeS0{i}(j)=TrinLatticeS0{i-1}(j-2)*d;
        end
    end
    clear x
end
%% Calculate the value at expiry
TrinLatticeOption=buildLattice(N+1,'trin');
for j=1:length(TrinLatticeOption{end})
    TrinLatticeOption{end}(j)=calcPayoff(TrinLatticeS0{end}(j), ...
        K, 'Put');
end
%% Loop backwards to get values at the earlier times
for i=N:-1:1
    for j=1:length(TrinLatticeOption{i})
        TrinLatticeOption{i}(j)= ...
            max(M^-1*(pu*TrinLatticeOption{i+1}(j) ...
                +pm*TrinLatticeOption{i+1}(j+1) ...
                +pd*TrinLatticeOption{i+1}(j+2)),...
                K-TrinLatticeS0{i}(j));
    end
end
OptionPrice=TrinLatticeOption{1}(1);
```

```
end
```

14. Accuracy Examination General Algorithm

(For instance: fa= TianTrin1AmerPut(S0, 185, r, T, 400, sigma); , fb= TianTrin1AmerPut(S0, 200, r, T, 400, sigma); , fc= TianTrin1AmerPut(S0, 215, r, T, 400, sigma);, where we define the rest of the input values)

```
AccurN=zeros(200,1);
for i=1:200
N=i;
p(N)= CRRBinAmerPut(S0, K, r, T, N, sigma);
f=fa; %We change for each execution the f value based on the model and the strike price
if abs((p(N)-f)/f)<e
AccurN(i)=N;
end
end
```

15. Parameter Estimation of Cox Ross Rubinstein Call Option Model with Time Steps N=100

```
function[x,resnorm,residual,exitflag]=CRRCalibration(~)
clear all
global S0;
global K;
global r;
global T;
global imp_vol;
global marketprice;
global k;
S0=zeros(191);
K=zeros(191,12);
r=zeros(191);
T=zeros(191,12);
imp_vol=zeros(191,12);
marketprice=zeros(191,12);
parameter=zeros(191,1);
res=zeros(191,1);
exit=zeros(191,1);
S0=xlsread('DataAppleCall.xls','price','B2:B192');
K=xlsread('DataAppleCall.xls','strike','B3:M193');
r=xlsread('DataAppleCall.xls','rate','B2:B192');
```

```
T=xlsread('DataAppleCall.xls','timetomaturity','B3:M193');
imp_vol=xlsread('DataAppleCall.xls','volatility','B3:M193');
marketprice=xlsread('DataAppleCall.xls','marketprice','B3:M193');
CRR_call_matrix=zeros(191,12);
for i=1:191
x0=[0.3792647];
lb=[0.001];
ub=[2];
k=i;
[x,resnorm,residual,exitflag]=lsqnonlin(@LSQDCRR,x0,lb,ub);
parameter(i)=x;
res(i)=resnorm;
exit(i)=exitflag;
for j=1:12
CRR_call_matrix(i,j)=priceOptionBinCRR(S0(i), K(i,j), r(i), T(i,j), 100, x(1), 'Call');
end
pricedata=(CRR_call_matrix);
end
xlswrite('DataAppleCall.xls',pricedata,'CRRresults','D3:O193');
xlswrite('DataAppleCall.xls',res,'CRRresults','B3:B193');
xlswrite('DataAppleCall.xls',parameter,'CRRresults','A3:A193');
end
```

16. Supporting Function for Parameter Estimation of Cox Ross Rubinstein Call Option Model with Time Steps N=100

```
function[CRR_lsqd]=LSQDCRR(x)
%Define Differences
global S0;
global K;
global r;
global T;
global imp_vol;
global marketprice;
global k;
CRR_lsqd=zeros(1,12);
for j=1:12
CRR_lsqd(j)=marketprice(k,j)-priceOptionBinCRR(S0(k), K(k,j), r(k), T(k,j), 100, x(1),
'Call');
end
```

17. Parameter Estimation of Tian Bin Call Option Model with Time Steps N=100

18. Supporting Function for Parameter Estimation of Tian Bin Call Option Model with Time Steps N=100

```
function[CRR_lsqd]=LSQDTIANBIN(x)
%Define Differences
global S0;
global K;
global r;
global T;
global imp_vol;
global marketprice;
global k;
CRR_lsqd=zeros(1,12);
for j=1:12
CRR_lsqd(j)=marketprice(k,j)-priceOptionBinTian(S0(k), K(k,j), r(k), T(k,j), 100, x(1), 'Call');
end
```

19. Parameter Estimation of Boyle Trinomial Call Option Model with Time Steps N=100

```
function[x,resnorm,residual,exitflag]=BoyleTRINCalibration(~)
clear all
global S0;
global K;
global r;
global T;
global imp_vol;
global marketprice;
global k;
S0=zeros(191);
K=zeros(191,12);
r=zeros(191);
T=zeros(191,12);
imp_vol=zeros(191,12);
marketprice=zeros(191,12);
parameter=zeros(191,2);
res=zeros(191,1);
```

```
exit=zeros(191,1);
S0=xlsread('DataAppleCall.xls','price','B2:B192');
K=xlsread('DataAppleCall.xls','strike','B3:M193');
r=xlsread('DataAppleCall.xls','rate','B2:B192');
T=xlsread('DataAppleCall.xls','timetomaturity','B3:M193');
imp_vol=xlsread('DataAppleCall.xls','volatility','B3:M193');
marketprice=xlsread('DataAppleCall.xls','marketprice','B3:M193');
BoyleTRIN_call_matrix=zeros(191,12);
for i=1:191
x0=[0.3792647,1.2];
lb=[0.001,0.7];
ub=[2,3.5];
k=i;
[x,resnorm,residual,exitflag]=lsqnonlin(@LSQDBOYLETRIN,x0,lb,ub);
parameter(i,:)=x;
res(i)=resnorm;
exit(i)=exitflag;
for j=1:12
BoyleTRIN_call_matrix(i,j)=priceOptionTrinBoyleEurC(S0(i), K(i,j), r(i), T(i,j), 100, x(1),
x(2), 'Call');
end
pricedata=(BoyleTRIN_call_matrix);
end
xlswrite('DataAppleCall.xls',pricedata,'BoyleTRINresults','E3:P193');
xlswrite('DataAppleCall.xls',res,'BoyleTRINresults','C3:C193');
xlswrite('DataAppleCall.xls',parameter,'BoyleTRINresults','A3:B193');
end
```

20. Supporting Function for Parameter Estimation of Boyle Trinomial Call Option Model with Time Steps N=100

```
function[CRR_lsqd]=LSQDBOYLETRIN(x)
%Define Differences
global S0;
global K;
global r;
global T;
global imp_vol;
global marketprice;
global k;
CRR_lsqd=zeros(1,12);
for j=1:12
```

```
CRR_lsqd(j)=marketprice(k,j)-priceOptionTrinBoyleEurC(S0(k), K(k,j), r(k), T(k,j), 100,
x(1),x(2), 'Call');
end
```

21. Boyle Trinomial Option Model used for Parameter Estimation

```
function [OptionPrice, TrinLatticeS0, TrinLatticeOption] = ...
            priceOptionTrinBoyleEurC(S0, K, r, T, N, sigma, lamda, CallorPut)
dt=T/N;
M=exp(r*dt);
V=M^2*(exp(sigma^2*dt)-1);
u=exp(lamda*sigma*sqrt(dt));
m=1;
d=exp(-lamda*sigma*sqrt(dt));
pu=((V+M^2-M)*u-(M-1))/((u-1)*(u^2-1));
pd=((V+M^2-M)*u^2-(M-1)*u^3)/((u-1)*(u^2-1));
pm=1-pu-pd;
%% Loop over each node of S0 price tree
TrinLatticeS0=buildLattice(N+1,'trin');
TrinLatticeS0{1}(1)=S0;
for i=2:N+1
  TrinLatticeS0{i}(1)=TrinLatticeS0{i-1}(1)*u;
  TrinLatticeS0{i}(2)=TrinLatticeS0{i-1}(1)*m;
  TrinLatticeS0{i}(3)=TrinLatticeS0{i-1}(1)*d;
  x=length(TrinLatticeS0{i});
  if  x>3
    for j=4:x
      TrinLatticeS0{i}(j)= TrinLatticeS0{i-1}(j-2)*d;
    end
  end
  clear x
end
%% Calculate the value at expiry
TrinLatticeOption=buildLattice(N+1,'trin');
for j=1:length(TrinLatticeOption{end})
  TrinLatticeOption{end}(j)=calcPayoff(TrinLatticeS0{end}(j), ...
    K, CallorPut);
end
%% Loop backwards to get values at the earlier times
for i=N:-1:1
  for j=1:length(TrinLatticeOption{i})
```

```
        TrinLatticeOption{i}(j)= ...
            M^-1*(pu*TrinLatticeOption{i+1}(j) ...
                +pm*TrinLatticeOption{i+1}(j+1) ...
                +pd*TrinLatticeOption{i+1}(j+2));
    end
end
OptionPrice=TrinLatticeOption{1}(1);
end
```

22. <u>Parameter Estimation of Tian Trin1 Call Option Model with Time Steps N=100</u>

```
function[x,resnorm,residual,exitflag]=TianTRIN1Calibration(~)
clear all
global S0;
global K;
global r;
global T;
global imp_vol;
global marketprice;
global k;
S0=zeros(191);
K=zeros(191,12);
r=zeros(191);
T=zeros(191,12);
imp_vol=zeros(191,12);
marketprice=zeros(191,12);
parameter=zeros(191,1);
res=zeros(191,1);
exit=zeros(191,1);
S0=xlsread('DataAppleCall.xls','price','B2:B192');
K=xlsread('DataAppleCall.xls','strike','B3:M193');
r=xlsread('DataAppleCall.xls','rate','B2:B192');
T=xlsread('DataAppleCall.xls','timetomaturity','B3:M193');
imp_vol=xlsread('DataAppleCall.xls','volatility','B3:M193');
marketprice=xlsread('DataAppleCall.xls','marketprice','B3:M193');
TianTRIN1_call_matrix=zeros(191,12);
for i=1:191
x0=[0.3792647];
lb=[0.001];
```

```
ub=[2];
k=i;
[x,resnorm,residual,exitflag]=lsqnonlin(@LSQDTIANTRIN1,x0,lb,ub);
parameter(i)=x;
res(i)=resnorm;
exit(i)=exitflag;
for j=1:12
TianTRIN1_call_matrix(i,j)=priceOptionTrin1Tian(S0(i), K(i,j), r(i), T(i,j), 100, x(1),
'Call');
end
pricedata=(TianTRIN1_call_matrix);
end
xlswrite('DataAppleCall.xls',pricedata,'TianTRIN1results','D3:O193');
xlswrite('DataAppleCall.xls',res,'TianTRIN1results','B3:B193');
xlswrite('DataAppleCall.xls',parameter,'TianTRIN1results','A3:A193');
end
```

23. Supporting Function for Parameter Estimation of Tian Trin1 Call Option Model with Time Steps N=100

```
function[CRR_lsqd]=LSQDTIANTRIN1(x)
%Define Differences
global S0;
global K;
global r;
global T;
global imp_vol;
global marketprice;
global k;
CRR_lsqd=zeros(1,12);
for j=1:12
CRR_lsqd(j)=marketprice(k,j)-priceOptionTrin1Tian(S0(k), K(k,j), r(k), T(k,j), 100, x(1),
'Call');
end
```

24. Parameter Estimation of Tian Trin2 Call Option Model with Time Steps N=100

```
function[x,resnorm,residual,exitflag]=TianTRIN2Calibration(~)
clear all
```

```
global S0;
global K;
global r;
global T;
global imp_vol;
global marketprice;
global k;
S0=zeros(191);
K=zeros(191,12);
r=zeros(191);
T=zeros(191,12);
imp_vol=zeros(191,12);
marketprice=zeros(191,12);
parameter=zeros(191,1);
res=zeros(191,1);
exit=zeros(191,1);
S0=xlsread('DataAppleCall.xls','price','B2:B192');
K=xlsread('DataAppleCall.xls','strike','B3:M193');
r=xlsread('DataAppleCall.xls','rate','B2:B192');
T=xlsread('DataAppleCall.xls','timetomaturity','B3:M193');
imp_vol=xlsread('DataAppleCall.xls','volatility','B3:M193');
marketprice=xlsread('DataAppleCall.xls','marketprice','B3:M193');
TianTRIN2_call_matrix=zeros(191,12);
for i=1:191
x0=[0.3792647];
lb=[0.001];
ub=[2];
k=i;
[x,resnorm,residual,exitflag]=lsqnonlin(@LSQDTIANTRIN2,x0,lb,ub);
parameter(i)=x;
res(i)=resnorm;
exit(i)=exitflag;
for j=1:12
TianTRIN2_call_matrix(i,j)=priceOptionTrin2Tian(S0(i), K(i,j), r(i), T(i,j), 100, x(1),
'Call');
end
pricedata=(TianTRIN2_call_matrix);
end
xlswrite('DataAppleCall.xls',pricedata,'TianTRIN2results','D3:O193');
xlswrite('DataAppleCall.xls',res,'TianTRIN2results','B3:B193');
xlswrite('DataAppleCall.xls',parameter,'TianTRIN2results','A3:A193');
end
```

25. Supporting Function for Parameter Estimation of Tian Trin2 Call Option Model with Time Steps N=100

```
function[CRR_lsqd]=LSQDTIANTRIN2(x)
%Define Differences
global S0;
global K;
global r;
global T;
global imp_vol;
global marketprice;
global k;
CRR_lsqd=zeros(1,12);
for j=1:12
CRR_lsqd(j)=marketprice(k,j)-priceOptionTrin2Tian(S0(k), K(k,j), r(k), T(k,j), 100, x(1), 'Call');
end
```

26. Forecast Function of CRR Call Option Model

```
function[pricedata,pricedata_squares,res]=CRRforecast(~)
clear all
sigma=0.289646; %Average of last 3 observations(21/4/2021-23/4/2021)
S0=zeros(10);
K=zeros(10,12);
r=zeros(10);
T=zeros(10,12);
marketprice=zeros(10,12);
%Εισάγω τις τιμές
S0=xlsread('DataAppleCall.xls','price','B193:B202');
K=xlsread('DataAppleCall.xls','strike','B194:M203');
r=xlsread('DataAppleCall.xls','rate','B193:B202');
T=xlsread('DataAppleCall.xls','timetomaturity','B194:M203');
marketprice=xlsread('DataAppleCall.xls','marketprice','B194:M203');
res=zeros(10,1);
CRR_call_matrix=zeros(10,12);
CRR_call_matrix_squares=zeros(10,12);
for i=1:10
for j=1:12
```

```
CRR_call_matrix(i,j)=abs(marketprice(i,j)-priceOptionBinCRR(S0(i), K(i,j), r(i), T(i,j),
100, sigma, 'Call'));
CRR_call_matrix_squares(i,j)=(abs(marketprice(i,j)-priceOptionBinCRR(S0(i), K(i,j), r(i),
T(i,j), 100, sigma, 'Call'))^2);
end
pricedata=[CRR_call_matrix];
pricedata_squares=[CRR_call_matrix_squares];
end
for i=1:10
res(i)=(sum(CRR_call_matrix_squares(i,:)));
end
xlswrite('DataAppleCall.xls',pricedata,'CRRforecast','B2:M11');
xlswrite('DataAppleCall.xls',pricedata_squares,'CRRforecast','B13:M22');
xlswrite('DataAppleCall.xls',res,'CRRforecast','B24:B33');
end
```

27. Forecast Function of Tian Binomial Call Option Model

```
function[pricedata,pricedata_squares,res]=TianBINforecast(~)
clear all
sigma=0.289816; %Average of last 3 observations(21/4/2021-23/4/2021)
S0=zeros(10);
K=zeros(10,12);
r=zeros(10);
T=zeros(10,12);
marketprice=zeros(10,12);
%Εισάγω τις τιμές
S0=xlsread('DataAppleCall.xls','price','B193:B202');
K=xlsread('DataAppleCall.xls','strike','B194:M203');
r=xlsread('DataAppleCall.xls','rate','B193:B202');
T=xlsread('DataAppleCall.xls','timetomaturity','B194:M203');
marketprice=xlsread('DataAppleCall.xls','marketprice','B194:M203');
res=zeros(10,1);
TianBIN_call_matrix=zeros(10,12);
TianBIN_call_matrix_squares=zeros(10,12);
for i=1:10
for j=1:12
TianBIN_call_matrix(i,j)=abs(marketprice(i,j)-priceOptionBinTian(S0(i), K(i,j), r(i), T(i,j),
100, sigma, 'Call'));
TianBIN_call_matrix_squares(i,j)=(abs(marketprice(i,j)-priceOptionBinTian(S0(i), K(i,j),
r(i), T(i,j), 100, sigma, 'Call'))^2);
end
```

```
pricedata=[TianBIN_call_matrix];
pricedata_squares=[TianBIN_call_matrix_squares];
end
for i=1:10
res(i)=(sum(TianBIN_call_matrix_squares(i,:)));
end
xlswrite('DataAppleCall.xls',pricedata,'TianBINforecast','B2:M11');
xlswrite('DataAppleCall.xls',pricedata_squares,'TianBINforecast','B13:M22');
xlswrite('DataAppleCall.xls',res,'TianBINforecast','B24:B33');
end
```

28. <u>Forecast Function of Boyle Trinomial Call Option Model</u>

```
function[pricedata,pricedata_squares,res]=BoyleTRINforecast(~)
clear all
sigma=0.289885; %Average of last 3 observations(21/4/2021-23/4/2021)
lamda=2.028614; %Average of last 3 observations(21/4/2021-23/4/2021)
S0=zeros(10);
K=zeros(10,12);
r=zeros(10);
T=zeros(10,12);
marketprice=zeros(10,12);
%Εισάγω τις τιμές
S0=xlsread('DataAppleCall.xls','price','B193:B202');
K=xlsread('DataAppleCall.xls','strike','B194:M203');
r=xlsread('DataAppleCall.xls','rate','B193:B202');
T=xlsread('DataAppleCall.xls','timetomaturity','B194:M203');
marketprice=xlsread('DataAppleCall.xls','marketprice','B194:M203');
res=zeros(10,1);
BoyleTRIN_call_matrix=zeros(10,12);
BoyleTRIN_call_matrix_squares=zeros(10,12);
for i=1:10
for j=1:12
BoyleTRIN_call_matrix(i,j)=abs(marketprice(i,j)-priceOptionTrinBoyleEurC(S0(i),   K(i,j),
r(i), T(i,j), 100, sigma, lamda, 'Call'));
BoyleTRIN_call_matrix_squares(i,j)=(abs(marketprice(i,j)-
priceOptionTrinBoyleEurC(S0(i), K(i,j), r(i), T(i,j), 100, sigma, lamda, 'Call'))^2);
end
pricedata=[BoyleTRIN_call_matrix];
pricedata_squares=[BoyleTRIN_call_matrix_squares];
end
for i=1:10
```

```
res(i)=(sum(BoyleTRIN_call_matrix_squares(i,:)));
end
xlswrite('DataAppleCall.xls',pricedata,'BoyleTRINforecast','B2:M11');
xlswrite('DataAppleCall.xls',pricedata_squares,'BoyleTRINforecast','B13:M22');
xlswrite('DataAppleCall.xls',res,'BoyleTRINforecast','B24:B33');
end
```

29. Forecast Function of Tian Trin1 Call Option Model

```
function[pricedata,pricedata_squares,res]=TianTRIN1forecast(~)
clear all
sigma=0.289697; %Average of last 3 observations(21/4/2021-23/4/2021)
S0=zeros(10);
K=zeros(10,12);
r=zeros(10);
T=zeros(10,12);
marketprice=zeros(10,12);
%Εισάγω τις τιμές
S0=xlsread('DataAppleCall.xls','price','B193:B202');
K=xlsread('DataAppleCall.xls','strike','B194:M203');
r=xlsread('DataAppleCall.xls','rate','B193:B202');
T=xlsread('DataAppleCall.xls','timetomaturity','B194:M203');
marketprice=xlsread('DataAppleCall.xls','marketprice','B194:M203');
res=zeros(10,1);
TianTRIN1_call_matrix=zeros(10,12);
TianTRIN1_call_matrix_squares=zeros(10,12);
for i=1:10
for j=1:12
TianTRIN1_call_matrix(i,j)=abs(marketprice(i,j)-priceOptionTrin1Tian(S0(i), K(i,j), r(i),
T(i,j), 100, sigma, 'Call'));
TianTRIN1_call_matrix_squares(i,j)=(abs(marketprice(i,j)-priceOptionTrin1Tian(S0(i),
K(i,j), r(i), T(i,j), 100, sigma, 'Call'))^2);
end
pricedata=[TianTRIN1_call_matrix];
pricedata_squares=[TianTRIN1_call_matrix_squares];
end
for i=1:10
res(i)=(sum(TianTRIN1_call_matrix_squares(i,:)));
end
xlswrite('DataAppleCall.xls',pricedata,'TianTRIN1forecast','B2:M11');
xlswrite('DataAppleCall.xls',pricedata_squares,'TianTRIN1forecast','B13:M22');
xlswrite('DataAppleCall.xls',res,'TianTRIN1forecast','B24:B33');
```

```
end
```

30. Forecast Function of Tian Trin2 Call Option Model

```
function[pricedata,pricedata_squares,res]=TianTRIN2forecast(~)
clear all
sigma= 0.290014; %Average of last 3 observations(21/4/2021-23/4/2021)
S0=zeros(10);
K=zeros(10,12);
r=zeros(10);
T=zeros(10,12);
marketprice=zeros(10,12);
%Εισάγω τις τιμές
S0=xlsread('DataAppleCall.xls','price','B193:B202');
K=xlsread('DataAppleCall.xls','strike','B194:M203');
r=xlsread('DataAppleCall.xls','rate','B193:B202');
T=xlsread('DataAppleCall.xls','timetomaturity','B194:M203');
marketprice=xlsread('DataAppleCall.xls','marketprice','B194:M203');
res=zeros(10,1);
TianTRIN2_call_matrix=zeros(10,12);
TianTRIN2_call_matrix_squares=zeros(10,12);
for i=1:10
for j=1:12
TianTRIN2_call_matrix(i,j)=abs(marketprice(i,j)-priceOptionTrin2Tian(S0(i), K(i,j), r(i),
T(i,j), 100, sigma, 'Call'));
TianTRIN2_call_matrix_squares(i,j)=(abs(marketprice(i,j)-priceOptionTrin2Tian(S0(i),
K(i,j), r(i), T(i,j), 100, sigma, 'Call'))^2);
end
pricedata=[TianTRIN2_call_matrix];
pricedata_squares=[TianTRIN2_call_matrix_squares];
end
for i=1:10
res(i)=(sum(TianTRIN2_call_matrix_squares(i,:)));
end
xlswrite('DataAppleCall.xls',pricedata,'TianTRIN2forecast','B2:M11');
xlswrite('DataAppleCall.xls',pricedata_squares,'TianTRIN2forecast','B13:M22');
xlswrite('DataAppleCall.xls',res,'TianTRIN2forecast','B24:B33');
end
```