Doctoral **Business**

2020-4

Valuation of American Options and Employee Stock Options

Qianru (Jennifer) Shang
Technological University Dublin

Follow this and additional works at: https://arrow.tudublin.ie/busdoc

Part of the Finance and Financial Management Commons

Recommended Citation

Valuation of American Options and Employee Stock Options

By

Qianru Shang (Jennifer)

Thesis Submitted for the Award of PhD

(Doctor of Philosophy)

School of Accounting and Finance

College of Business

Technological University Dublin

Supervisor: Dr. Brian Byrne

April 2020

Abstract

Options play an important role in the financial world and are actively traded with huge trading volume. Most of the options traded on exchanges are American options. Spanning over a few decades, the American option pricing problem continues to intrigue scholars and practitioners in finance. The employee stock options (ESOs), a variant of American options, has been increasingly popular for firms to compensate, motivate and retain employees. ESOs importantly do not trade in markets nevertheless fair value must be determined – often by accountants. Unique features of ESOs however complicate the valuation. Our research, consisting of three papers, focuses on the improved lattice techniques for valuing American options and ESOs. Research paper 1 (Chapter 2) introduces an intelligent lattice search algorithm to efficiently locate the optimal exercise boundary for American options. The computational runtime can be reduced from over 18 minutes down to less than 3 seconds to estimate a 15,000-step CRR binomial tree. Research paper 2 (Chapter 3) introduces a set of lattice techniques to the Leisen-Reimer and Tian binomial models for American options pricing. A level of accuracy and efficiency combined can be achieve that surpass analytical solution models prominent in the literature. Moreover, lattices importantly afford an explicit trade-off locus between accuracy and speed that can be navigated according to predetermined precision tolerance levels and option types. These should have practical relevance to trading platforms that require real-time estimates of implied volatility. Research paper 3 (Chapter 4) proposes adjustments to the Hull-White ESO pricing model, based on insights developed by Boyle-Lau and Tian specifications. The proposed Hull-White-Boyle-Lau and Hull-White-Tian revamps expand the practicable menu choice available to stakeholders tasked with the valuation of these ESOs. Accountants, across many jurisdictions, are subjected to higher demands for disclosure and fair valuation. The streamlined valuation approaches developed here may prove

useful in expanding the tool kit of practicable/workable models. This improved efficiency can be harnessed even at the level of a basic spreadsheet and this this should assist in testing, validating and benchmarking valuation in lattices and in evaluating the newer generation of closed-form solutions emerging in the literature.

Declaration

I certify that this thesis, which I now submit for examination for the award of PhD (doctor of philosophy) is entirely my own work and has not been taken from the work of others save and to the extent that such work has been cited and acknowledged within the text of my work.

This thesis was prepared according to the regulations for PhD of the Technological University Dublin has not been submitted in whole or in part for an award in any other institute or university. The work reported on in this thesis conforms to the principles and requirements of TU Dublin's guidelines for ethics in research.

TU Dublin has permission to keep, to lend or to copy this thesis in whole or in part, on condition that any such use of the material be duly acknowledged.

Signature _____Qianru Shang_____

Date _____12/05/2020_____

Acknowledgements

List of Abbreviations

Accel CRR	Accelerated CRR model with intelligent lattice search, dynamic memory, and truncation technologies
Accel LR	Accelerated LR model with intelligent lattice search, dynamic memory, and truncation technologies
Accel LRBS	Accel LR model with Black-Scholes smoothing
Accel LRRE	Accel LR model with Richardson Extrapolation
Accel LRBS&RE	Accel LR model with Black-Scholes smoothing and Richardson Extrapolation
Accel Tian	Accelerated Tian model with intelligent lattice search, dynamic memory, and truncation technologies
Accel TianBS	Accel Tian model with Black-Scholes smoothing
Accel TianRE	Accel Tian model with Richardson Extrapolation
Accel TianBS&RE	Accel Tian model with Black Scholes smoothing and Richardson Extrapolation
AER	Ratio between American value and European value of the option
ASC	Accounting Standards Codification
BAW	Barone-Adesi and Whaley (1987) analytical formula
Bjerksund93	Bjerksund and Stensland (1993) analytical formula
Bjerksund02	Bjerksund and Stensland (2002) analytical formula
CAS	Chinese Accounting Standards
Chen-Joshi	Chen and Joshi (2012) binomial model
CRR	Cox, Ross and Rubinstein (1979) binomial model
ESOs	Employee Stock Options
FAS	Financial Accounting Standard
FASB	Financial Accounting Standard Board
HW	Hull and White (2004) binomial model
IASB	International Accounting Standard Board
IFRS	International Financial Reporting Standard
IV	Implied Volatility
Ju-Zhong	Ju and Zhong (1999) analytical formula

LR	Leisen and Reimer (1996) binomial model
RMSRE	Root mean squared relative error
OTC	Over the counter
SAB	Staff Accounting Bulletin
SEC	Securities and Exchange Commission
Tian	Tian (1993) binomial option pricing model
Time	Execution time
HW	Hull and White (2004) ESOs pricing model
HWBL	Modified HW model with Boyle and Lau (1994) specification
HWTian	Modified HW model with Tian (1999) specification
IDE	Integrated development environment

Table of Contents

List of Figures and Tables

Chapter 1 – Introduction

Introduction

Options have become increasing important in the financial world. Options are actively traded both on exchanges and in the over-the-counter (OTC) market with huge trading volume in terms of underlying assets. Generally, options give the holder a right to buy or sell the underlying assets at a fixed price. In terms of the exercise policy, the options can be divided into European options and American options. European options can only be exercised at their maturity while American options allow the option holders to exercise the options at any time before the maturity. It is clear that American options afford greater flexibility than European analogues. This may confer greater value but also complicates valuation. The American option pricing problem, spanning over the past few decades, continues to intrigue scholars and practitioners in finance. A closed-form solution for European option pricing is available using Black and Scholes (1973), while no equivalently simple closed-form solution exists for valuing American options robustly. American options valuation presents challenges with trade-offs embedded along an accuracy-efficiency spectrum. The pricing methods can be largely classified into analytical approximations or numerical methods[1]. Approximation methods are heavily relied upon to expedite estimation but necessarily produce error. Numerical techniques are regularly used as benchmarks for analytical approximations on account of the high level of estimation accuracy and low speed.

Options can be used for hedging, speculation and arbitrage (Hull and White, 2013). Financial institutions usually use options to eliminate/mitigate their exposure against the movement of underlying asset price. Delta hedging, a commonly-used hedging strategy, conceptually creates a

---

[1] Here we only consider the option written on a single underlying asset.

zero-delta (delta neutral) position by offsetting the delta of the stock position with that of the option position. The option position, however, has to be frequently adjusted as the delta of the option is continuously changing. In order to achieve this dynamic hedging, accurate and timely estimation for options by necessity prerequisites. Nowadays, options as well as other derivatives are more heavily regulated given the well documented concerns relating to the role of leverage during the financial crisis, starting 2007.

Employee stock options (ESOs) are now a relatively standard form of remuneration particularly amongst new technology companies. In the early part of the century, these American-European hybrid instruments grew in popularity. Hall and Murphy (2002) reported that 94% of S&P 500 companies granted options to the top executives. Frydman and Saks (2007) stated that ESOs constitute over 40% of their total compensation for the same cohort of executives. In 2004, the International Financial Reporting Standard No. 2 (IFRS 2) and the Revised Financial Accounting Standard No. 123 (FAS 123R) both require companies to measure and report the fair value of ESOs. In 2009, FASB integrated FAS 123R into Accounting Standards Codification as Topic 718 (ASC 718), and a corresponding updated version for non-employees was released in 2018. IFRS 2, FAS 123R and ASC 718 all suggest applying either the Black-Scholes model with an expected holding term or a lattice framework with suboptimal exercise and the term being an output of the model.

Rubinstein (1995) identified key hallmark features that distinguish ESOs from other plain vanilla options, including longer maturities, vesting periods, suboptimal early exercise, and non-hedge

and non-transferability[2]. These characteristics indicate that the option structure of ESOs are typically a hybrid with a blend of European and American exercise rights. For pricing ESOs, the Black-Scholes model is evidently simpler to apply with less computational cost while lattice models have advantages in terms of higher accuracy and more flexibility. Given the well documented computational cost of standard Hull and White (2004) implementation, the Black-Scholes model is normally applied by industry practitioners while academics favour the binomial framework.

Research Objectives

As discussed above, advances in American option theory and valuation play an important role in in the development of ancillary valuation models and this development still continues. Employee stock options, a variant of American options, have become increasingly important but their valuation is characterised by a trade-off between accuracy and speed of computation. This thesis focuses on the valuation of American options and ESOs and consists of three distinct papers with separate but interlinked themes. The first two paper (Chapter 2 and 3) propose optimal lattice-based models for valuing American option and the third paper (Chapter 4) extends findings to ESO pricing. The chronology of research output/objectives of each paper are as follows.

The first research paper: "American Option Pricing: An Accelerated Lattice Model with Intelligent Lattice Search" (Shang and Byrne, 2019) introduce an intelligent lattice search routine to accelerate the classic Cox, Ross and Rubinstein (1979, CRR) binomial model for American option

---

[2] ESOs cannot be exercised during the vesting period and the early exercise is possible after the vesting period and before the maturity.

pricing. The literature relating to analytical and numerical methods for American option pricing as well as the development of optimal exercise boundary theory for lattice models is reviewed. It has been found that lattice models, even though they possess high estimation accuracy, are characterised by low computational speed. Also, the existing optimal exercise boundary theory of the CRR binomial framework has limitations linked to dividend yield. This paper mitigates these two challenges that are signalled in theory and practice:

1. Remove the restriction of the existing optimal exercise boundary theory related to dividends, as pointed out by Curran (1995).
2. Accelerate CRR binomial model for pricing American options without disturbing the accuracy.

To accomplish the first objective, the optimal exercise boundary of a CRR binomial model is investigated. A number of Propositions and Theorems are proposed with mathematical proof. This serves to address limitations regarding the effect of dividend yield which originally posed a stumbling block for Curran (1995) and restricted the wide applicability of his model when dividends were pertinent to estimation. For the second objective, an intelligent lattice search algorithm is created in Excel VBA and later in C++, incorporating and extending Kim and Byun (1994) and Curran (1995) boundary theory. In addition, truncation technology and dynamic memory are also applied, to accelerate the lattice model for pricing American options.

The second paper, entitled "American Option Pricing: Optimal Lattice Models and Multidimensional Efficiency Tests", aims to apply Intelligent Lattice Search to Leisen and Reimer (1996, LR) and Tian (1993, Tian) binomial models for American option pricing and examine the efficiency of the existing models in pricing different types of the options. The literature on the

4

analytical methods, numerical techniques, boundary theory, and efficiency tests is reviewed. It has been found that the acceleration techniques proposed in Shang and Byrne (2019) may be able to apply to other lattice models that are similar in construction to the CRR model. Also, option classification can be used to design multidimensional efficiency tests for option pricing models (Pressacco, et al, 2008). Hence, this paper sets the following objectives:

1. Improve LR and Tian binomial models for pricing American options.

2. Run multidimensional efficiency tests for the existing option pricing models.

To achieve the first objective, it is necessary to establish that the optimal exercise boundary applies to the LR and Tian models. This is examined and a set of lattice techniques are adapted to LR and Tian models. For the second objective, lattice models and analytical formulae are tested for pricing different groups of options according to the deepness of American quality and the moneyness of the options.

The final research paper, titled "Accounting for Employee Stock Options: Accelerating Convergence", aims to accelerate the Hull and White (2004, HW) ESOs pricing model. The reform of accounting standards for ESOs and the literature on the ESOs pricing models including adjusted Black-Scholes model and numerical models are reviewed. It has been found that accounting standards do widely specify valuation models including an adapted Black Scholes (1973) framework, Lattices and Monte Carlo. The Hull-White lattice, widely considered to provide a sounder theoretical treatment, is typically slow to converge (Chendra and Sidarto, 2020). This paper introduces a number of improvements and sets the following objectives:

1. Analysing error source of Hull-White ESOs pricing model.

2. Solve the sluggish convergence of the Hull-White model.

3. Further accelerate Hull-White model using numerical techniques.

To accomplish the first objective, the unique characteristics of the Hull-White model and the error source of lattice models for barrier options are analysed. The second objective is progressed by imposing two specifications for barrier option pricing which to mitigate the error of Hull-White model. For the final objective, numerical techniques previously proposed for American options are applied to accelerate Hull-White model.

Thesis Structure

Chapter 1 includes an introduction to the valuation of American options and employee stock options, as well as mapping out research objectives. Three distinct research papers are formatted into Chapter 2, 3, and 4 respectively. Each of these three Chapters has the following structure: Abstract, Literature Review, Methodology, Numerical Results, and Conclusion. Chapter 5 concludes the main research findings and discusses future research. Finally, References, Appendices, List of Publication and Conference, and List of Employability Skills and Discipline Specific Skills Training are provided.

Chapter 2: American Option Pricing: An Accelerated Lattice Model with Intelligent Lattice Search

Abstract

We introduce to the literature an intelligent lattice search algorithm to efficiently locate the optimal exercise boundary for American options. Lattice models can be accelerated by incorporating intelligent lattice search, truncation and dynamic memory. We reduce computational runtime from over 18 minutes down to less than 3 seconds to estimate a 15,000-step binomial tree where the results obtained are consistent with a widely acclaimed literature. Delta and Implied Volatility can also be accelerated relative to standard models. Lattice estimation, in general, is considered to be slow and not practical for valuing large books of options or for promptly rebalancing a risk neutral portfolio. Our technique transforms standard binomial trees and renders them to be at least on par with commonly used analytical formulae. More importantly, intelligent lattice search can be tweaked to reach varying levels of accuracy with different step size, while conventional analytical formulae are less flexible.

Literature Review

American option pricing can present challenges with trade-offs embedded along an accuracy-efficiency spectrum. Approximation methods necessarily produce error. The margin of that error must be weighed against the uptick in estimation speed. Geske and Johnson (1984), Bunch and Johnson (1992), Huang et al. (1996), Carr (1998) and Ju (1998) developed analytic approximations that were convergent in the sense that when additional terms were included, their respective techniques became increasingly accurate while less efficient. Zhu (2006) developed an exact

solution in the form of a Taylor's series expansion, which contains infinitely many terms. The model, however, is not practicable in terms of speed (Medvedev and Scaillet 2010). Barone-Adesi and Whaley (1987, BAW) developed typically faster approximation techniques. Ju and Zhong (1999, Ju-Zhong), based on BAW, served to improve the longer maturity options accuracy with little sacrifice in terms of efficiency. Li (2010) extended further Ju-Zhong by introducing an improved smoothing condition for American options. BAW, Ju-Zhong and Li (2010) approaches were all found to share the limitation that pricing is not convergent to the "true" price. Including additional terms does not invariably produce greater accuracy (Fabozzi et al. 2016). Instead of applying a quadratic approximation, Bjerksund and Stensland (1993, Bjerksund93) simplify the optimal exercise strategy by assuming a unique flat boundary. The improved approximation is presented in Bjerksund and Stensland (2002, Bjerksund02) where a second bound is also introduced. In practice, their lower bound approach represents an accurate and very computer efficient approximation to the true American option value. In this paper, we compare our accelerated binomial model with these commonly used analytical approximations. We show that our model can be comparable with these analytical approximations in speed for option pricing, delta estimation and implied volatility estimation.

In practice, traders are likely to prioritize speed if approximation only entails a small compromise in accuracy. Nimble responses are viewed as important for preserving profit margins and for timely hedging. Not surprisingly then approximation techniques are relied upon heavily despite limitations. Estimating a book of options requires some insight on how the respective parameter values are likely to be incorporated into a given model. This might not be as simple as it seems. Uniform performance of analytical models cannot be guaranteed. This typically means that analytic models are regularly benchmarked against numerical techniques. Cox, Ross and

Rubinstein (1979, CRR) developed a binomial approach. It is widely accepted that greater accuracy can be introduced by making the CRR lattice mesh finer or by, more prosaically, incorporating a larger number of steps. Broadie and Detemple (1996) used a 15,000-step binomial model to obtain "True values". The binomial model is therefore acknowledged as a reliable workhorse and serves to benchmark other techniques. This lattice method, however, is distinctly viewed as having substantial computational workloads, runtimes and gluttonous memory requirements. Not surprisingly, many variants and extensions of the original CRR binomial model were developed to accelerate estimation. Leisen and Reimer (1996, LR) and Tian (1993) extended the CRR binomial model by changing the function of the increment and decrement of stock price at each step. The Adaptive Mesh Method proposed by Figlewski and Gao (1999), builds a strip of finer lattice over a substrate tree and can be adapted to a wide variety of options. Staunton (2005) proposed several modifications to the LR model by adapting Curtailed Ranges and Richardson extrapolation. This modified LR model was found to be more efficient than other approximation methods considered. Joshi (2009) used standard deviation truncation, smoothing and Richardson extrapolation, which led to better performance than the modified LR model proposed in Staunton (2005). The Chen and Joshi (2012, Chen-Joshi) incorporated tolerance truncation, Black-Scholes smoothing and Richardson Extrapolation. This model constituted the premier model relative to 220 lattice permutations evaluated previously in Joshi (2009). In this paper, we compare our accelerated binomial model with Chen-Joshi and show that our model performs better than this leading benchmark tree model.

This paper outlines a lattice search algorithm to rapidly locate the early exercise node in each column of a binomial model. This ideally should dispense with the standard blanket test to repetitively compare the relative magnitudes of the exercise and holding values at each node. The

methodology in this paper follows the literature that has focused on discerning a continuous early exercise boundary so that the tree can be cleanly delineated between exercising and holding regions. Knowing in advance the vicinity of the optimal exercise boundary reduces greatly the quantum of computation. Kim and Byun (1994) specify the optimal exercise boundary for an American put option written on a non-dividend-paying stock. Curran (1995) subsequently extended the Kim and Byun (1994) approach to American put options with continuous dividend yields, y, and proposed the Diagonal Method which can efficiently locate the early exercise boundary. A stipulation of this model however is that the risk-free interest rate r necessarily exceeds (is inferior to) the dividend yield y for puts (calls). In reality, this implies the model cannot be guaranteed to work in every instance. Basso et al. (2002, 2004) developed the insights of Kim-Byun-Curran to discern a binomial approximation to the optimal exercise boundary. Areal and Rodrigues (2013) use the early exercise boundary theory of Curran (1995) to accelerate the binomial model for pricing American options with discrete dividends. In this paper, we open the Kim-Byun-Curran boundary theory to include the wider subset of parameter inputs where the American put (call) valuation is not constrained by $r \geq y$ ($r \leq y$). Furthermore, we propose an accelerated CRR model, incorporating the intelligent lattice search algorithm based on revamped optimal boundary theory as well as two acceleration technologies, for efficiently pricing American options with unrestricted continuous dividends.

The remaining paper is organized as follows. In Section 2, optimal exercise boundary theory is reviewed and extended to make practicable intelligent lattice search. The latter we introduce. In Section 3, we describe two acceleration technologies: truncation and dynamic memory which are standard in the literature. In Section 4, we integrate two acceleration technologies with the

intelligent lattice search algorithm to accelerate the CRR pricing process. In Section 5, we evaluate

the relative performances of our accelerated CRR model, with varying benchmarks including a

conventional CRR model implementation, other tree models from a recent literature, and several

commonly used analytical formulae. We develop several metrics that capture the relative

efficiencies in terms of option pricing, delta estimation, and implied volatility estimation. In

Section 6, we tease out conclusions. The Appendix includes the proof of the Propositions and

Theorems developed in Section 2 and an Excel VBA implementation of our accelerated CRR

model.

Methodology

Extending Kim-Byun-Curran: The Optimal Exercise Boundary Adapted for Unrestricted Continuous Dividends

Consider an American put option with an initial stock price S, strike price X, time to maturity T,

risk-free interest rate r, continuous dividend yield y and volatility σ, priced by a n-step binomial

tree. We define the number of time steps as i, the number of upward steps as j, and $S_{(i,j)}$ and $V_{(i,j)}$

as the stock price and option price respectively.[3] Kim and Byun (1994) give the definition of the

stopping region S, the continuation region C and the optimal exercise state B(i): all nodes in a

binomial tree are divided into two groups which fall on two different regions. The *stopping region*

S is a series of nodes whose option values are equal to their exercise values, which can be given

by:

---

[3] The lattice structure is presented in Figure A2 complete with (i, j) mapping.

$$S \equiv \{(i,j)|\ V_{(i,j)} = X - S_{(i,j)}\} \tag{2.1}$$

The nodes that belong to the stopping region are called *stopping nodes*. In addition, the *continuation region C* is a set of nodes where options are worth more if they are held instead of exercised, which can be defined as:

$$C \equiv \{(i,j)|\ V_{(i,j)} > X - S_{(i,j)}\} \tag{2.2}$$

In this case, the option values are equal to their *holding values* below:

$$V_{(i,j)} = \left[pV_{(i+1,j+1)} + (1-p)V_{(i+1,j)}\right]R^{-1} \tag{2.3}$$

p represents the risk-neutral probability of an upward movement and $R = exp(rT/n)$. The nodes that belong to the continuation region are called *continuation nodes*. $\boldsymbol{I}$ is defined as the set of time steps at which there is at least one stopping node, which can be given by:

$$\boldsymbol{I} \equiv \{\ i\ |(i,j) \in \boldsymbol{S}, 0 \le\ j \le i \le n\} \tag{2.4}$$

Based on $\boldsymbol{I}$, $\boldsymbol{I_0} \equiv \{\ i\ |(i,j) \in \boldsymbol{S}, 0 \le\ j \le i \le n-1\}$ (from the penultimate column back) is proposed, which will be used later. In addition, the *optimal exercise state* B(i) represents the *biggest* j at the ith column when (i, j) $\epsilon\ \boldsymbol{S}$ for i $\epsilon\ \boldsymbol{I}$, which can be defined as:

$$B(i) \equiv max\{\ j\ |(i,j) \in \boldsymbol{S}, i \in \boldsymbol{I}\} \tag{2.5}$$

Therefore, (i, B(i)) is a series of *optimal exercise nodes* for i $\epsilon\ \boldsymbol{I}$, which constitutes the *optimal exercise boundary*. Kim and Byun (1994) propose three Propositions and two Theorems relating to the continuity of the optimal exercise boundary. Curran (1995) extends the optimal exercise boundary theory for American put options with continuous dividend yields y where r ≥ y. He then applied this to American calls by invoking McDonald and Schroder (1998) who proposed put-call symmetry conditions for American options where:

$$C\ (S,X,r,y,T,\sigma)\ =\ P\ (X,S,y,r,T,\sigma) \tag{2.6}$$

The optimal exercise boundary can also be applied for pricing American call options with y where $r \le y$. The Kim-Byun-Curran construction is augmented here by pricing American options without imposing any restrictions on y. The methodology developed in this paper departs from Kim-Byun-Curran by locating/initializing in the penultimate column the seed node consistent with the optimal exercise boundary. The key intuition to the proposed approach relates to properties of the boundary. If $r \ge y$ the boundary is always continuous for put options, and this simplifies the demarcation of the stopping and continuation regions up to and including the final column. Otherwise, when $r <$ y, a break of the early exercise boundary between the last column and the penultimate column for an American put option can occur. In Figure 2.1, S represents the optimal exercise node (i, B(i)) for $i \in \boldsymbol{I}$ and C represents (i, B(i) + 1) for $i \in \boldsymbol{I}$, which is the *first* continuation node at each column from the bottom. The heavy solid lines represent the continuous optimal exercise boundary and the heavy dashed lines represent the discontinuation. The upper binomial tree (Figure 2.1 (a)) follows that of Curran (1995, p.13), where S = 100, X = 100, T = 1, r = 0.05, y = 0, σ = 0.3 and n = 10. It is clear that the boundary is continuous where $r > y$. The lower binomial tree (Figure 2.1(b)), however, has the same set of parameters as Curran's except y = 0.07 so that $r < y$ (r = 0.05). In this instance, the stipulation that $r \ge y$ set out by Curran (1995) is violated. The impact of this violation is illustrated in the lower binomial tree. When $r < y$, the optimal exercise boundary is only continuous from the penultimate column back, while a discontinuous boundary between the penultimate and last column manifests itself.
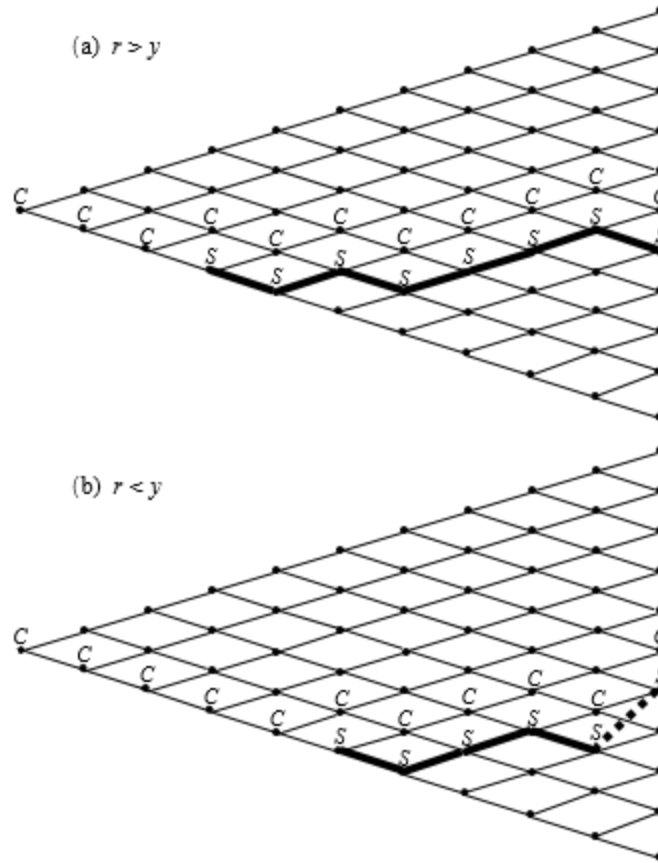
Figure 2.1: A discontinuous optimal exercise boundary when r < y

The main insight here is that the region of discontinuation is limited to merely the final and penultimate columns. If we exclude the final column, three Propositions and two Theorems of the optimal exercise boundary developed by Kim and Byun (1994) can be extended to American put options with unrestricted continuous dividends. These revamped Propositions/Theorems are developed in the Appendix. The restrictions imposed on dividends by Curran (1995) are also relaxed by seeding the continuous boundary from the penultimate column. The Propositions and Theorems in the Appendix can be used to identify the optimal exercise boundary for an American put option with unrestricted continuous dividend yield. We assert that *from the penultimate column back, the new optimal exercise state B(i - 1) is always equal to the old optimal exercise state B(i)*

minus a value of 1 or 0 as time to expiry increases.[4] Discerning the adjustment behavior of the optimal exercise boundary permits an elaboration of an intelligent lattice search algorithm. Figure 2.2 shows the simple mechanism steering the intelligent lattice search, where this seed value at column n - 1 has been confirmed as an integer value of k. The optimal exercise state B(i - 1) is invariably equal to either B(i) or B(i) - 1. Therefore, we efficiently locate the boundary by verifying no more than one node at each column from antepenultimate column back. Unlike Kim-Byun-Curran, the intelligent lattice search technique is anchored by reference to the penultimate column from where the recursion is initiated. This involves some further computation as the exercise condition of additional nodes must be verified, (no more than n) at the penultimate column. The extra computation workload is comparatively trivial - especially for a large number of steps. In so doing, the restriction imposed on dividend yields can be relaxed and the spectrum of feasible parameter value inputs can be extended significantly.

---

[4] Moving back through time – consistent with backward induction. Also, McDonald and Schroder (1998) put-call symmetry permits us to generalize to the case of the American call options.
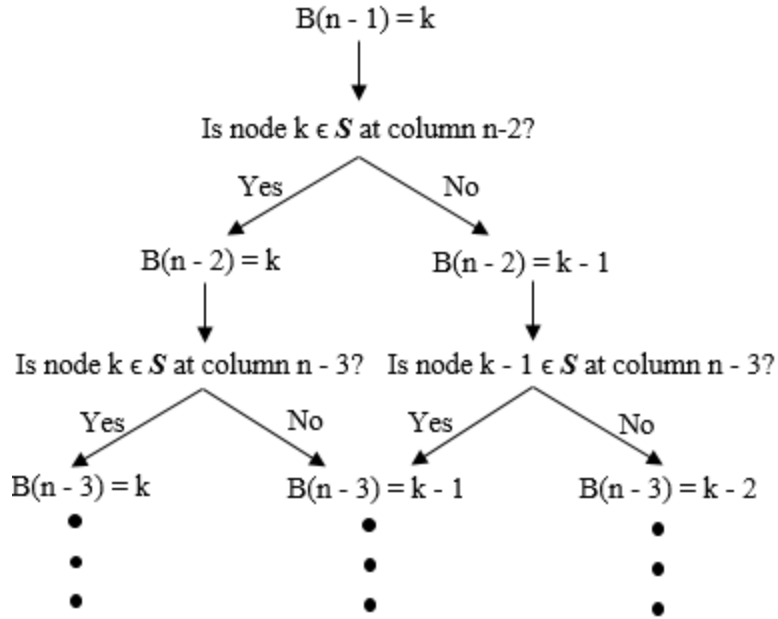
Figure 2.2: An intelligent lattice search algorithm

Two Acceleration Technologies: Truncation and Dynamic Memory

As noted by Curran (1995), there are subtrees within the binomial tree, where the nodes exercise no influence on the present value of the option. In the classic backward induction approach set out by CRR, all the nodes are given an equal weight in the estimation. This blanket method implies that large regions of the binomial tree are incorporated into the estimation but do not materially exert any impact in terms of the ultimate valuation. As noted by Curran (1995), this creates scope to apply the Diagonal Method where redundant nodes can be isolated and eliminated for the purposes of estimation.[5] Acceleration is obtained by locating and truncating a portion of redundant stopping nodes (The Redundant-stopping region enclosed by heavy dashed lines) and all redundant zero-value nodes (The Zero-value zone enclosed by light dashed lines), which are shown in Figure

---

[5] Consistent with Curran (1995) the first passage probabilities are not applied here. No increase in computational efficiency from that technique has been obtained.

2.3. Significantly, some redundant stopping nodes (hollow nodes) are not truncated because more complex programming would be required for recognizing those nodes and this likely would produce slower computational speed. On balance, when the extra programming cost against speed is taken into account, it was considered suffice to truncate the Redundant-stopping region and the value of the redundant hollow nodes to make pricing more efficient.

Making use of Dynamic Memory can produce an important reduction in computational cost. In Figure 2.4, we try to reveal how computer memory can be used more efficiently. A conventional two-dimensional static n-step binomial model requires $(n + 1)(n + 2) / 2$ nodes to be memorized (Figure 2.4 (a)). Broadie and Detemple (1996) and Haug (2007) propose using a one-dimensional dynamic binomial tree (Figure 2.4 (b)). This approach takes the option values at the last column and stores them in a dynamic vector $Opt(j)$ for $j = 0, 1, \ldots , n$. After moving one step back, the values in the re-dimensioned $Opt(j)$ for $j = 0, 1, \ldots, n - 1$ will be replaced by the option values of the corresponding nodes at the penultimate column (Figure 2.4 (c)). Similarly, the values of $Opt(j)$ for $j = 0, 1, \ldots, k - 1$ at k^{th} column will always be substituted by the option values at $(k - 1)^{th}$ column for $1 \leq k \leq n$. Therefore, a dynamic binomial tree only requires $n + 1$ contemporaneous storage spaces.
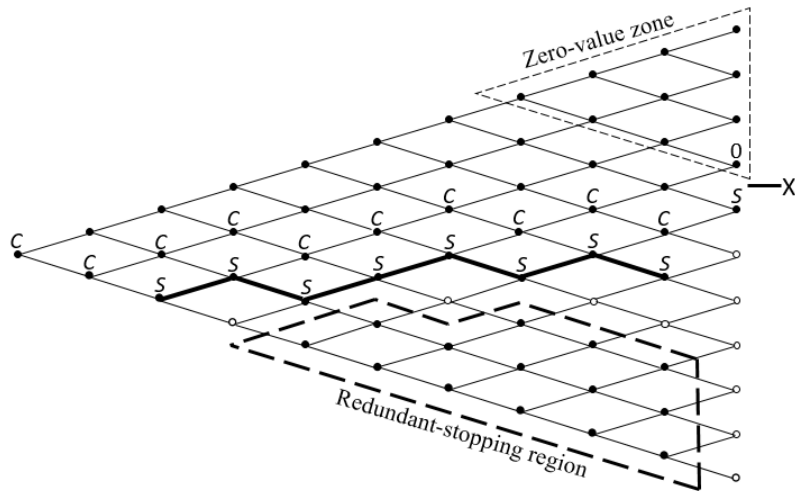
Figure 2.3 Truncating zero-value zone and redundant-stopping region

(a) Static Memory

(b) Dynamic Memory

(c) Substitution

18
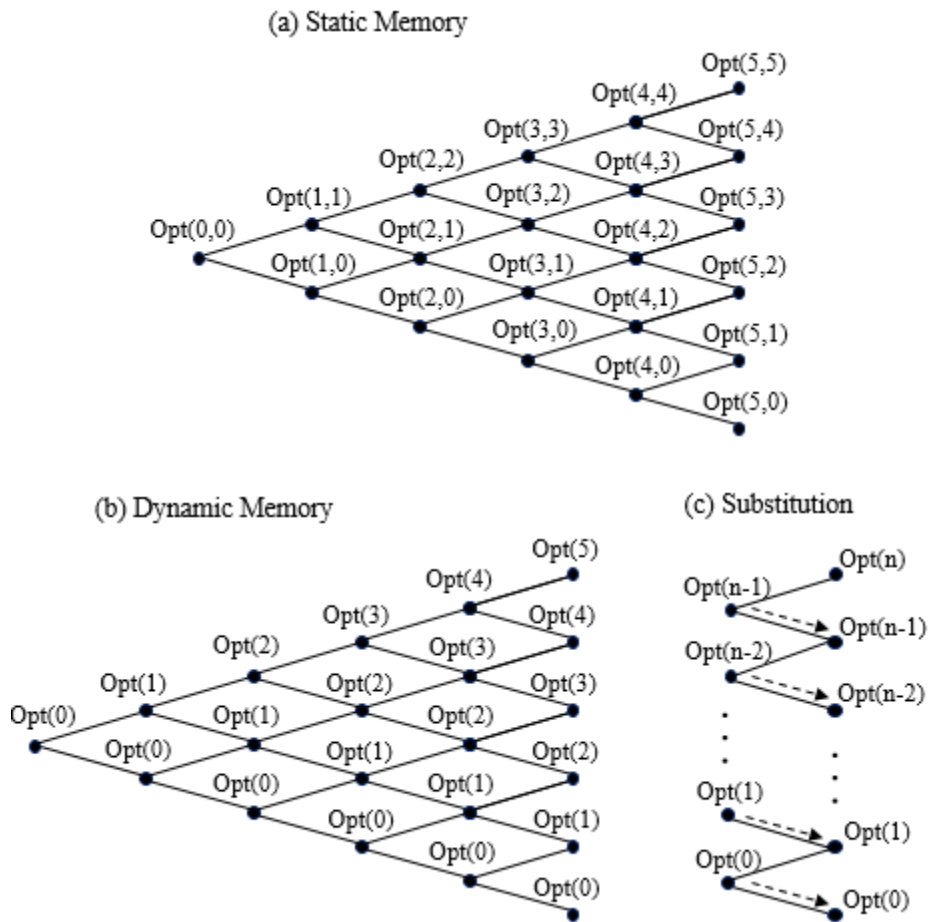
Figure 2.4: A one-dimensional dynamic CRR tree

The Efficient Pricing Process of an Accelerated CRR model: Applying Intelligent Lattice Search Algorithm, Truncation and Dynamic Memory

In this section, we demonstrate how an accelerated CRR model, incorporating an intelligent lattice search algorithm, dynamic memory and truncation, can efficiently price an American put option. We explain how the intelligent lattice search algorithm can be used to efficiently locate the optimal exercise boundary. We employ a one-dimensional dynamic binomial tree with truncation for an American put option. The rationale for presenting the sequence of steps involved in Figure 2.5 relates to teasing out a viable framework appropriate for coding. The CRR binomial tree in Figure 2.5 has the same set of parameters as Figure 2.1 (b), which also follows Curran (1995), where S = 100, X = 100, T = 1, r = 0.05, σ = 0.3 and n = 10. We differ by setting the dividend yield, y = 0.07. Since r < y, the stipulation that r ≥ y advanced by Curran (1995) is deliberately violated. The nodes falling along each row share the same underlying asset price and by extension exercise value. These are given in the final two columns. Array mappings in Figure 2.5 are set out consistent with the one-dimensional dynamic binomial tree depicted in Figure 2.4 (b). Each node outside the truncated regions can be identified by its series number. For illustrative purposes, Nodes with series numbers enclosed by no-fill circles (e.g. ⓪) represent the stopping value. Nodes with series numbers enclosed by a black-fill circle (e.g. ⬤) represent the optimal exercise nodes (first stopping nodes), B(i), at i^{th} column. Continuation nodes are in contrast denoted by square brackets (e.g. [0]). The nodes with series numbers followed by question marks, are those that are minimally investigated by checking the exercise condition (e.g. ⬤? and [0]?). The optimal boundary check is efficiently reduced to determine the status of these nodes – no more than one single node at each

column from the ante penultimate column. The pathway of these checks is shown by the presence of question marks. In this regard, clear efficiency gains are discernible vis-à-vis more common systems of blanket checking. The nodes represented by X without option values and series numbers in Figure 2.5 are redundant and can be truncated to also increase efficiency.

To elaborate the sequence of steps involved in the optimization process, we begin with the last column. The expression in Equation (2.7) below is used to ascertain the first non-zero/stopping node at the maturity:

$$j = [(ln(X/S)/ln(u) + n)/2] \qquad (2.7)$$

where [.] locally means the largest integer lower than its argument and $u = exp(\sigma\sqrt{T/n})$. Logically, for a put the nodes beneath the optimal exercise node at the maturity belong to the stopping region. Their exercise values are calculated and assigned to the appropriate nodes.[6] In Figure 2.5, j is initially calculated to be 4 using the expression in Equation (2.7), which implies that the optimal exercise node at the maturity is node ❹. Then we assign respectively the exercise values: 61.2749, 53.1841, 43.4028, 31.5778 and 17.2823, from node ⓪ to node ❹ at column 10.

Once these values are established, they then are used to enable backward induction leading to the penultimate column. The exercise condition is investigated somewhat more painstakingly from the first node beneath the zero-value zone by calculating and comparing the exercise values and holding values of the nodes until the optimal exercise node is confirmed. Then the respective exercise values are calculated again and assigned to the optimal exercise node and the node immediately below it while consecutive holding values are calculated again and assigned to the

---

[6] All assignments of the option values to nodes should strictly follow the order, from the bottom to the top, consistent with the dynamic memory approach.

nodes lying between the stopping zone and the zero-value zone. A more exhaustive search routine is required for the penultimate array because *the boundary is not guaranteed to be continuous going from the final column to the preceding column.* In Figure 2.5, moving to column 9 we check the exercise condition from node [4] by comparing the exercise values relative to the holding values of the nodes until node ❷ which is ultimately confirmed as the optimal exercise node. The respective exercise values, 48.5252 and 37.7705, are assigned to node ① and to node ❷. The respective holding values, 24.7949 and 9.1864, are assigned to node [3] and node [4]. By locating and verifying the optimal exercise value, the seed value of the continuous portion pertaining to the optimal exercise boundary, B(n - 1) is also identified. Node ❷ at column 9 provides the root value that initiates the continuous optimal boundary.

Then we move to column 8. Node ❷ should be initially inspected (the uncertain node) since it has the same series number as the optimal exercise node ❷ in column 9. We check its exercise condition and find that it is determined as a stopping node. This indicates that it is the optimal exercise node at this column. Thereafter, exercise values are assigned to it and the nodes immediately below it, and respective holding values are assigned to the nodes lying between the stopping zone and the zero-value zone at this column. Accordingly, we assign respective exercise values, 43.4028 and 31.5778, to node ① and node ❷ in column 8. The respective holding values, 17.4373 and 4.8830, are by default assigned to node [3] and node [4]. When we move to column 7, we check node [2] which has the same series number as the optimal exercise node ❷ in column 8 and find that node [2] is a continuation node, which means the node immediately below it, node ❶, is the optimal exercise node in column 7. Accordingly, we assign the exercise value 37.7705 to node ❶ and assign respective holding values, 24.8667, 11.5319 and 2.5956, to node [2], [3],

[4] in column 7. This option valuing process is iterated until we move to column 4 by virtue that the value [0] associated with the last node in column 4, is also verified as a continuation node, which triggers the exit mechanism of the algorithm. The optimal exercise boundary searching stops. All nodes in the remaining columns (column 3, 2, 1, 0) are continuation nodes. From column 4 back, we assign automatically holding values to each node in each column.

We also present a closed-loop optimal exercise boundary search routine in Figure 2.6. When the previous exercise state B(i) = k, the new optimal exercise state either remains unchanged or minus 1 (B(i − 1) = k or k − 1). k should not breach 0 or exceed the step size associated with any given column in the tree. Otherwise, the exit feature is primed to trigger. In Table 2.1, a direct comparison using the accelerated CRR tree vis-à-vis a standard CRR tree is made. The reported exhibit is based on the sample tree outlined in Figure 2.5. In a conventional CRR tree, we are normally obliged to estimate and assess the exercise value relative to holding value for 55 nodes. 11 terminal nodes are by default exercise values. The accelerated CRR tree only requires 8 direct comparisons to be made of the exercise value relative to the holding value. 9 nodes and 25 nodes are automatically assigned exercise values and holding values independently given that the early exercise boundary can be used to efficiently demarcate. Otherwise, 24 redundant nodes are truncated which incorporates hardly any processing costs.

0	1	2	3	4	5	6	7	8	9	10	Exercise Value	Underlying Price
									X		0.0000	258.2307
								X			0.0000	234.8590
							X		X		0.0000	213.6025
						X		X			0.0000	194.2699
					X		X		X		0.0000	176.6871
				X		X		X			0.0000	160.6956
			0.3898 [4]		X		X		X		0.0000	146.1515
		1.6433 [3]		0.7334 [4]		X		X			0.0000	132.9237
	4.0259 [2]		2.7516 [3]		1.3797 [4]		X		X		0.0000	120.8931
7.5286 [1]		6.1410 [2]		4.5371 [3]		2.5956 [4]		X			0.0000	109.9514
11.9473[0]		10.6532[1]		9.1540 [2]		7.3327 [3]		4.8830 [4]		X	0.0000	100.0000
	15.9121[0]		14.6874[1]		13.2653[2]		11.5319[3]		9.1864 [4]?		9.0507	90.9493
		20.6467[0]		19.6498[1]		18.5625[2]		17.4373[3]		17.2823❹	17.2823	82.7177
			26.0364[0]		25.4009[1]		24.8667[2]?		24.7949[3]?		24.7689	75.2311
				31.8493[0]?		31.6017[1]?		31.5778❷?		31.5778③	31.5778	68.4222
					37.7705⓿?		37.7705❶		37.7705❷?		37.7705	62.2295
						43.4028⓿		43.4028①		43.4028②	43.4028	56.5972
							X		48.5252①		48.5252	51.4748
								X		53.1841①	53.1841	46.8159
									X		57.4213	42.5787
										61.2749⓿	61.2749	38.7251
Column 0	1	2	3	4	5	6	7	8	9	10		

Figure 2.5: An efficient American put option pricing process

i = n-1

B(i) = k

i = i - 1   k = i or i = 0 ? —Yes→ i = i - 1 and k = k - 1

No   Exit   No

Yes—Is node k ∈ S at column i - 1? —No→ k = 0 ?

Yes

Exit
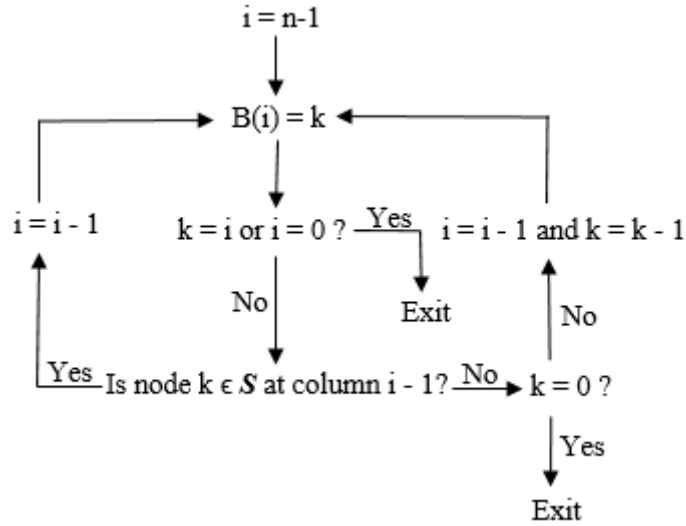
Figure 2.6: The closed-loop optimal exercise boundary search routine

Type of Nodes	Number of Nodes	
	Accelerated CRR	Standard CRR
Truncated Nodes	24	0
Stopping Nodes	9	11
Continuation Nodes	25	0
Uncertain Nodes	8	55
Total Nodes	66	66

Table 2.1: Comparing the computational workload associated with the Accelerated CRR tree and a Standard CRR tree.

Numerical Results

Numerical results can be divided into three sections: In this first section, we show how option pricing efficiency can be improved by applying dynamic memory, truncation, and intelligent lattice search sequentially to a standard CRR tree. The most accelerated CRR model combines intelligent lattice search, dynamic memory and truncation together. In the second section, we compare the efficiency of our most accelerated CRR model to a standard CRR model, to a leading benchmark tree Chen and Joshi (2012), and to four popular analytical formulae. These comparisons are made relative to both option pricing and delta estimation. In the final section, we compare the accelerated CRR model to Chen-Joshi, and to four analytical formulae for

implied volatility estimation. All reported results are obtained using Excel VBA. A DELL Latitude E5470 with Intel's Core i3 processors ran these algorithms and models.

In the first section, we gauge successively how improvements in estimation efficiency can be introduced by using dynamic memory, intelligent lattice search, and truncation, where the initial baseline tree is a standard CRR tree[7]. Benchmark values for American call and put option samples are obtained from Broadie and Detemple (1996). In addition, in order to test the intelligent lattice search algorithm, two sets of parameters with r > y (r < y) for call (put) are expressly selected, which violate the restrictions imposed by Curran (1995) on continuous dividend yields. In Table 2.2, CRR baseline represents a standard two-dimensional static CRR tree. CRR_Dyn are accelerated purely by employing a one-dimensional dynamic tree. CRR_Dyn_Bound augments the dynamic binomial tree by using the intelligent lattice search algorithm. CRR_Dyn_Bound_Trun represents the most accelerated binomial model which comprehensively applies intelligent lattice search, dynamic memory and truncation. The computational times are presented using a mm:ss.00 format, located under the corresponding option values. We found that all four binomial models with the same set of parameters have resolutely identical results, which in turn are also consistent with the benchmark values. The acceleration effects can be gauged by noting how estimation time is reduced - moving from the baseline. Replacing the two-dimensional static tree by a one-dimensional dynamic tree saved almost half of the computational time. This pales in comparison to the acceleration effect of applying intelligent lattice search algorithm, which produces improvements in speed by at least one order of magnitude. Then the application of truncation technology further speeds up the computation several times. With the number of steps increasing, the effect of accelerations becomes more obvious. The baseline binomial model took more than 18 minutes at a 15,000-

---

[7] This approach to pricing is by far the slowest. It is also typically the most common method introduced in text books.

step size to complete. The most accelerated CRR tree (CRR_Dyn_Bound_Trun) took only 2.30 seconds for the call and 5.66 seconds for the put. The improvement in estimation time is noteworthy and accuracy has not been compromised relative to the standard CRR tree.

Option Param	n	CRR	CRR_Dyn	CRR_Dyn_ Bound	CRR_Dyn_ Bound_Trun	Broadie &Detemple
Amer Call S = 80 X = 100 T = 3 r = 0.07 σ = 0.30 y = 0.03	1000	12.1456 00:05.07	12.1456 00:02.70	12.1456 00:00.09	12.1456 00:00.02	
	3000	12.1455 00:43.09	12.1455 00:22.13	12.1455 00:00.67	12.1455 00:00.33	
	5000	12.1452 01:58.79	12.1452 01:00.91	12.1452 00:01.82	12.1452 00:00.69	
	10000	12.1455 07:58.54	12.1455 04:02.83	12.1455 00:06.34	12.1455 00:01.18	
	15000	12.1453 18:06.41	12.1453 09:23.60	12.1453 00:13.88	12.1453 00:02.30	12.145
Amer Put S = 100 X = 100 T = 0.5 r = 0.03 σ = 0.30 y = 0.07	1000	9.2486 00:05.07	9.2486 00:02.73	9.2486 00:00.12	9.2486 00:00.02	
	3000	9.2499 00:42.70	9.2499 00:22.08	9.2499 00:00.84	9.2499 00:00.39	
	5000	9.2502 01:59.00	9.2502 01:01.25	9.2502 00:01.87	9.2502 00:00.73	
	10000	9.2504 07:57.43	9.2504 04:07.90	9.2504 00:07.25	9.2504 00:02.31	
	15000	9.2505 18:51.19	9.2505 09:34.87	9.2505 00:16.67	9.2505 00:05.66	9.251

Table 2.2: Acceleration effect comparison among dynamic memory, truncation, and intelligent lattice search

Before moving to the second section, there is some initial preparatory work relating to generating a large number of sample options parameters. We use these to determine the level of error relative to a benchmark ("true value") and the time required for estimation is also recorded. We additionally sketch out a number of delta estimation methods. Following Broadie and Detemple (1996), we design a uniform distribution of parameters inputs for S, T, r, y, σ and PutCall to generate 2,500 American options. The spot price S was given to be uniformly

distributed between 70 and 130. The exercise price X was fixed as a constant at a value of 100. Time to maturity T, with probability of 0.75 was uniform between 0.1 and 1 years. A probability of 0.25 was attributed to maturity being randomly between 1 and 5 years. The riskless rate r was uniformly distributed between 0 and 0.1 with a probability of 0.8 and 0 generated with the residual probability of 0.2. The dividend rate y was uniform between 0 and 0.1. Volatility σ was distributed uniformly between 0.1 and 0.6. There was also a random 0.5 probability of the option being a call or put. Consistent with Broadie and Detemple (1996), the main error measure, root mean squared relative error (RMSRE), is defined as:

$$\text{RMSRE} = \sqrt{\frac{1}{m}\sum_{i=1}^{m} e_i^2} \tag{2.8}$$

where m is the number of options and $e_i = \frac{\widehat{C}_i - C_i}{C_i}$ where C_i and $\widehat{C}_i$ is the true and estimated value of the option respectively. The true value C_i is generated using a 15,000-step CRR model. 170 of the 2,500 American options with extremely-small true value ($C_i < 0.50$) are excluded. The residual number of American options involved in the valuation is 2,330 (m = 2,330). The time consumption measure (Time) represents the average execution time (seconds) for pricing per American option, which can be calculated as:

$$\text{Time} = \frac{Total\ Execution\ Time}{m} \tag{2.9}$$

where m is the number of options. For delta estimation, the delta of the tree model, including a standard CRR, our accelerated CRR, and Chen-Joshi, are estimated using:

$$\Delta_{CRR} = \frac{V_{(1,1)} - V_{(1,0)}}{S_{(1,0)} - S_{(1,1)}} \tag{2.10}$$

where the numerator is the difference between the option value of the upper node and lower node at the end of the first period, and the denominator is the difference between the stock price of these two nodes. For analytical formulae, however, the delta is calculated as:

$$\Delta_{anal} = \frac{C(S+\varepsilon) - C(S-\varepsilon)}{2\varepsilon} \tag{2.11}$$

where ε is a perturbation introduced for the spot price S, and numerator is the difference between the option value estimated with the spot price $S + \varepsilon$ and $S - \varepsilon$ using analytical formulae.

In the second section, we first compare the 2,330 options sample for pricing and delta estimation with a view to teasing out the relative efficiency of a standard CRR model (CRR) vis-à-vis our accelerated CRR model (Accel CRR). The latter introduces intelligent lattice search, dynamic memory and truncation. For both pricing options and estimating delta, shown in Table 2.3, as the number of steps increase, the estimation error (RMSRE) generated by CRR and our Accel CRR decreases while the execution time (Time) increases. We found that Accel CRR always generates identical RMSRE as CRR at different step size but with much less Time, which implies that the pricing process is effectively accelerated without disturbing the accuracy. The "Multiple of speed" shows how many times Accel CRR is faster than CRR attaining the same estimation accuracy. From 50 to 1,000 steps, Accel CRR model can be from 7 to 220 (182) times faster than CRR in option pricing (estimating delta).

To provide a yardstick relative to a more recent literature, we replicate Chen-Joshi and run it with tolerance level of 1E-05 to estimate 2,330 generated sample options. The comparison between our Accel CRR and Chen-Joshi is demonstrated in Table 2.4. To make a direct comparison, the number of steps of the two models are selected to achieve a similar level of accuracy (RMSRE) so that the efficiency can be easily juxtaposed according to execution time (Time). For each column, Accel CRR and Chen-Joshi generate a similar RMSRE but expend different amounts of Time. The "Multiple of speed" indicates how many times Accel CRR is faster than Chen-Joshi. For option pricing, Accel CRR is roughly 1.5 to 2 times faster than Chen-Joshi consistent with a similar level of accuracy. This differential is amplified by an order of 2 to 3 times for delta estimation.

Next, we compare our accelerated CRR model with four analytical formulae briefly alluded to in the literature review: BAW, Bjerksund93, Bjerksund02, and Ju-Zhong. We reuse the 2,330 option parameter sets generated previously. In Table 2.5 and Table 2.6, the results generated from analytical formulae are deliberately placed under the results of Accel CRR that have approximately the same RMSRE. The execution time (Time) with similar levels of accuracy (RMSRE) are investigated. The "Multiple of speed" tentatively maps out how many times our Accel CRR is faster or slower relative to the analytical formulae linked by approximate levels of accuracy. For option pricing (Table 2.5), Accel CRR is roughly 1.5 times faster than BAW. In contrast, Accel CRR is discernibly slower than Bjerksund93 and Bjerksund02, but almost as fast as Ju-Zhong for similar levels of error. For delta estimation (Table 2.6), we find that Accel CRR is about 2 and 2.5 times faster than respectively Ju-Zhong and BAW to obtain the same level of accuracy. It is comparable to Bjerksund02 but 0.3 (1 - 0.68) times slower than Bjerksund93. Accel CRR has an obvious advantage in that it provides varying levels of accuracy with different step size. Our accelerated CRR model provides a full spectrum of choice to practitioners varyingly tasked with pricing, repricing and hedging accuracy criteria. Each analytical formula can only provide one combination of speed and accuracy.

		Steps	50	100	200	300	400	500	600	700	800	900	1000
Option Pricing	Accel CRR	RMSRE	7.65E-03	3.61E-03	1.79E-03	1.08E-03	8.05E-04	6.85E-04	5.71E-04	4.64E-04	4.10E-04	3.67E-04	3.85E-04
		Time (T_a)	1.77E-03	2.91E-03	3.42E-03	6.72E-03	8.06E-03	9.18E-03	1.01E-02	1.25E-02	1.45E-02	1.75E-02	2.20E-02
	CRR	RMSRE	7.65E-03	3.61E-03	1.79E-03	1.08E-03	8.05E-04	6.85E-04	5.71E-04	4.64E-04	4.10E-04	3.67E-04	3.85E-04
		Time (T_o)	1.27E-02	4.84E-02	1.72E-01	3.79E-01	6.60E-01	1.0433	1.4884	2.0246	2.6490	3.8083	4.8481
	Multiple of Speed (T_0/T_a)		7.19	16.61	50.34	56.37	81.97	113.59	147.32	162.16	182.77	217.75	220.37
Delta Estimation	Accel CRR	RMSRE	7.34E-03	3.55E-03	1.76E-03	1.09E-03	8.32E-04	7.14E-04	5.92E-04	4.95E-04	4.40E-04	3.95E-04	4.01E-04
		Time (T_a)	1.62E-03	2.31E-03	3.71E-03	5.39E-03	8.04E-03	9.26E-03	1.13E-02	1.36E-02	1.62E-02	1.88E-02	2.18E-02
	CRR	RMSRE	7.34E-03	3.55E-03	1.76E-03	1.09E-03	8.32E-04	7.14E-04	5.92E-04	4.95E-04	4.40E-04	3.95E-04	4.01E-04
		Time (T_o)	1.13E-02	4.70E-02	1.69E-01	3.81E-01	6.65E-01	1.0371	1.4815	1.9546	2.5505	3.2325	3.9707
	Multiple of Speed (T_0/T_a)		7.01	20.37	45.57	70.67	82.66	111.98	130.56	143.26	157.42	172.12	182.08

Table 2.3: Comparing Accel CRR and CRR in option pricing and delta estimation

Option Pricing	Accel CRR	Steps	30	120	200	300	400	500	600	700	900	1100	
		RMSRE	1.08E-02	2.88E-03	1.79E-03	1.08E-03	8.05E-04	6.85E-04	5.71E-04	4.64E-04	3.67E-04	3.13E-04	
		Time (T_a)	1.46E-03	3.22E-03	3.42E-03	6.72E-03	8.06E-03	9.18E-03	1.01E-02	1.25E-02	1.75E-02	2.42E-02	
	Chen-Joshi	Steps	10	20	30		40	50	60	70	80	90	100
		RMSRE	1.14E-02	2.87E-03	1.39E-03		8.60E-04	6.26E-04	5.72E-04	4.58E-04	3.38E-04	3.06E-04	3.04E-04
		Time (T_0)	2.81E-03	4.76E-03	7.63E-03		1.12E-02	1.53E-02	2.06E-02	2.60E-02	3.27E-02	3.94E-02	4.79E-02
	Multiple of Speed (T_0/T_a)		1.92	1.48	1.50		1.39	1.66	2.04	2.08	1.87	1.80	
Delta Estimation	Accel CRR	Steps	20	80	150	200	250	300	400		500		
		RMSRE	1.76E-02	4.53E-03	2.23E-03	1.76E-03	1.47E-03	1.09E-03	8.32E-04		7.14E-04		
		Time (T_a)	1.17E-03	1.97E-03	3.09E-03	3.71E-03	4.60E-03	5.39E-03	8.04E-03		9.26E-03		
	Chen-Joshi	Steps	10	20	30	40	50	60	70	80	90	100	
		RMSRE	1.53E-02	4.28E-03	2.15E-03	1.56E-03	1.25E-03	1.08E-03	8.93E-04	7.92E-04	7.85E-04	7.64E-04	
		Time (T_o)	2.13E-03	4.10E-03	7.19E-03	1.13E-02	1.54E-02	2.07E-02	2.67E-02	3.50E-02	4.58E-02	5.51E-02	
	Multiple of Speed (T_0/T_a)		1.82	2.08	2.33	3.04	3.35	3.84	3.32		4.89		

Table 2.4: Comparing Accel CRR and Chen-Joshi in option pricing and delta estimation

	Steps	10	20	30	40	50	60	70	80	90	100	110	120
Accel CRR	RMSRE	3.56E-02	1.85E-02	1.08E-02	9.98E-03	7.65E-03	6.16E-03	5.58E-03	4.71E-03	4.11E-03	3.61E-03	3.16E-03	2.88E-03
	Time (T_a)	1.10E-03	1.31E-03	1.46E-03	1.62E-03	1.77E-03	1.99E-03	2.08E-03	2.54E-03	2.78E-03	2.91E-03	2.98E-03	3.22E-03
BAW	RMSRE			1.22E-02									
	Time (T_1)			2.24E-03									
Bjerksund93	RMSRE					7.66E-03							
	Time (T_2)					9.91E-04							
Bjerksund02	RMSRE							5.91E-03					
	Time (T_3)							1.38E-03					
Ju Zhong	RMSRE												2.86E-03
	Time (T_4)												3.10E-03
Multiple of Speed (T_i/T_a, i = 1, 2, 3, 4)				1.53		0.56		0.68					0.96

Table 2.5: Comparing Accel CRR and analytical formulae in option pricing

	Steps	10	20	30	40	50	60	70	80	90	100	120	140
Accel CRR	RMSRE	3.36E-02	1.76E-02	1.04E-02	9.58E-03	7.34E-03	5.99E-03	5.41E-03	4.53E-03	3.99E-03	3.55E-03	2.83E-03	2.40E-03
	Time (T_a)	1.03E-03	1.17E-03	1.36E-03	1.52E-03	1.62E-03	1.74E-03	1.82E-03	1.97E-03	2.04E-03	2.31E-03	2.44E-03	2.67E-03
BAW	RMSRE					7.38E-03							
	Time (T_1)					4.25E-03							
Bjerksund93	RMSRE					7.80E-03							
	Time (T_2)					1.03E-03							
Bjerksund02	RMSRE					7.25E-03							
	Time (T_3)					1.69E-03							
Ju Zhong	RMSRE											2.61E-03	
	Time (T_4)											4.91E-03	
Multiple of Speed (T_i/T_a, i = 1, 2, 3, 4)					0.68	2.63	1.03					1.92	

Table 2.6: Comparing Accel CRR and analytical formulae in delta estimation

In Figure 2.7 and Figure 2.8, we visualize the performance of our accelerated CRR model, Chen-Joshi, and four analytical formulae for option pricing and delta estimation. Better model performance can be achieved by reaching a closer proximity to the origin. Lower RMSRE and Time indicate higher efficiency. For option pricing (Figure 2.7), Bjerksund93 and Bjerksund02 perform well and efficiency is signalled in terms of their relative proximity to the origin. Accel CRR and Ju-Zhong foster similar levels of error with comparable levels of speed. BAW would appear furthest from the origin and we interpret this be consistent with a relatively poorer performance. Accel CRR provides varying combinations of accuracy and speed represented as a line in the graph while each analytical formula has only one combination shown as a single dot. The locus representing Chen-Joshi also contains different levels of accuracy but further from the origin than Accel CRR, which indicates a worse performance. For delta estimation (Figure 2.8), the triangular shape representing Bjerksund93 performs well in terms of proximity to the origin. The other three analytical formulae and Chen-Joshi are farther away from the origin, which means they are less efficient. Again, the locus mapped out by Accel CRR represents different combinations of accuracy and speed relative to the singular dots of analytical formulae. Also, these varying combinations are closer to the origin than Chen-Joshi which means Accel CRR has relatively better performance.

This final section focuses on the relative efficiency of Accel CRR, Chen-Joshi, four analytical formulae in estimating implied volatility (IV). We collected chain market price of all 1,152 live American call and put options on Apple, Inc. from Datastream reported on 8th April 2019 with a contemporaneous stock price of 200.1. The expiry dates span 18th April 2019, 17th May 2019, 21st June 2019, 19th July 2019, 16th August 2019, 20th September 2019, 18th October 2019, 17th January 2020, 19th June 2020, 18th September 2020, 15th January 2021, 18th June 2021 respectively. The strike price ranges from about 100 to 300 with a uniform interval of 5. The

dividend yield of 1.46% with the risk-free rate of 2.37%[8]. The implied volatility is estimated using the Bisection method with a lower and upper bound respectively of a = 0.1 and b = 1[9]. The true value of IV is generated using a 15,000-step CRR model combined with Bisection. 306 of the 1,152 Apple's options present with true IV < 0.1% or > 1, or strike price K < 100 or > 300, are excluded. The residual number of options involved in IV estimation is 846 (m = 846). In Table 2.7, we report the RMSRE and Time of IV estimation using Accel CRR, Chen-Joshi, and four analytical formulae, which is sketched out in Figure 2.9. Again, the sweet spot leans towards lower RMSRE coupled with lower Time leading to higher efficiency estimates of IV. In the graph, better model performance can be realized by reaching a closer proximity to the origin. Ju-Zhong represented by the circle is closest to the origin with best performance in IV estimation, followed by BAW that can achieve the same level of accuracy (RMSRE) with a shorter time than Accel CRR. Bjerksund93 and Bjerksund02 approximately fall on the locus representing Accel CRR, which indicate that they share similar levels of performance. Different from single data point characteristic of the analytical formulae, Accel CRR provides varying combinations of accuracy (RMSRE) and speed (Time) in IV estimation. Chen-Joshi also possesses this advantage although performs worse than Accel CRR as its locus is further from the origin than Accel CRR.

---

[8] The risk-free rate is defined as the annualized 3-month US Treasury bill rate at 30th April 2019 obtained from Datastream.

[9] Regarding to the Bisection method, please refer to Rouah and Vainberg (2007, p.9). Since the smallest and largest IV of the residual 846 options is found to reside between 0.18 and 0.98 respectively using a 15,000-step CRR model, we set up the upper and lower bound of 0.1 and 1 correspondingly.
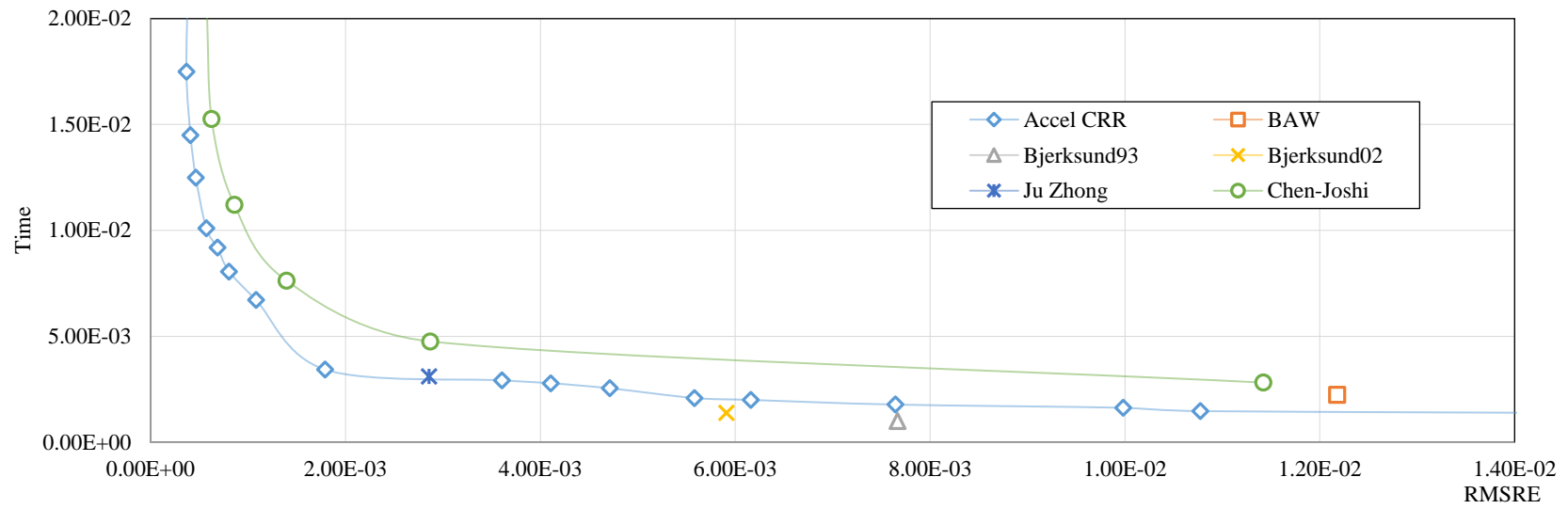
Figure 2.7: Comparing Accel CRR, Chen-Joshi, and analytical formulae in option pricing
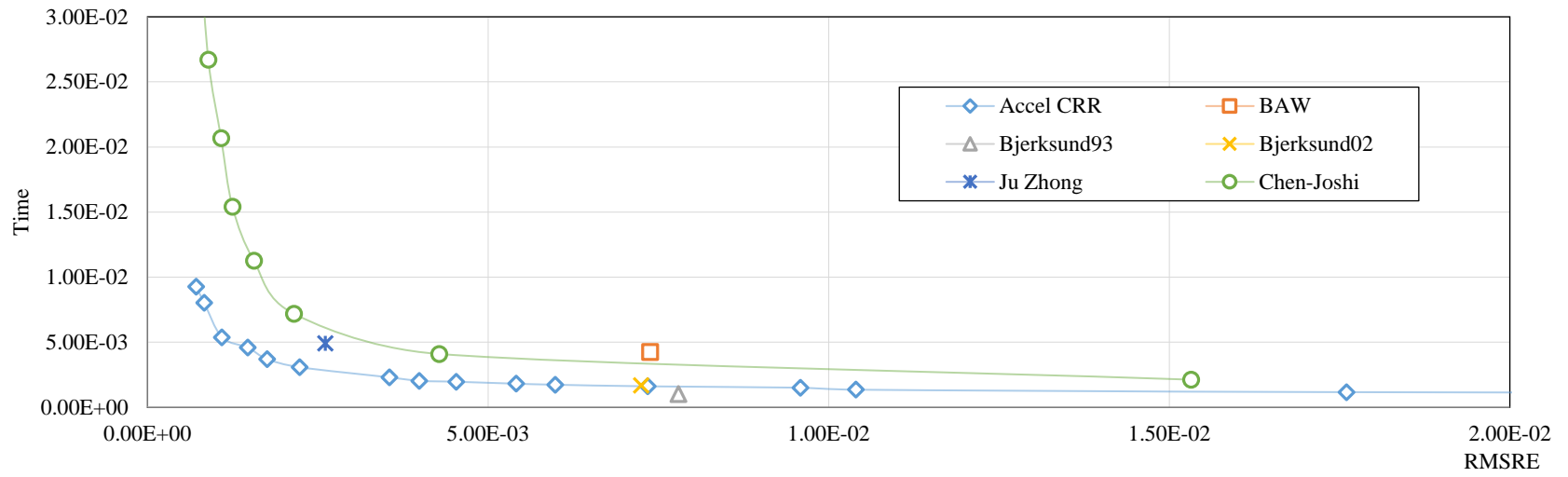
Figure 2.8: Comparing Accel CRR, Chen-Joshi, and analytical formulae in delta estimation

	AccelCRR_10Steps	AccelCRR_20Steps	AccelCRR_30Steps	AccelCRR_35Steps	AccelCRR_100Steps	AccelCRR_200Steps	AccelCRR_300Steps	AccelCRR_350Steps
RMSRE	2.99E-02	1.29E-02	9.59E-03	7.71E-03	2.69E-03	1.27E-03	8.51E-04	6.91E-04
Time	3.37E-03	5.41E-03	7.47E-03	8.81E-03	2.38E-02	5.14E-02	8.29E-02	1.01E-01

	Bjerksund 93	Bjerksund 02	ChenJoshi_10Steps	BAW	JuZhong	ChenJoshi_20Steps	ChenJoshi_30Steps	ChenJoshi_40Steps	ChenJoshi_50Steps
RMSRE	4.59E-02	5.78E-02	9.77E-03	8.42E-03	2.14E-03	2.33E-03	1.27E-03	9.24E-04	7.01E-04
Time	1.64E-03	6.02E-03	1.67E-02	4.18E-03	4.83E-03	4.40E-02	8.61E-02	1.41E-01	2.14E-01

Table 2.7: Comparing Accel CRR, Chen-Joshi, and analytical formulae in estimating implied volatility
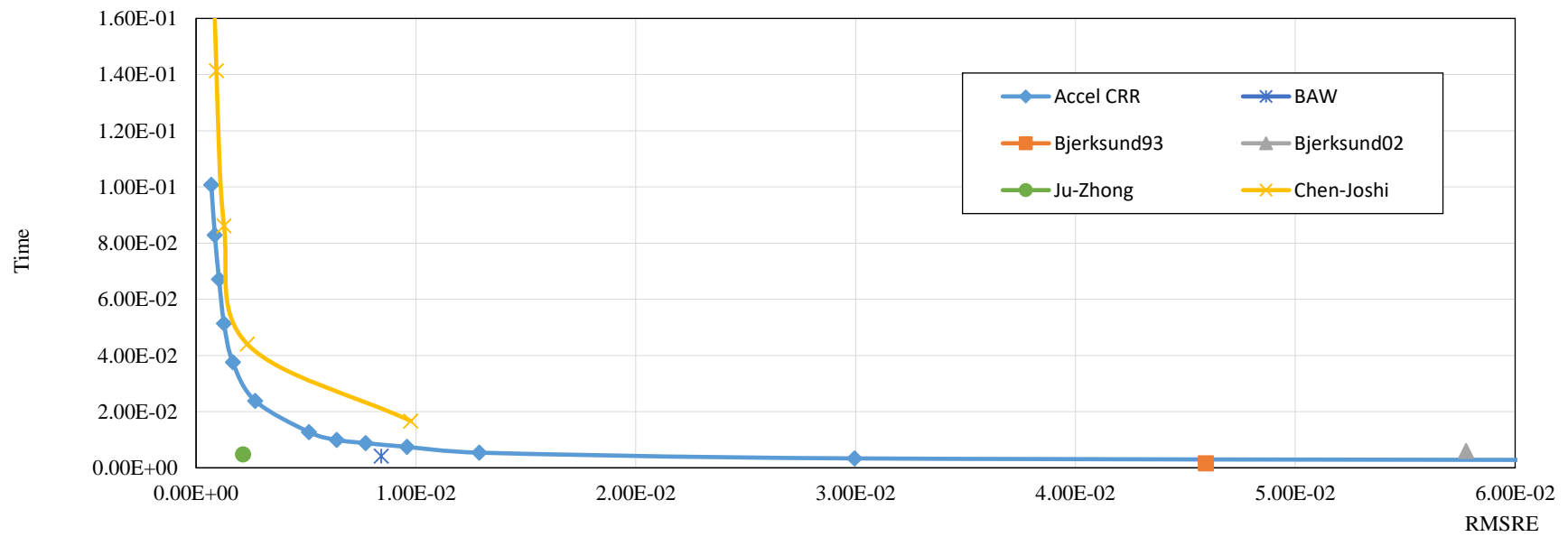
Figure 2.9: Comparing Accel CRR, Chen-Joshi, and analytical formulae in estimating implied volatility

Conclusion

In this paper, Cox, Ross and Rubinstein (1979) is revisited with the ambition to drive down the computational time for estimating American Options. The continuous optimal exercise boundary theory proposed by Kim and Byun (1994) and subsequently by Curran (1995) opened a significant vista for reaching greater efficiency for lattice type models despite restrictions being imposed on dividends. We augment their approach to include unrestricted dividends here. Our revamped continuous boundary theory addresses a non-trivial gap for practitioners, as early exercise can occur for an American put (call) option even when r < y (r > y). Furthermore, an intelligent lattice search algorithm is introduced to promptly locate the optimal exercise boundary for American options on assets with unrestricted dividends (yield).

Our accelerated CRR model combines intelligent lattice search, truncation and dynamic memory technologies. In each instance, we produce equivalent results to the original CRR model. A full Excel VBA implementation is made available in the Appendix to researchers. Computational runtime can be reduced from over 18 minutes down to less than 3 seconds to estimate a 15,000-step CRR tree. Significantly, American option pricing and delta estimation are accelerated, in terms of efficacy, dozens of times to hundreds of times as lattice steps increase. In addition, we compare Accel CRR with the leading benchmark tree Chen-Joshi, and four popular analytical formulae including BAW, Bjerksund93, Bjerksund02, and Ju-Zhong. These comparisons are made in terms of option pricing, delta estimation and implied volatility estimation. Our accelerated CRR model proves to be more efficient than Chen-Joshi and is capable of producing levels of speed consistent with analytical formulae. More importantly, our accelerated CRR model is advantageous to market professionals, in so much, that it flexibly provides varying levels of accuracy with different lattice step size. In contrast, each analytical formula can only afford a single combination of speed and accuracy.

Chapter 3: American Option Pricing: Optimal Lattice Models and Multidimensional Efficiency Tests

Abstract

This paper introduces a set of lattice techniques with a view to accelerating computation time and improving the accuracy of American Option valuation. Estimation speed can be enhanced through developing a parsimonious early exercise boundary search routine combined with reliance on dynamic memory and lattice truncation. Furthermore, Black-Scholes and Richardson extrapolation modifications can also be applied individually and/or together to improve the accuracy of lattices. In this paper, we investigate the improvement introduced by obtaining the best combination of varying features. By introducing these techniques to the Leisen-Reimer and Tian binomial model, we can achieve a level of accuracy and efficiency combined that surpass analytical analogues prominent in the literature. Significantly, the Leisen-Reimer and Tian structure can accommodate arbitrary improvements in accuracy by simply increasing the density of their own mesh. Analytical methods generally do not afford much scope for optimising speed and efficiency in a granular fashion. We also compare efficient lattice models with analytical formulae for pricing different groups of options according to the deepness of American quality and the moneyness of the options. The appropriate model is recommended for pricing particular type of the options. Lattices importantly afford an explicit trade-off locus between accuracy and speed that can be navigated according to predetermined precision tolerance levels and option types. This should have practical relevance to trading platforms that require real-time estimates of implied volatility.

Literature Review

A put (call) option provides the holder with the right to sell (buy) the underlying asset at a fixed price. If the holder can only exercise the option at the maturity, the option is referred to as being European. In contrast, an American option can be exercised early, at any time before the maturity. With the possibility of early exercise, American option pricing gives rise to an optimal exercise problem. A closed-form solution for European option pricing is available using Black and Scholes (1973). No equivalently simple closed-form solution exists for valuing American options robustly. McKean (1965) and Merton (1973) lay the foundation for the American option pricing. They show that the American option valuation problem can be viewed as a free boundary (also known as the optimal exercise boundary) problem of a partial differential equation. Since then, varying methods have been proposed for pricing American options written on a single underlying asset, which can be largely classified into two categories: analytical approximations or numerical methods.

Approximate analytic solutions are heavily relied upon to expedite estimation. Geske and Johnson (1984) valued American options using an infinite sequence of multivariate cumulative normal terms, which was improved in Bunch and Johnson (1992) by locating two exercise points. A key limitation however relates to the inefficiency associated with adding more and more terms to capture an exact representation of the free boundary problem typical in American options. Barone-Adesi and Whaley (1987, BAW) proposed a quadratic approximation. It is known to be fast and accurate for most input values but it becomes less accurate when pricing an American option with longer maturity horizons. American options values tend to diverge more markedly from their European equivalents as maturity increases. Ju and Zhong (1999, Ju-Zhong) extended the quadratic approach and improved pricing for longer maturities instruments relative to BAW. Ju-Zhong pointed out that their approach is more accurate than

the less efficient methods proposed, such as, the four-point extrapolation schemes of Geske and Johnson (1984) and Huang, et al. (1996). Li (2010) further extended the Ju-Zhong approach by suggesting an improved smoothing condition for American options. This had the effect of producing a more exact estimate of the optimal exercise price although did not guarantee a markedly better option pricing performance. The approximations of BAW, Ju-Zhong and Li models however do not in essence converge to the true value. In addition to quadratic approximation techniques, Bjerksund and Stensland (1993, Bjerksund93) obtained an approximation by simplifying the optimal exercise strategy with a unique flat early exercise boundary. Bjerksund and Stensland (2002, Bjerksund02) made further improvement by incorporating the second bound. These approaches represent accurate and efficient approximations to the American option value. Zhu (2006) proposed an exact solution by making use of a Taylor series expansion potentially leading to an infinite many terms. Zhu's approach points to the existence of a closed-form solution for American option pricing but is hard to implement in terms of computational costs (Kim et al. 2013; Chen and Joshi 2012; Medvedev and Scaillet 2010). In this paper, we compare our efficient lattice models with analytical approximations prominent in the literature and we show that our most refined models surpass the accuracy and speed of these commonly used analytical frameworks.

Numerical techniques are regularly used as benchmarks for analytical approximations on account of the high level of estimation accuracy. Brennan and Schwartz (1977) and Dempster and Hutton (1999) transformed the partial differential equation into linear equations and linear programming problems respectively using finite difference methods. Wu and Kwok (1997) and Nielsen et al. (2002) convert free-boundary problems into nonlinear problems using front-fixing methods. Lattice models are generally more relied upon relative to other numerical techniques used for the valuation of American options. In their seminal paper, Cox, Ross,

Rubinstein (1979, CRR) proposed a binomial model and noted that higher accuracy could be achieved by incorporating a larger number of time steps. Broadie and Detemple (1996) obtained the "true" value of American options by elaborating a 15,000-step benchmark binomial model for CRR. Using probabilistic methods, Amin and Khanna (1994) prove that the discrete-time models converge to the corresponding continuous-time models. This is important, as it provides a roadmap for establishing benchmarks. A number of variants of the original CRR trees have been proposed. Boyle (1986) proposed the trinomial tree in which each node generates three branches. Leisen and Reimer (1996, LR) and Tian (1993, Tian) reconstructed the CRR binomial model by modifying the effect of parameters on jumps between nodes and probabilities. The adaptive mesh method proposed in Figlewski and Gao (1999) embeds a secondary finer lattice mesh in key parts of the grid, which dramatically reduces the nonlinear error and has a wide range of applications. Staunton (2005) improved the LR model using Curtailed Ranges and Richardson extrapolation. The revamped LR model is the most efficient compared with the two best base approximations. Joshi (2009) and Chen and Joshi (2012) accelerate the Tian model using truncation, smoothing and Richardson extrapolation, which produce better performance than the improved LR model. Shang and Byrne (2019) propose an accelerated CRR model incorporating an intelligent lattice search algorithm, truncation and dynamic memory technologies. This model proved to be more efficient than the best performing model proposed by Chen and Joshi (2012), one of the leading benchmarks for lattice estimation, while did not in every instance surpass the closed form solution performance for American Option valuation.[10] In this paper, we improve the LR and Tian model both in speed and accuracy with intelligent lattice search, acceleration technologies

---

[10] In that paper, we used VBA to test and develop our models so that algorithms could be easily ported to any other desktop and peers could reproduce our research/results. In this paper, we rely upon C++ which tends on the whole to be more efficient that VBA. It also tends to influence the relative performance/efficiency of proposed models.

and accuracy modifications. We also show that the improved LR and Tian model can be more efficient than the Shang and Byrne (2019) tree and more efficient than leading closed formed solutions.

Identification of the optimal exercise boundary plays a central role in American option pricing. The optimal exercise boundary problem also exists for lattice models. Kim and Byun (1994) demonstrated the properties associated with the optimal exercise boundary of a CRR binomial model for an American put option without dividends payment. A definable locus of nodes that map out the stopping and continuation regions of the binomial tree can be revealed. Curran (1995) further extended the Kim-Byun boundary theory for American options with continuous dividend yields. Curran's boundary theory however is constrained in that the risk-free interest rate, r is necessarily superior (inferior) to the dividend yield, y for the put (call) option. Shang and Byrne (2019) overcome that limitation by initiating backward recursion from the penultimate column. This relatively small tweak implies that Curran's boundary theory can be relaxed to include a wider subset of parameter inputs for CRR. The effect of altering the seed column for backward recursion is fundamental to removing the constraint that American put valuation is subject to r ≥ y. We similarly relaxed the restriction r ≤ y imposed on call options which means that the limitations imposed by Curran (1995) can be circumvented. Basso et al. (2002, 2004) developed a binomial approximation to the optimal exercise boundary with the insight of Kim-Byun boundary theory. Areal and Rodrigues (2013) extended Curran's boundary theory to accelerate the binomial lattice for valuing American options with discrete dividends. Guo and Liu (2019) further proposed a binomial model for pricing options with known dollar dividends. In this paper, we extend the insights developed in Shang and Byrne (2019) and apply intelligent lattice search insights to both the LR and Tian trees with unrestricted continuous dividend yield.

In order to test and compare the efficiency of option pricing models, we need to amply generate random samples of option parameters. Broadie and Detemple (1996) set out criterion for producing random parameters, which has been adapted in turn by Chen and Joshi (2012) and Fabozzi, et al (2016). Gaudenzi and Pressacco (2003) and Pressacco, et al (2008) however found that the accuracy of American option pricing is linked to strong/weak underlying condition of parameters. Strong options, while few relative to all options, contribute proportionately the largest part of the estimation error. In this paper, we generate parameter samples following Broadie and Detemple (1996) and then filter results in terms of moneyness and the ratio of American to European (AER) analogues to signal strong/weak.[11]

The remaining paper is organized as follows. In Section 2, we introduce the LR tree and Tian tree and their optimal exercise boundary theory. In Section 3, we discuss two acceleration technologies and two accuracy modifications which can be used to improve LR and Tian models in speed and accuracy respectively. In Section 4, we demonstrate the pricing process of the improved LR and Tian models. In Section 5, we compare a series of improved LR and Tian models with the leading benchmark tree and more importantly to well-known analytical formulae. In each instance, we follow Strong-Weak-In-Out filtering consistent Pressacco, et al (2008). In Section 6, we conclude.

---

[11] Pressacco et al. (2008) chose an AER ratio of 1.4 to filter options in terms of being strong. They noted that large and erratic errors were common in a number of techniques that purported to value American options when AER exceeded 1.4.

Methodology

LR and Tian Trees and their Optimal Exercise Boundary Condition

Consider an American put option with spot price S, strike price X, time to maturity T, risk-free interest rate r, continuous dividend yield y, volatility σ, and a step size n. The time step is Δt = T / n. Leisen and Reimer (1996) propose an optimized binomial tree that centres around the exercise price using Preizer-Pratt inversion methods. They define the probability of moving upward and downward, p and q respectively, and the magnitude of upward and downward factors u and d, as:

$$p = h(d_2) \tag{3.1}$$

$$q = 1 - p \tag{3.2}$$

$$u = e^{(r-y)\Delta t} \frac{h(d_1)}{h(d_2)} \tag{3.3}$$

$$d = \frac{e^{(r-y)\Delta t} - pu}{1 - p} \tag{3.4}$$

where

$$d_1 = \frac{\ln\left(\frac{S}{X}\right) + \left(r - y + \frac{\sigma^2}{2}\right)T}{\sigma\sqrt{T}} \tag{3.5}$$

$$d_2 = d_1 - \sigma\sqrt{T} \tag{3.6}$$

$$h(x) = \frac{1}{2} + \frac{Sgn(d_1)}{2}\sqrt{1 - exp\left[-\left(\frac{x}{n + \frac{1}{3}}\right)^2\left(n + \frac{1}{6}\right)\right]} \tag{3.7}$$

Sgn() is the sign function and the number of time steps n should be odd.[12] Tian (1993) proposed that the binomial parameters are selected such that discrete-time distribution converges to the continuous-time distribution. Tian (1993) stipulated that p, q, u, and d are defined as:

$$p = \frac{M-d}{u-d} \tag{3.8}$$

$$q = 1 - p = \frac{u-M}{u-d} \tag{3.9}$$

$$u = \frac{MV}{2}[(V+1) + \sqrt{V^2 + 2V - 3}] \tag{3.10}$$

$$d = \frac{MV}{2}[(V+1) - \sqrt{V^2 + 2V - 3}] \tag{3.11}$$

where $M = \exp(r\Delta t)$ and $V = \exp(\sigma^2 \Delta t)$. Other than the varying definition of u, d and p, the stock price tree adheres to the same construction. Likewise, the backward induction method relevant for LR and Tian model is performed in the usual fashion, not unlike, the conventional CRR binomial tree. Each node in the binomial tree can be identified as having a (i, j) mapping, where i represents the ith column in which the node is located and j represents the number of upward movements of stock price consistent with each node. The maximum j is determined by the ith column count. The option value of node (i, j) is represented as V(i, j). Since the LR and Tian binomial tree uses a backward inductive pricing process, the pricing starts from the nodes at the maturity. At the final column (expiry), the respective put option values of the nodes are determined by:

$$V_{(n,j)} = \max(Su^j d^{n-j} - X, 0) \tag{3.12}$$

From the penultimate column back, the option values of the nodes are equal to the maximum of the exercise value and the holding value which is defined as:

---

[12] Here we calculate h(x) using the Preizer-Pratt inversion method 2. Alternatively, Preizer-Pratt inversion method 1 can also be used.

$$V_{(i,j)} = \max\{Su^j d^{i-j} - X, [pV_{(i+1,j+1)} + (1-p)V_{(i+1,j)}]/R\} \tag{3.13}$$

where R = exp(rΔt). The exercise value equates to the difference between the stock price of the node (i, j) and the fixed strike price X. The holding value is defined as the discounted weighted-average sum-product of two nodes from the preceding column, node $(i+1, j+1)$ and node $(i+1, j)$.

Kim and Byun (1994) developed some definitions/nomenclature for pricing non-dividend paying American put options using a CRR lattice. We extend these naming and mapping conventions to the LR and Tian binomial trees. A binomial tree can be divided into two regions: the stopping region S where the option is exercised and a continuation region C where the option is held. Formally, two regions are defined as:

$$S \equiv \{(i,j)|\, V_{(i,j)} = X - S_{(i,j)}\} \tag{3.14}$$

$$C \equiv \{(i,j)|\, V_{(i,j)} > X - S_{(i,j)}\} \tag{3.15}$$

The nodes in the stopping region are called stopping nodes and their values are equal to their corresponding exercise values. Similarly, the nodes pertaining to the continuation region are referred to as continuation nodes and equate to holding values. The **optimal exercise node** is the last stopping node which delineate the stopping nodes from the continuation nodes for each column. The nodes on and above the optimal exercise node are stopping nodes while the nodes below are continuation nodes. Optimal exercise nodes at each column constitute the optimal exercise boundary which separates the stopping and continuation regions. Following Kim and Byun (1994) and Curran (1995), Shang and Byrne (2019) propose an extended optimal exercise boundary theory that draws from efficiently delineating the stopping and holding regions. We found that the intelligent lattice search algorithm can be applied to LR and Tian trees. As before: *the optimal exercise boundary of a LR and Tian binomial tree is **continuous** from the*

49

penultimate column back when pricing American options. Consistent with Shang and Byrne (2019) no restrictions must be imposed on dividend yield relative to the risk-free rate.
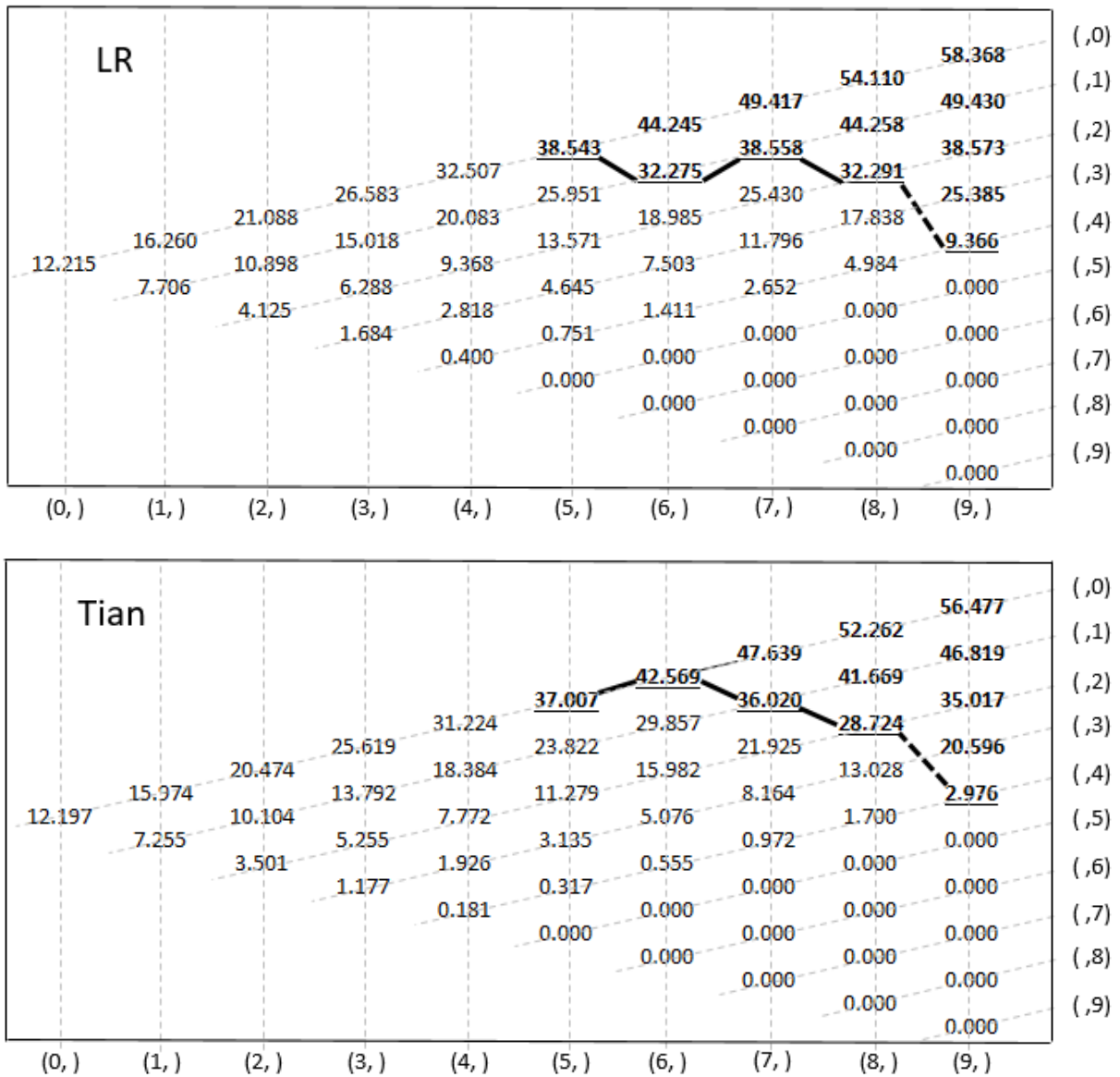
Figure 3.1: Optimal exercise boundary of a LR and Tian tree

The simplified numerical examples Figure 3.1 can be used to demonstrate some properties of the optimal exercise boundary for a LR and Tian tree, where it is noted that the boundary is continuous when implementing backward recursion from the penultimate column. Consider an American put option with the following parameter values: S = 100, X = 100, T = 1, r = 0.05, y

= 0.07 [13] and σ = 0.3. The 9-step LR and Tian binomial trees for pricing this option are shown in Figure 3.1.[14] Each node is represented by its corresponding option value. Importantly, the nodes in each column are organised in descending values for options which implies the stock price tree is flipped. The values marked in bold represent stopping nodes and the rest of values that are not marked in bold constitute continuation nodes. $(i,)$ and $(, j)$ signify the column and row number of the node respectively, where i and $j = 0, 1, \ldots 9$. The nodes falling on the same horizontal (tilted) light dashed line have the same column (row) number. The optimal exercise nodes are underlined and connected by the heavy lines; marking the optimal exercise boundary. As with Shang and Byrne (2019), it is clear that the boundary is continuous from the penultimate column stretching backward. In contrast, between the final and penultimate column, continuity cannot be guaranteed. Here, the solid continuous locus is interrupted by a jump between non-adjacent nodes marked by a heavy dashed line. The continuous boundary indicates that the j value (row number) associated with the optimal exercise node at each column and the previous column, are either equal or differ by a value of 1. In other words, if the optimal exercise node at i^{th} column is known as (i, j), the optimal exercise node at $(i-1)^{th}$ column is either $(i-1, j)$ or $(i-1, j-1)$. Therefore, the optimal exercise boundary can be confirmed by verifying the exercise condition of **merely a single node** in each column.[15] Once the boundary is determined, stopping and continuation regions/nodes can be further determined without any additional checks. Using the optimal exercise boundary theory, disposes of the requirement for blanket inspection - verifying the exercise condition of each node to bifurcate the tree. This dramatically reduces computational costs and effectively accelerates the pricing process.

---

[13] y > r which violates the assumption set out by Curran (1995). y = 0.07 is deliberately engineered to exceed r.
[14] A step-size of 9 adheres to the LR stipulation that n should be odd. In the algorithm we developed, if n happens to be even then the step sizes defaults to n+1.
[15] Conventional trees require a blanket search.

Acceleration Technologies and Accuracy Modifications

In this section, we introduce two acceleration technologies relating to truncation and dynamic memory and two accuracy modifications relating to Black-Scholes and Richardson Extrapolation. The latter two were proposed by Broadie and Detemple (1996). Truncation eliminates nodes that exert no effects on backward recursion but otherwise are nearly as computationally expensive as relevant nodes. Truncation purges nodes with a value of zero and stopping notes beyond the optimal exercise boundary. The dynamic approach releases memory because it is not necessary to store the entire tree at once. These two acceleration technologies, as well as the optimal exercise boundary theory, accelerate the computation but do not impact on accuracy. The accelerated trees can be made to replicate the original estimations but with a much higher speed. In contrast, accuracy modifications focus on improving the pricing accuracy while the speed is reduced marginally. Broadie and Detemple (1996) explained how the Black-Scholes formula can be introduced into the lattice at the $(n-1)^{th}$ time step (penultimate column). This was coupled with Richardson Extrapolation which improves efficiency, in principle, "by extrapolating to the limit". In principle, one might expect that numerical models adapted using Black-Scholes modification and/or Richardson Extrapolation have higher accuracy than their original configurations. Interestingly, we find that Intelligent Lattice Search can be combined with truncation and dynamic memory. This combination is developed to configure the **accelerated** model. Models that apply Black-Scholes modification or Richardson Extrapolation bear the suffix "BS" or "RE" or "BS&RE".[16]

When we price a European option using an original unmodified LR or Tian binomial tree, the option value of the node in one column is always generated from two nodes in the preceding column.[17] Therefore, through the backward induction pricing process, all nodes in the tree

---

[16] Broadie and Detemple (1996) refer to the combined approach as "BBRS".
[17] The exception being at maturity.

partially contribute to the generation of the apex node or present value of option. As noted previously for American lattices in Equation (3.13), each node from the penultimate column back is equal to the maximum of the exercise value or the holding value.[18] If the holding value is greater than the exercise value, the former will be assigned to the node. Otherwise, the option's value is equal to the exercise value which is the difference between the fixed strike price and the corresponding stock price of the node. In this event, the two adjacent nodes in the preceding column do not influence any value in the backward induction pricing process. This leads to a powerful conclusion. The value of an American option is only dependent on nodes within specific regions of the LR binomial tree. Curran (1995) proposes dividing the CRR lattice into subtrees, where some nodes exert no influence on the present value of the option but other nodes do. In this section, we apply the same truncation technology to LR and Tian binomial models and by extension prune the redundant nodes. In Figure 3.2, we take an example of the same 9-step LR binomial tree as Figure 3.1. Each number represents the option value of each node for this binomial tree. The values marked in bold are exercise values and the rest (not marked in bold) are holding values. The arrows connecting two option values represent the value-delivering process. It is not hard to observe that the apex node at the first column is a repository that contains information filtered from the entire LR binomial tree. There are some redundant nodes, enclosed in the two triangular regions, which are not involved in the value-delivering process. The nodes associated with the upper triangle do not influence the "next generation", which means their option values cannot be transferred to the next column and thereafter. For the lower triangle, the option values of nodes are zero. Truncation is typically implemented by recognising the redundant regions and excluding them algorithmically to reduce the computational load.

---

[18] The holding value is derived from two nodes in the previous column.
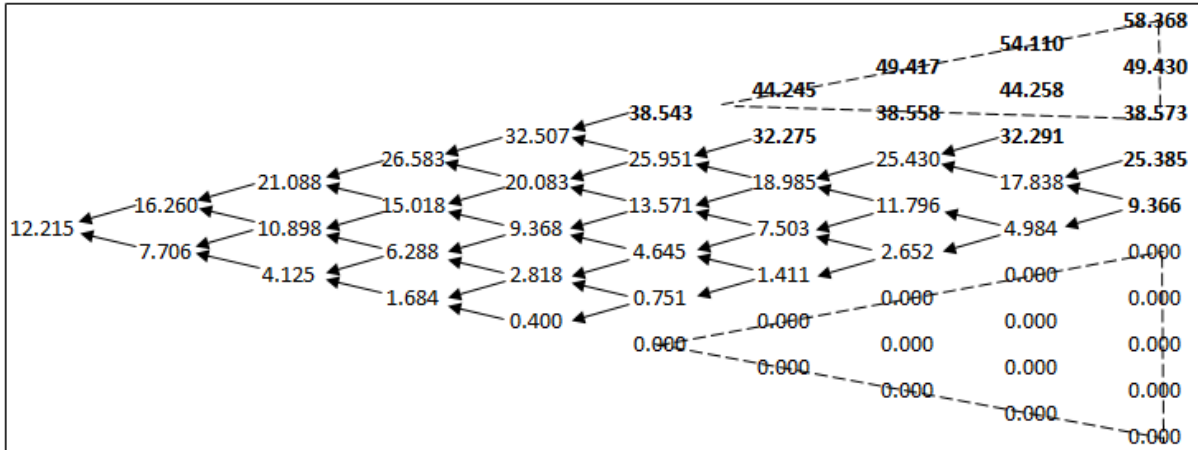
Figure 3.2: Truncation Technology

In a static memory environment, spaces are allocated in preparation for memorizing each option value computed in the valuation process. For a n-step conventional binomial tree, the option values of (n+1)(n+2)/2 nodes have to be memorized so that an appropriate quantity of static memory spaces are assigned[19]. Broadie and Detemple (1996) and Haug (2007) propose a dynamic memory method. For either a n-step dynamic LR or Tian binomial tree, only a maximum of (n+1) memory spaces need to be allocated and option values of (n+1) nodes need to be memorized sequentially. Using dynamic memory, the backward inductive pricing is also a substitution process where option values previously stored in the memory spaces are replaced by newly generated option values. In Figure 3.3, we construct a 5-step dynamic LR binomial tree for pricing the same American put option as Figure 3.1. Here we use (j) instead of (i, j) to locate the node at each column, where $j = 0, 1, \ldots, 5$. The nodes falling on the same slant dashed line (upper figure) share the same serial number j. In the dynamic substitution process (the lower figure), we first allocate six memory spaces, V(0) - V(5), represented by rectangles, in which the computed option values of nodes at the last column are stored. When we move backward to the penultimate column, the option values at this column are computed and replace

---

[19] For example, a 2-step binomial tree has 6 nodes with option values V(0,0), V(1,0), V(1,1), V(2,0), V(2,1), V(2,2) respectively so that 6 static memory spaces are allocated.

option values previously stored in V(0) – V(4). Similarly, option values at antepenultimate column are computed and replace values previously memorized in V(0) – V(3). This computation and substitution process lasts until the option value at the first column is computed and memorized in V(0). Dynamic memory conserves precious processing power by virtue that this style of implementation simply employs $O(n)$ storage. Conventional static lattice implementations, with n steps, possess $O(n^2)$ nodes and consequently computation time increases in tandem with $O(n^2)$.

```
                                              47.249   (0)
                                      40.051           (1)
                              31.870          31.923   (2)
                      24.071          22.714           (2)
              17.428          15.228          12.145   (2)
      12.195          9.826           6.564            (3)
              6.173           3.548           0.000     (3)
                      1.918           0.000            (4)
                              0.000           0.000    (4)
                                      0.000            (5)
                                              0.000     (5)
```

V(0)	47.249→	40.051→47.249	31.870→40.051	24.071→31.870	17.428→24.071	12.195→17.428
V(1)	31.923→	22.714→31.923	15.228→22.714	9.826→15.228	6.173→9.826	6.173
V(2)	12.145→	6.564→12.145	3.548→6.564	1.918→3.548	1.918	0.000
V(3)	0.000→	0.000→0.000	0.000→0.000	0.000	0.000	0.000
V(4)	0.000→	0.000→0.000	0.000	0.000	0.000	0.000
V(5)	0.000→	0.000	0.000	0.000	0.000	0.000

Figure 3.3: Dynamic Memory

Broadie and Detemple (1996) introduce two modifications to the binomial method: Black-Scholes continuity values at the penultimate nodes and Richardson extrapolation. In principle, the Black-Scholes modification improves the binomial tree by replacing the conventional holding values of each node at the penultimate column with the corresponding set of Black-Scholes values. The option value of the nodes at the penultimate column is given by:

$$V_{(n-1,j)} = \max\{X - Su^j d^{(n-1)-j}, BS(Su^j d^{(n-1)-j}, X, \Delta t, r, y, \sigma)\} \qquad (16)$$

Where BS(.) represents the Black-Scholes value. Since the Black-Scholes model is originally designed for pricing European options at the limit, more accurate holding values can be obtained. It should be noted, of course, that when applying Black-Scholes valuation, the zero-value nodes will be replaced by non-zero Black-Scholes values. This impedes the truncation of the zero-value zone and imposes an additional computational workload.

Broadie and Detemple (1996) shows that the accuracy of a binomial model can be significantly improved by using two-point Richardson Extrapolation. Allow f(n) to denote the estimation of an option price using a n-steps binomial model[20]. The extrapolated option price, $f_{RE}(n)$, is easily obtained by:

$$f_{RE}(n) = 2f(n) - f\left(\frac{n}{2}\right) \qquad (127)$$

Unlike other technologies, Richardson Extrapolation will be applied only after the option prices have been obtained from two binomial trees with n and a half-n number of steps. Richardson Extrapolation can be easily adapted to a lattice model but does impose added computational costs. Joshi (2009) and Chen and Joshi (2012) noted that the Black-Sholes modification interacts well with Richardson Extrapolation. The combination of these two modifications significantly improves the accuracy but only marginally decreases the speed within conventional lattice structures.

---

[20] Here we assume n is even.
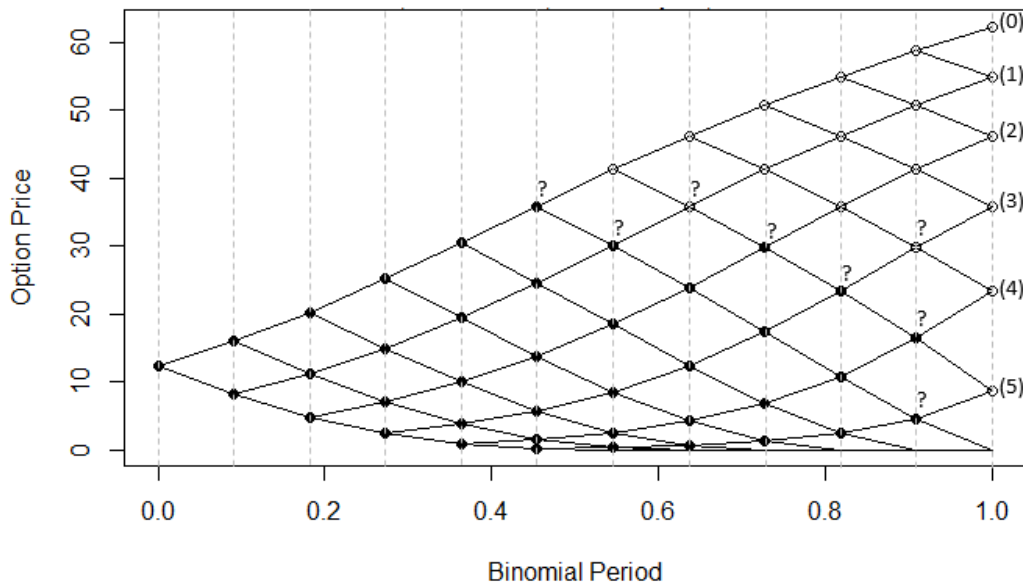
Figure 3.4: Pricing an American option using an accelerated LR binomial model

In this section, we tease out the pricing process of both the accelerated LR binomial model and accelerated Tian model. The original LR and Tian model can be accelerated by using intelligent lattice search, dynamic memory, and truncation technologies. The accelerated pricing process will be presented first and then the work-flow logic outlined in Figure 3.4 will be described. Consider the same American put option as identified in Figure 3.1 where S = 100, X = 100, T = 1, r = 0.05, y = 0.07 and σ = 0.3. An 11-step[21] LR binomial tree for pricing this option is shown in Figure 3.4. The X axis represents the binomial period. The Y axis represents the option price so that the zero-value region of the tree naturally disappears. The serial number from (1) to (5) correspond to each upward curve of the tree. The nodes falling on the same upward curve have the same serial number. The vertical dashed lines represent each column. Solid and hollow nodes represent the continuation and stopping nodes respectively. Only the

---

[21] An odd number of steps is required for option pricing in a LR binomial model.

nodes with a question mark above, also called uncertain nodes, need to be inspected to verify the exercise condition.

The backward inductive option pricing starts from the maturity (the final column). An interesting feature of a LR binomial tree relates to the first non-zero value node or node (h) at each step, where h = (n - 1) / 2. In other words, the nodes in the LR tree with a serial number greater than h are all zero-value redundant nodes, which can be easily recognised and truncated. The Tian tree, however, does not possess this feature so that the exercise condition of the nodes at the maturity has to be checked one by one until the first non-zero-value node is determined. In Figure 3.4, the first non-zero value node at the last column of the tree is node (5) as h = 5, which is also the optimal exercise node (first stopping node) at that column. The corresponding exercise values, from 62.22 to 8.54, should be computed and assigned to nodes (0) - (5) in the final column. The nodes below node (5) are recognized and truncated as zero-value nodes.

Next, we move backward to the penultimate column. The optimal exercise boundary of the American put is not continuous between the last and penultimate column given that r < y. We therefore have to verify the exercise condition of each node in the penultimate column. We start from the first non-zero-value node (h) in this column, comparing the holding value and exercise value of each node, until the optimal exercise node is found. Once the optimal exercise node is determined, we can cease iteratively verifying the exercise condition and start value assignment. The exercise values of the optimal exercise node, and the holding values of the nodes between the optimal exercise node and node (h), are then computed.[22] The remaining nodes in this column can then be truncated as redundant nodes. In Figure 3.4, at the penultimate column, we examine the exercise condition of each node, starting from node (5) until node (3) which is confirmed as the optimal exercise node. Then the exercise value of 29.85 is computed

---

[22] The nodes between the optimal exercise nodes and node (h) include node (h) but exclude optimal exercise node.

and assigned to node (3), and the holding value, 16.35 and 4.52, are calculated and assigned to nodes (4) and (5).

From the penultimate column back, we are able to apply intelligent search and avoid a blanket inspection of the exercise-condition. For the remaining columns, the exercise condition of only one uncertain node needs to be examined – the node with the same serial number as the optimal exercise node of the previous column.[23] There are two possible outcomes for the uncertain node at each column:

(1) If its exercise value is greater than the holding value, this node is the optimal exercise node at this column. The corresponding exercise values are calculated and assigned to the optimal exercise node and the node immediately above it.

(2) If its exercise value is smaller than holding value, this implies that the node is a continuation node and the node immediately above it should be the optimal exercise node at the same column. The corresponding exercise values are calculated and assigned to the optimal exercise node.

The holding values are computed and assigned to the nodes between the optimal exercise node and node (h). The remaining nodes at that column: the zero-value nodes and the redundant stopping nodes are truncated. This efficient pricing process can be applied to the antepenultimate column and each column thereafter. In Figure 3.4, when we move to the antepenultimate column, only the exercise condition of node (3) needs to be verified since the optimal exercise node at the penultimate column has the serial number of 3. Once node (3) is determined as a continuation node (holding value > exercise value), the optimal exercise node at the antepenultimate column should be node (2). Then the exercise value 35.78 is assigned to

---

[23] The optimal exercise boundary theory implies that if the optimal exercise node at i^{th} column is node (j), the optimal exercise node at $(i-1)^{th}$ column has only two possibilities: either node (j) or node (j-1).

node (2) and the holding values 23.42, 10.77 and 2.40 are assigned to node (3) – (5) respectively. The rest of nodes at this column can be truncated.

This efficient pricing process continues to be applied to each of the remaining column until the column at which all nodes are stopping nodes (or continuation nodes), which indicates that all nodes in the rest of the columns are stopping nodes (or continuation nodes). If the remaining nodes are all stopping nodes, the present value of the option can be directly equal to the difference between strike price and spot price. Otherwise, if they are all continuation nodes, the holding value of each node in the rest of columns are computed to obtain the present value of the option. In Figure 3.4, the efficient backward inductive pricing process ends at the column fully consisting of solid nodes (continuation nodes). Hence, the corresponding holding value are computed and assigned to the nodes at each of the other columns. Finally, the present value of the option is obtained as 12.22.

Combining intelligent lattice search, dynamic memory and truncation conspires to dramatically accelerate the computation speed for the LR binomial model. Instead of examining the exercise condition of each node of the tree, we only need to inspect at most one uncertain node in each column from the antepenultimate column back. Furthermore, redundant zero-value nodes and redundant stopping nodes can also be easily recognised and truncated. Moreover, this algorithm saves significant memory space by virtue that only (h+1) instead of (n+1)(n+2)/2 memory space is required.

Both the accelerated LRBS and TianBS models apply Black Scholes smoothing in the accelerated LR and Tian models. This entails making a number of changes. All nodes at the last column are discarded because they will not infer their values backward. The pricing process starts from the penultimate column, where we verify the exercise condition of each node by comparing the Black-Scholes value with the corresponding exercise value. This evaluation

ends once the optimal exercise node is determined where the exercise value is greater than the Black-Scholes value. Then the corresponding exercise values are assigned to the optimal exercise node and the corresponding Black-Scholes values are assigned to the nodes below the optimal exercise node. From the antepenultimate column back, the pricing process of the accelerated LRBS and TianBS trees are the same as the accelerated LR and Tian trees.[24]

Numerical Results

In this section, we assert that the standard LR and Tian model can be effectively accelerated by adapting intelligent lattice search, truncation and dynamic memory without disturbing the accuracy of the parent model. We shall compare variants of the accelerated LR and Tian models with commonly used analytical formulae for pricing American options. We make a similar comparison to leading benchmark lattice, based on randomly sampled parameters. Finally, we divide sample options into four groups according to the AER ratio and the moneyness of the options. We test the relative performance of the efficient LR and Tian models based on intelligent lattice search for varying option specification.

As with previous authors, we need an error measure and time consumption measure that go beyond purely convergence and step size limited to one set of parameters with an incremental step size. In addition, we need a sufficient level of randomization, so our results are not the product of any specific pattern in the choice of parameters or reflecting other biases. To meet this challenge, we follow Broadie and Detemple (1996) and generate 2,500 American option parameter combinations. The stock price S is distributed uniformly between 70 and 130. Time to maturity T is also uniformly randomised with boundaries between 0.1 and 1 years with a

---

[24] Notice that for an accelerated LRBS and TianBS tree, the zero-value zone disappears and is replaced by non-zero Black-Scholes values.

probability weight of 0.75 and additionally 1 to 5 years with a probability weighting of 0.25. The risk-free interest rate r is bounded between 0 and 0.1 with probability of 0.8 and equal to 0 with a probability of 0.2. The dividend yield, y, is uniformly generated spanning 0 to 0.10. Volatility σ ranges between 0.1 and 0.6 and adheres to a uniform distribution. The strike price X is fixed at 100. The options are configured with a 0.5 probability of being a call or a put. The error measure, root mean squared relative error (RMSRE), is defined consistently with Broadie and Detemple (1996) as:

$$\text{RMSRE} = \sqrt{\frac{1}{m}\sum_{i=1}^{m} e_i^2} \qquad (3.18)$$

where m = the number of American options and $e_i = \frac{\hat{C}_i - C_i}{C_i}$ with $\hat{C}_i$ representing the estimation of the option value and C_i representing the true value of the option generated by a 15,000-step CRR binomial model. We discard 170 options with true values smaller than 0.5. Finally, 2,330 American options are finally contained in the valuation. The time consumption measure, average CPU time (Time), is defined as:

$$\text{CPU time} = \frac{Sum\ of\ processsor\ time}{2330} \qquad (3.19)$$

All the numerical results are generated in C++ Visual Studio 2015 on a DELL Latitude E5470 laptop with Intel Core i3-6100U CPU (2.30 GHz) and 8GB of RAM.

Initially we demonstrate that intelligent lattice search, truncation and dynamic memory modifications unambiguously accelerate the LR and Tian models without altering original estimates. We price 2,330 American options generated above using both an accelerated LR and a standard LR model and using both an accelerated Tian and a standard Tian model. The number of steps for all pairings range from 100 to 1500 in intervals of 100.[25] The corresponding

---

[25] Notice that the actual number of steps that a series of LR models used is the step number shown on the table plus one since LR models require odd steps.

RMSRE and CPU Time are reported in Table 3.1, where Accel LR and LR / Accel Tian and Tian always have identical RMSRE but the former takes much less computational time than the latter. In Figure 3.5, we graph the results associated with LR models in Table 3.1 to make it more intuitive. Two solid curves represent the Accel LR and LR respectively. Each point represents a combination of RMSRE and Time generated by the corresponding tree. It is noteworthy that the Accel LR and LR options sharing the same number of steps share also the same reported RMSRE. This provides some evidence that the Accel LR and LR produce exactly the same estimations.[26] The length of the vertical dashed lines separating the CPU Time for the Accel LR arc and the LR arc in Figure 3.5, points to inherent efficiencies in respect to the Accelerated model. This increases exponentially as the number of steps are increased. This improvement in efficiency can be explored further by introducing modifications for Black Scholes smoothing and Richardson Extrapolation. These findings would appear also apply to the Accel Tian and Tian model.

	Accel LR		LR		Accel Tian		Tian	
Steps	RMSRE	Time	RMSRE	Time	RMSRE	Time	RMSRE	Time
100	0.0562%	3.72E-03	0.0562%	1.24E-02	0.389%	3.71E-03	0.389%	1.79E-02
200	0.0285%	7.49E-03	0.0285%	5.76E-02	0.187%	1.09E-02	0.187%	5.71E-02
300	0.0193%	1.59E-02	0.0193%	1.19E-01	0.128%	2.09E-02	0.128%	1.14E-01
400	0.0144%	2.67E-02	0.0144%	1.83E-01	0.095%	3.49E-02	0.095%	1.94E-01
500	0.0115%	4.22E-02	0.0115%	2.65E-01	0.074%	5.11E-02	0.074%	2.97E-01
600	0.0096%	5.69E-02	0.0096%	3.67E-01	0.064%	7.44E-02	0.064%	4.05E-01
700	0.0082%	7.88E-02	0.0082%	5.13E-01	0.056%	9.67E-02	0.056%	5.36E-01
800	0.0072%	9.86E-02	0.0072%	6.47E-01	0.045%	1.25E-01	0.045%	7.26E-01
900	0.0065%	1.25E-01	0.0065%	8.90E-01	0.045%	1.57E-01	0.045%	9.30E-01
1000	0.0059%	1.71E-01	0.0059%	1.01E+00	0.041%	1.91E-01	0.041%	1.11E+00
1100	0.0055%	1.87E-01	0.0055%	1.23E+00	0.038%	2.33E-01	0.038%	1.34E+00
1200	0.0051%	2.11E-01	0.0051%	1.45E+00	0.030%	2.76E-01	0.030%	1.71E+00
1300	0.0047%	2.47E-01	0.0047%	1.78E+00	0.028%	3.55E-01	0.028%	1.89E+00
1400	0.0045%	2.85E-01	0.0045%	2.00E+00	0.027%	3.96E-01	0.027%	2.21E+00
1500	0.0043%	3.43E-01	0.0043%	2.35E+00	0.025%	4.42E-01	0.025%	2.53E+00

---

[26] No differences between either sets of options were found for the Accel LR and Standard LR models.

Table 3.1: Comparing Accel LR (Accel Tian) and Standard LR (Tian)
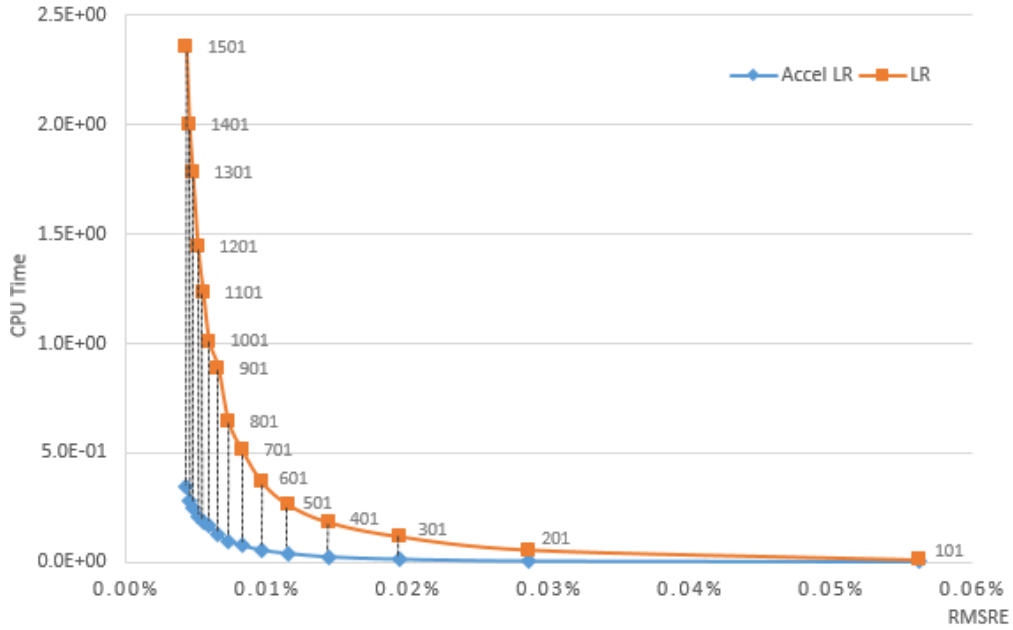
Figure 3.5: Comparing the Accelerated LR and Standard LR models

In what follows, we compare accelerated LR and Tian models adapted different accuracy modifications with commonly-used analytical formulae and the leading benchmark tree. These eight LR and Tian models include:

i.   Accel LR: The accelerated LR model, which is the original LR model accelerated by using intelligent lattice search, dynamic memory, and truncation technologies

ii. Accel LRBS: The accelerated LR model with Black-Scholes smoothing

iii. Accel LRRE: The accelerated LR model with Richardson Extrapolation

iv. Accel LRBS&RE: The accelerated LR model with Black Scholes smoothing and Richardson Extrapolation

v.   Accel Tian: The accelerated Tian model, which is the original Tian model accelerated by using intelligent lattice search, dynamic memory, and truncation technologies

vi. Accel TianBS: The accelerated Tian model with Black-Scholes smoothing

vii. Accel TianRE: The accelerated Tian model with Richardson Extrapolation

64

viii.   Accel TianBS&RE: The accelerated Tian model with Black Scholes smoothing and Richardson Extrapolation

Importantly, as mentioned in the Introduction section, BAW, Bjerksund 93, Bjerksund 02, Ju-Zhong are four commonly used analytical formulae and the accelerated CRR model (Accel CRR) serves as a benchmark lattice model. We, initially, use relatively small step numbers ranging from 10 to 100 with increments of 10. In Table 3.2, we show the RMSRE and CPU Time when pricing 2,330 American options using eight LR and Tian models described above, the leading benchmark tree and four analytical formulae. It is obvious that each analytical formula can only provide one combination of accuracy and speed while lattice models offers more flexibly generate arbitrary levels of precision in tandem with a given tolerance for execution time. Figure 3.6 allows us to visualize the results reported in Table 3.2, where X (Y) axis represents RMSRE (CPU Time) respectively. In Figure 3.6, lattice models are represented as curves and visualized in three sub-figures according to their performance relative to the benchmark analytical formulae. Four analytical formulae are represented by a single point separately in each of three sub-figures with the method name ascribed for clarity. We can evaluate each model in terms of Pareto Improvement i.e. the arc (or point) closer to the origin that boasts the highest level of performance. It is discernible in the top-left sub-figure of Figure 3.6 that the large segments of the two arcs embedded in the Accel LR and Accel LRRE loci are unmistakably closer to the origin relative to the four analytical formulae points, which indicates that Accel LR and Accel LRRE surpass analytical formulae in both speed and accuracy. In the top-right sub-figure, the Accel LRBS&RE and Accel TianBS&RE models perform better than three of the analytical formulae including Ju-Zhong, Bjerksund 02 and BAW although fare worse than Bjerksund 93. As evident in the bottom sub-figure, the rest of the lattice models, including Accel CRR, Accel LRBS, Accel Tian and Accel TianBS, have

comparable performance relative to the analytical formulae. The four analytical formulae points largely coincide and are in the vicinity of these lattice models curves. Accel TianRE is not reported in Figure 3.6 as performance proved to be inferior to the analytical formulae outlined in Table 3.2.

Higher accuracy can be achieved arbitrarily by increasing step size. From this vantage point, we proceed to compare eight accelerated LR and Tian models and the leading benchmark lattice model. We apply pricing to 2,330 American options using a relatively larger number of steps ranging from 100 to 1,000. Table 3.3 reports the corresponding RMSRE and CPU Time of these nine lattice models, which are graphed in Figure 3.7. As before, the X (Y) axis represents RMSRE (CPU Time). It is evident that Accel TianBS&RE, Accel LRRE and Accel LRBS&RE boast the highest efficiency as their respective loci are closer to the origin. More specifically, the convergences of Accel TianBS&RE is smoother and less noticeably oscillating relative to Accel LRRE and Accel LRBS&RE. Accel LR also has the advantage of the smooth convergence but the efficiency is slightly lower than the best three. The secondary tier includes Accel LRBS, Accel CRR, Accel TianBS and Accel Tian, whose efficiency is one level lower than the previous four models.

Method	Accel LR		Accel LRBS		Accel LRRE		Accel LRBS&RE		BAW	
Steps	RMSRE	Time	RMSRE	Time	RMSRE	Time	RMSRE	Time	RMSRE	Time
11	0.5966%	2.84E-03	2.1327%	2.93E-03	0.6642%	2.89E-03	0.6817%	3.02E-03	1.24%	3.17E-03
21	0.2783%	2.92E-03	1.1373%	3.03E-03	0.2282%	2.94E-03	0.2918%	3.09E-03	Bjerksund93	
31	0.1952%	2.99E-03	0.7764%	3.07E-03	0.2481%	3.03E-03	0.2564%	3.12E-03	RMSRE	Time
41	0.1434%	3.00E-03	0.5860%	3.17E-03	0.0774%	3.05E-03	0.0911%	3.23E-03	0.77%	2.96E-03
51	0.1207%	3.12E-03	0.4726%	3.26E-03	0.0698%	3.14E-03	0.0768%	3.32E-03	Bjerksund02	
61	0.0946%	3.13E-03	0.3929%	3.43E-03	0.0440%	3.19E-03	0.0500%	3.38E-03	RMSRE	Time
71	0.0771%	3.25E-03	0.3359%	3.45E-03	0.0343%	3.27E-03	0.0406%	3.65E-03	0.59%	3.06E-03
81	0.0741%	3.38E-03	0.2958%	3.51E-03	0.0339%	3.42E-03	0.0364%	4.18E-03	Ju-Zhong	
91	0.0665%	3.55E-03	0.2631%	4.05E-03	0.0337%	3.70E-03	0.0363%	4.65E-03	RMSRE	Time
101	0.0562%	3.72E-03	0.2358%	4.48E-03	0.0312%	3.99E-03	0.0327%	5.47E-03	0.29%	3.96E-03

Method	Accel Tian		Accel TianRE		Accel TianBS		Accel TianBS&RE		Accel CRR	
Steps	RMSRE	Time	RMSRE	Time	RMSRE	Time	RMSRE	Time	RMSRE	Time
10	3.6776%	2.80E-03	5.3312%	2.78E-03	3.0778%	3.00E-03	1.1417%	3.02E-03	3.5614%	2.90E-03
20	1.9925%	3.29E-03	3.1237%	2.66E-03	1.6298%	2.87E-03	0.2837%	3.06E-03	1.8507%	2.95E-03
30	1.2083%	2.86E-03	1.8952%	3.26E-03	1.1000%	3.29E-03	0.1350%	3.24E-03	1.0773%	2.96E-03
40	0.9452%	3.43E-03	1.3783%	3.43E-03	0.8327%	2.91E-03	0.0823%	3.29E-03	0.9981%	3.03E-03
50	0.8105%	2.93E-03	1.1529%	3.52E-03	0.6704%	3.26E-03	0.0572%	3.30E-03	0.7646%	3.08E-03
60	0.6914%	3.10E-03	0.9408%	3.63E-03	0.5602%	3.29E-03	0.0503%	3.46E-03	0.6161%	3.09E-03
70	0.5470%	3.57E-03	0.8421%	3.75E-03	0.4807%	3.53E-03	0.0376%	3.57E-03	0.5579%	3.24E-03
80	0.4817%	3.40E-03	0.6799%	4.01E-03	0.4208%	3.46E-03	0.0264%	3.68E-03	0.4714%	3.38E-03
90	0.4189%	3.46E-03	0.6222%	4.34E-03	0.3752%	3.95E-03	0.0234%	4.03E-03	0.4108%	3.43E-03
100	0.3891%	3.71E-03	0.5891%	4.88E-03	0.3376%	4.24E-03	0.0210%	4.73E-03	0.3604%	3.79E-03

Table 3.2: Comparing the Accelerated LR and Tian models with analytical formulae and the leading benchmark tree using the smaller step-size
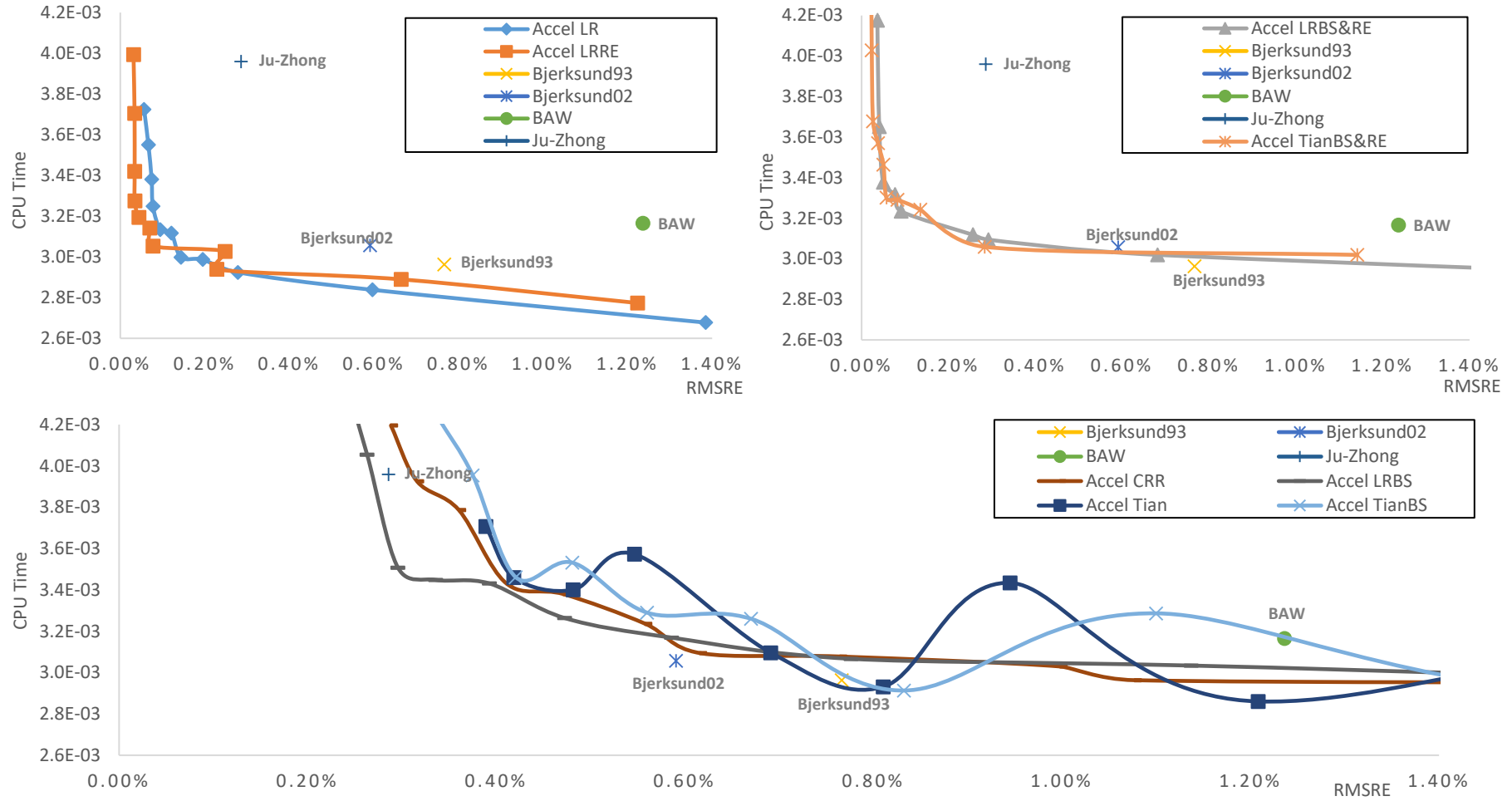
Figure 3.6: Comparing Accelerated the LR and Tian models with analytical formulae and the leading benchmark tree using the smaller steps-size

Method	Accel LR		Accel LRRE		Accel LRBS		Accel LRBS&RE		Accel CRR	
Steps	RMSRE	Time	RMSRE	Time	RMSRE	Time	RMSRE	Time	RMSRE	Time
100	0.0562%	3.72E-03	0.0312%	3.99E-03	0.2358%	4.48E-03	0.0327%	5.47E-03	0.3604%	3.79E-03
200	0.0285%	7.49E-03	0.0106%	1.02E-02	0.1173%	1.32E-02	0.0113%	1.52E-02	0.1791%	7.80E-03
300	0.0193%	1.59E-02	0.0069%	2.12E-02	0.0780%	2.76E-02	0.0071%	3.19E-02	0.1079%	1.57E-02
400	0.0144%	2.67E-02	0.0043%	3.37E-02	0.0582%	4.37E-02	0.0045%	5.47E-02	0.0805%	2.61E-02
500	0.0115%	4.22E-02	0.0061%	5.16E-02	0.0463%	6.78E-02	0.0061%	8.41E-02	0.0684%	3.98E-02
600	0.0096%	5.69E-02	0.0039%	7.38E-02	0.0385%	9.92E-02	0.0040%	1.22E-01	0.0570%	5.84E-02
700	0.0082%	7.88E-02	0.0030%	9.85E-02	0.0328%	1.35E-01	0.0031%	1.63E-01	0.0463%	7.86E-02
800	0.0072%	9.86E-02	0.0033%	1.26E-01	0.0287%	1.69E-01	0.0034%	2.08E-01	0.0410%	9.94E-02
900	0.0065%	1.25E-01	0.0035%	1.56E-01	0.0255%	2.14E-01	0.0035%	2.68E-01	0.0367%	1.22E-01
1000	0.0059%	1.71E-01	0.0036%	1.90E-01	0.0229%	2.66E-01	0.0036%	3.25E-01	0.0385%	1.53E-01

Method	Accel Tian		Accel TianRE		Accel TianBS		Accel TianBS&RE	
Steps	RMSRE	Time	RMSRE	Time	RMSRE	Time	RMSRE	Time
100	0.3891%	3.71E-03	0.5891%	4.56E-03	0.3376%	4.24E-03	0.0210%	4.73E-03
200	0.1872%	1.09E-02	0.2852%	1.36E-02	0.1688%	1.22E-02	0.0082%	1.50E-02
300	0.1276%	2.09E-02	0.1894%	2.59E-02	0.1121%	2.62E-02	0.0064%	3.33E-02
400	0.0954%	3.49E-02	0.1402%	4.45E-02	0.0838%	4.66E-02	0.0041%	5.65E-02
500	0.0741%	5.11E-02	0.1155%	6.57E-02	0.0668%	6.72E-02	0.0037%	8.72E-02
600	0.0636%	7.44E-02	0.0920%	9.18E-02	0.0553%	9.52E-02	0.0035%	1.22E-01
700	0.0557%	9.67E-02	0.0825%	1.31E-01	0.0474%	1.29E-01	0.0035%	1.60E-01
800	0.0455%	1.25E-01	0.0693%	1.68E-01	0.0413%	1.65E-01	0.0029%	2.10E-01
900	0.0451%	1.57E-01	0.0640%	2.20E-01	0.0363%	2.07E-01	0.0028%	2.70E-01
1000	0.0412%	1.91E-01	0.0579%	2.57E-01	0.0327%	2.61E-01	0.0026%	3.22E-01

Table 3.3: Comparing Accelerated LR and Tian models and the leading benchmark tree using the larger steps-size
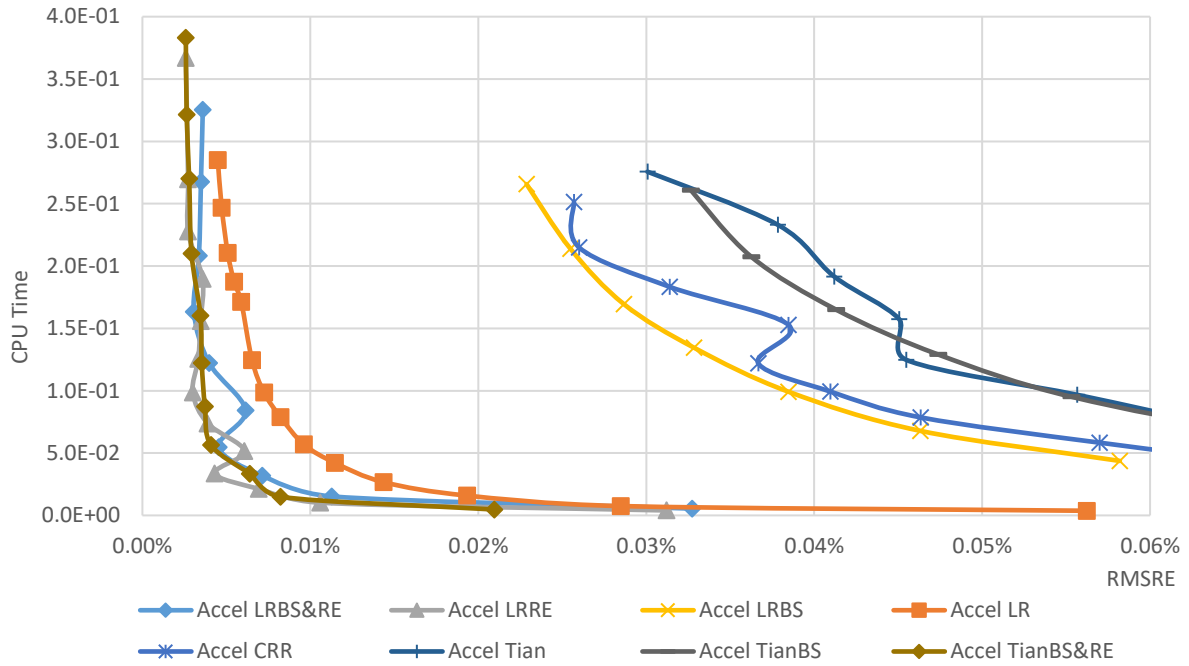
Figure 3.7: Comparing Accelerated LR and Tian models and the leading benchmark tree using the larger steps-size

Consistent with Pressacoo, et al. (2008), we further divide American options samples into four different groups according to their AER ratio and the moneyness classifications. We initially test the performance for each model for varying classification. First, according to the moneyness of the options, 2,330 previously used American options are divided into 1,190 in-the-money and 1,140 out-of-the-money options. Next, we separate the options according to the AER: quantitatively measured by the Pressacoo, et al. (2008) ratio of 1.4 where American value[27] exceeds the European twin. The 2,330 option sample filters into 2,281 Weak options where AER < 1.40 and 49 Strong options where the AER ≥ 1.40. In aggregate that leaves 2,330 options being separated into 1,157 Weak-In options, 1,124 Weak-Out options, 33 Strong-In options and 16 Strong-Out options. The Weak-In and Weak-Out options are large enough to make reliable tests on while the Strong-In and Strong-Out options are not plausibly sufficient.

---

[27] The American value of options used here is the quick reliable estimation.

We therefore continue to randomly generate American options, following the same criterion of parameter selection as we generated 2,330 before. We retain only Strong options up to 1,000 from the new generated options and add them to the 49 Strong options already obtained. Finally, we obtain 569 Strong-In options and 302 Strong-Out options.

Based on the results of the previous tests, Accel LR, Accel LRRE, Accel LRBS&RE and Accel TianBS&RE have outstanding performance in pricing American options with both lower and higher level of accuracy (small and large step-size). Here we apply these four lattice models as well as four analytical formulae to value 589 Strong-In, 302 Strong-Out, 1,157 Weak-In, and 1,124 Weak-Out options previously generated. For lattice models, in addition, the number of steps is ranged from 10 to 1,000.[28] The corresponding RMSRE and CPU Time are reported in Table 3.4 and visualized in Figure 3.8 and Figure 3.9. First of all, we compare the difficulty of pricing different types of options. As it is shown in Table 3.4, the RMSRE generated by lattice models for pricing Weak options is much less than when typically pricing Strong options, at different numbers of steps. This indicates that, in general, pricing Strong options is more difficult than Weak options for lattice models. For analytical formulae, however, Strong-Out options are much harder to value than Weak-Out options while Strong-In options are easier to price than Weak-In options. Secondly, we compare analytical formulae and lattice models with a smaller number of steps (≤ 100) in pricing different types of options. To make the observation clearer, only Accel LR and Accel LRRE, the two top performing lattice models, are presented in Figure 3.8. When pricing Weak (In/Out) options, lattice models easily surpass analytical formulae since the curves are clearly closer to the origin relative to the four analytical formulae points. More specifically, Accel LRRE offers the best performance in pricing Weak options

---

[28] For pricing Strong-In options, the steps are ranged from 20 to 1,000. In addition, the accuracy and speed level of Accel LR for pricing Strong-In options improves more slowly than the other three lattice models as the number of steps increasing. To make the results comparable, we broaden the step range of Accel LR from 20 to 3,000.

when considering both efficiency and stability. On the contrary, for pricing Strong (In/Out) options, lattice models have no significant advantage over analytical formulae in achieving the same level of accuracy.[29] The four analytical formulae points are in the vicinity of the lattice models curves. However, we should not overlook the natural advantage of lattice models in offering granular combinations of speed and accuracy. In particular, the Accel LR provides the best performance in valuing Strong options. Thirdly, we consider four lattice models in terms of valuing varying types of options with a larger number of steps (>100) in Figure 3.9. The Accel TianBS&RE clearly performs best in pricing Strong-In and Strong-Out and offers relatively better performance in valuing Weak-In options. For Weak-Out options, Accel LRBS&RE is associated with the best performance by virtue that Accel TianBS&RE produces a lower level of efficiency and Accel LRRE lacks smooth convergence. In practice, we can determine the moneyness and the American quality of options (Strong/Weak/In/Out) and then select the appropriate lattice model to for estimation.

---

[29] This only applies in a narrow minority of cases by virtue that strong options are quite rare.

a) Strong out of the Money Sample

Method	Accel LR		Accel LRRE		Accel LRBS&RE		Accel TianBS&RE			
Steps	RMSRE	Time	RMSRE	Time	RMSRE	Time	RMSRE	Time	BAW	
10	3.0471%	2.43E-03	4.7669%	3.37E-03	4.7800%	3.69E-03	2.8573%	3.31E-03	RMSRE	Time
20	1.5064%	3.26E-03	1.7461%	3.32E-03	1.7593%	3.36E-03	0.9883%	3.30E-03	3.1896%	3.67E-03
30	0.9760%	3.33E-03	0.8530%	3.07E-03	0.9019%	3.57E-03	0.5264%	3.59E-03		
50	0.5733%	3.81E-03	0.3894%	3.74E-03	0.4090%	3.47E-03	0.4090%	3.84E-03	Bjerksund93	
100	0.2733%	3.75E-03	0.1655%	3.97E-03	0.1674%	4.83E-03	0.0788%	5.38E-03	RMSRE	Time
200	0.1339%	7.63E-03	0.0915%	9.35E-03	0.0924%	1.36E-02	0.0380%	1.79E-02	1.6936%	3.33E-03
300	0.0864%	1.55E-02	0.0492%	1.75E-02	0.0502%	2.68E-02	0.0186%	3.63E-02		
400	0.0638%	2.35E-02	0.0326%	2.88E-02	0.0334%	5.47E-02	0.0153%	5.64E-02	Bjerksund02	
500	0.0508%	3.57E-02	0.0193%	4.41E-02	0.0199%	7.43E-02	0.0105%	8.17E-02	RMSRE	Time
600	0.0414%	4.50E-02	0.0185%	6.00E-02	0.0188%	1.05E-01	0.0081%	1.12E-01	1.6980%	3.47E-03
700	0.0343%	6.22E-02	0.0172%	7.97E-02	0.0188%	1.12E-01	0.0086%	1.55E-01		
800	0.0294%	8.77E-02	0.0128%	1.02E-01	0.0131%	1.91E-01	0.0062%	2.09E-01	JuZhong	
900	0.0265%	9.89E-02	0.0091%	1.23E-01	0.0094%	2.28E-01	0.0065%	2.71E-01	RMSRE	Time
1000	0.0230%	1.18E-01	0.0109%	1.50E-01	0.0112%	2.77E-01	0.0055%	2.99E-01	2.0285%	3.29E-03

b) Strong in the Money Sample

Method	Accel LRRE		Accel LRBS&RE		Accel TianBS&RE			Accel LR			
Steps	RMSRE	Time	RMSRE	Time	RMSRE	Time	Steps	RMSRE	Time	BAW	
20	0.9092%	3.53E-03	0.9145%	3.50E-03	0.6363%	3.48E-03	20	0.8893%	2.99E-03	RMSRE	Time
30	0.6415%	3.42E-03	0.6421%	3.46E-03	0.3932%	3.64E-03	30	0.5916%	3.22E-03	0.5517%	3.35E-03
50	0.2822%	3.55E-03	0.2841%	3.82E-03	0.1869%	3.51E-03	50	0.3657%	3.28E-03		
100	0.1722%	3.58E-03	0.1724%	5.06E-03	0.1043%	4.94E-03	100	0.1851%	3.32E-03	Bjerksund93	
150	0.0994%	5.84E-03	0.0995%	9.98E-03	0.0677%	1.03E-02	150	0.1220%	4.31E-03	RMSRE	Time
200	0.0825%	9.51E-03	0.0826%	1.56E-02	0.0438%	1.44E-02	200	0.0921%	7.46E-03	0.2556%	3.27E-03
300	0.0516%	1.69E-02	0.0515%	3.17E-02	0.0339%	3.13E-02	300	0.0647%	1.34E-02		
400	0.0364%	2.79E-02	0.0364%	5.45E-02	0.0225%	5.02E-02	500	0.0366%	3.39E-02	Bjerksund02	
500	0.0297%	5.76E-02	0.0296%	8.18E-02	0.0167%	7.33E-02	800	0.0213%	7.59E-02	RMSRE	Time
600	0.0240%	6.85E-02	0.0240%	1.07E-01	0.0140%	1.05E-01	1000	0.0174%	1.32E-01	0.2378%	3.45E-03
700	0.0173%	8.87E-02	0.0173%	1.35E-01	0.0117%	1.41E-01	1500	0.0116%	2.61E-01		
800	0.0147%	9.55E-02	0.0148%	1.74E-01	0.0089%	1.86E-01	2000	0.0085%	4.73E-01	JuZhong	
900	0.0125%	1.14E-01	0.0125%	2.31E-01	0.0081%	2.34E-01	2500	0.0067%	6.92E-01	RMSRE	Time
1000	0.0113%	1.50E-01	0.0113%	2.59E-01	0.0073%	2.68E-01	3000	0.0054%	9.92E-01	0.1550%	3.41E-03

73

c) Weak out of the Money Sample

Method	Accel LR		Accel LRRE		Accel LRBS&RE		Accel TianBS&RE			
Steps	RMSRE	Time	RMSRE	Time	RMSRE	Time	RMSRE	Time	BAW	
10	0.4970%	2.82E-03	0.4675%	2.90E-03	0.5101%	3.01E-03	1.5608%	3.15E-03	RMSRE	Time
20	0.2418%	2.86E-03	0.0998%	2.98E-03	0.2667%	3.06E-03	0.3267%	3.51E-03	1.5850%	3.21E-03
30	0.1612%	3.24E-03	0.0724%	3.15E-03	0.1063%	3.34E-03	0.1509%	3.29E-03		
50	0.0971%	3.48E-03	0.0305%	3.50E-03	0.0521%	3.55E-03	0.0609%	3.63E-03	Bjerksund93	
100	0.0493%	4.43E-03	0.0077%	5.03E-03	0.0157%	5.76E-03	0.0188%	6.57E-03	RMSRE	Time
200	0.0252%	1.05E-02	0.0033%	1.26E-02	0.0061%	1.78E-02	0.0075%	1.85E-02	0.7797%	3.49E-03
300	0.0168%	1.95E-02	0.0034%	2.35E-02	0.0041%	3.48E-02	0.0049%	3.64E-02		
400	0.0128%	3.08E-02	0.0034%	3.80E-02	0.0039%	6.13E-02	0.0046%	6.41E-02	Bjerksund02	
500	0.0104%	4.80E-02	0.0034%	5.58E-02	0.0036%	8.53E-02	0.0042%	9.07E-02	RMSRE	Time
600	0.0088%	6.61E-02	0.0034%	8.35E-02	0.0035%	1.24E-01	0.0038%	1.33E-01	0.4372%	3.39E-03
700	0.0077%	8.29E-02	0.0034%	1.05E-01	0.0035%	1.60E-01	0.0036%	1.82E-01		
800	0.0069%	1.13E-01	0.0034%	1.41E-01	0.0034%	2.09E-01	0.0037%	2.29E-01	JuZhong	
900	0.0063%	1.38E-01	0.0034%	1.75E-01	0.0034%	2.61E-01	0.0035%	2.91E-01	RMSRE	Time
1000	0.0059%	1.69E-01	0.0034%	1.98E-01	0.0034%	3.22E-01	0.0035%	3.38E-01	0.2939%	3.89E-03

d) Weak in the Money Sample

Method	Accel LR		Accel LRRE		Accel LRBS&RE		Accel TianBS&RE			
Steps	RMSRE	Time	RMSRE	Time	RMSRE	Time	RMSRE	Time	BAW	
10	0.4463%	2.67E-03	0.3229%	2.78E-03	0.3143%	3.22E-03	0.2451%	3.22E-03	RMSRE	Time
20	0.2249%	2.73E-03	0.0747%	2.86E-03	0.0943%	3.44E-03	0.1007%	3.33E-03	0.6010%	3.43E-03
30	0.1543%	3.41E-03	0.0549%	3.46E-03	0.0598%	3.47E-03	0.1007%	3.56E-03		
50	0.0931%	3.51E-03	0.0409%	3.35E-03	0.0421%	3.69E-03	0.0248%	3.71E-03	Bjerksund93	
100	0.0458%	3.85E-03	0.0150%	4.77E-03	0.0151%	5.41E-03	0.0109%	5.39E-03	RMSRE	Time
200	0.0225%	9.49E-03	0.0069%	1.23E-02	0.0069%	1.75E-02	0.0050%	1.89E-02	0.4111%	3.38E-03
300	0.0149%	1.95E-02	0.0035%	2.39E-02	0.0035%	3.64E-02	0.0026%	3.84E-02		
400	0.0113%	3.06E-02	0.0030%	3.76E-02	0.0030%	5.65E-02	0.0019%	6.54E-02	Bjerksund02	
500	0.0088%	4.33E-02	0.0019%	5.41E-02	0.0019%	8.89E-02	0.0015%	9.80E-02	RMSRE	Time
600	0.0076%	6.06E-02	0.0019%	7.49E-02	0.0019%	1.18E-01	0.0015%	1.37E-01	0.3213%	3.23E-03
700	0.0065%	7.79E-02	0.0014%	9.89E-02	0.0014%	1.60E-01	0.0011%	1.94E-01		
800	0.0056%	1.04E-01	0.0012%	1.33E-01	0.0012%	2.06E-01	0.0009%	2.56E-01	JuZhong	
900	0.0050%	1.33E-01	0.0013%	1.58E-01	0.0012%	2.65E-01	0.0009%	2.70E-01	RMSRE	Time
1000	0.0046%	1.53E-01	0.0011%	1.96E-01	0.0010%	3.32E-01	0.0007%	3.17E-01	0.1299%	3.97E-03

Table 3.4: Comparing Accelerated LR and Tian models and analytical formulae for pricing strong/weak in/out-of the money options
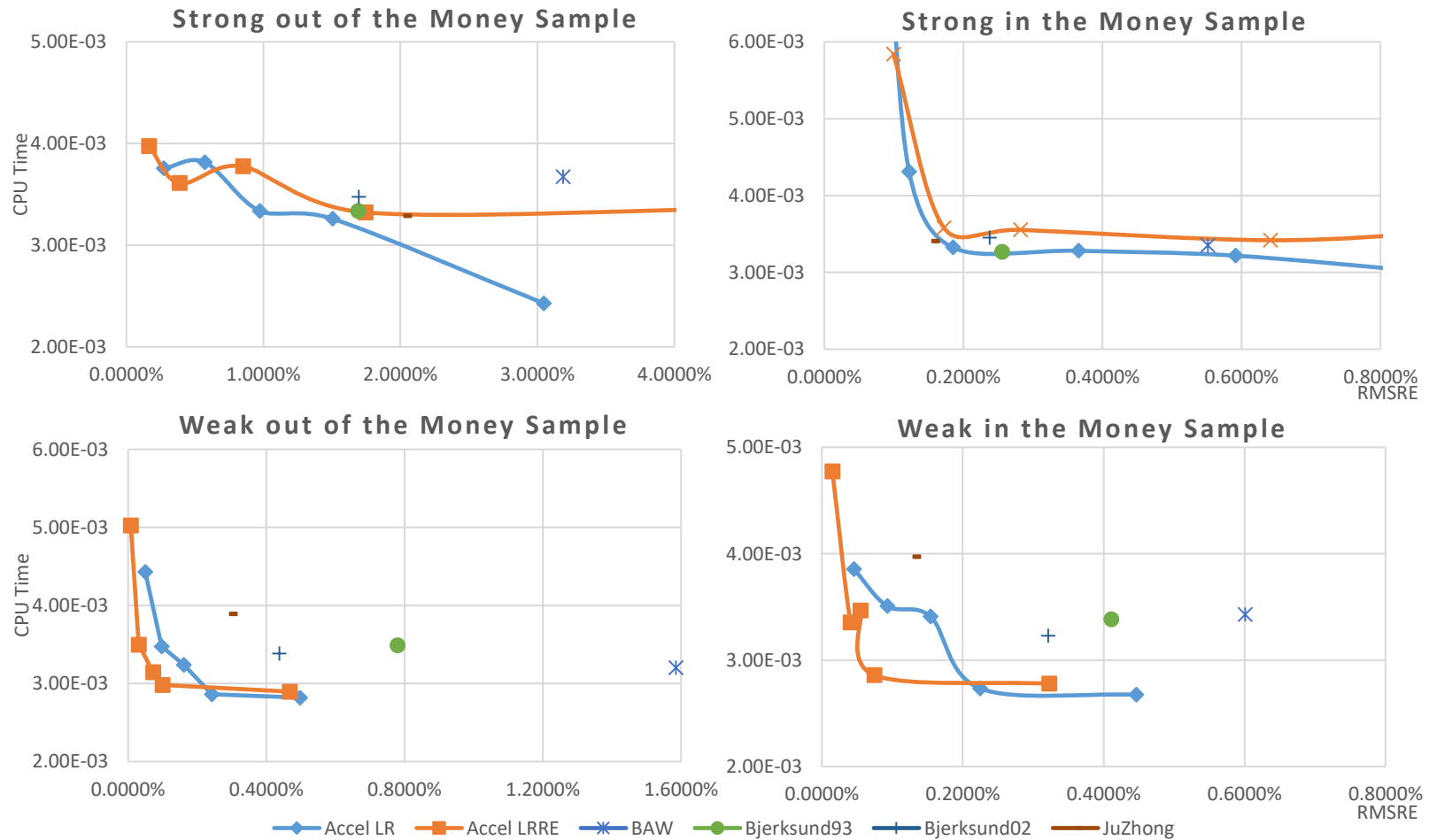

Figure 3.8: Comparing optimal lattice models with analytical formulae for pricing strong/weak in/out-of the money options with smaller step size (≤100)
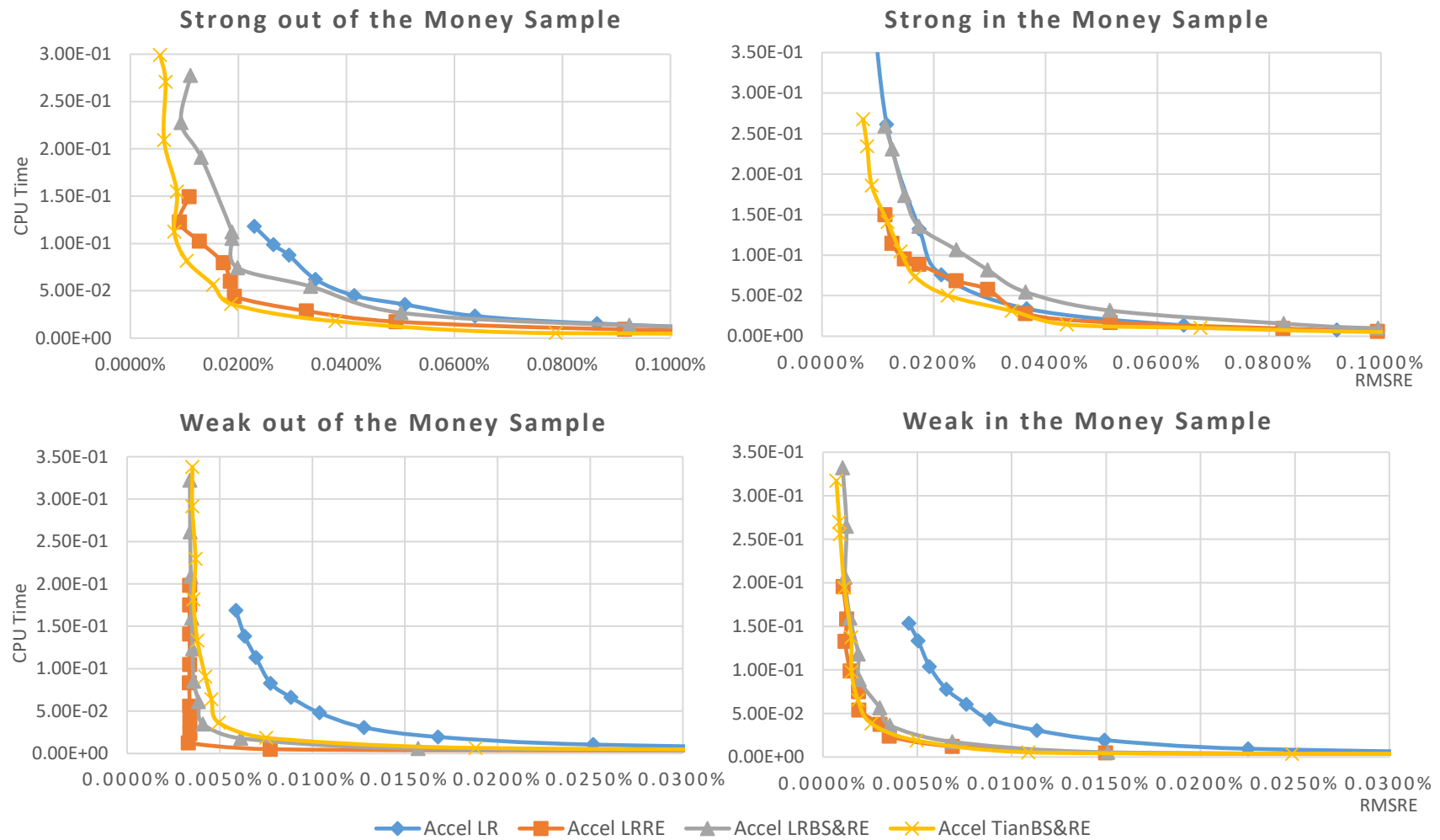
Figure 3.9: Comparing optimal lattice models for pricing strong/weak in/out-of the money options with larger step size (>100)

Conclusion

In this paper, we extend the insights developed in Shang and Byrne (2019) which were originally applied uniquely to the CRR tree. We set out the conditions under which the Intelligent Lattice Search Algorithm framework can be used to promptly detect the optimal exercise boundary for a LR and Tian tree. We apply dynamic memory and truncate the LR and Tian lattice where appropriate but are careful not to disturb the original estimation. Furthermore, Black-Scholes and Richardson extrapolation modifications to the tree structures can also be applied individually and/or together to improve the accuracy of the LR and Tian models. The speed and accuracy we obtain can be made to outperform the analytical models here. This holds for all four analytical models that are known to have wide dispersion of varying efficiency and error attributes. This finding reverses the received wisdom that closed form solutions do better in producing faster option values. Unlike the analytic models, the accelerated lattices however can also be made arbitrarily accurate. That is, lattices provide users with a highly granular pull-down menu choice to optimise speed relative to accuracy. One can move seamlessly between varying levels of precision.

Consistent with the literature, we also applied multi-dimensional efficiency tests. We benchmarked varying configurations of the LR and Tian models using several thousand random sets of parameters with each permutation linked to a given step size. This culminates with a very cleanly defined locus of accuracy relative to speed trade-offs for varying mesh densities. Our lower density mesh Accel LR and Accel LRRE lattices prove to be more efficient than better-known analytical models, achieving a higher level of accuracy. We can reach arbitrary levels of precision by using higher density mesh lattices. The Accel TianBS&RE model possesses the best performance compared with other LR and Tian models and the leading benchmark tree Accel CRR model. We further randomly generate extensive options

and divide them into four groups according to the deepness of American quality and the moneyness of the option. We found that four efficient lattice models, including Accel LR, Acccel LRRE, Accel LRBS&RE and Accel TianBS&RE, greatly surpass analytical formulae for pricing Weak options and have comparable performance with analytical formulae in valuing Strong options. Lattices importantly afford an explicit trade-off locus between accuracy and speed that can be navigated according to predetermined precision tolerance levels and option types. In practice, with higher tolerance to error (lower accuracy level), Accel LRRE is recommended for pricing Weak options and Accel LR should be applied to value Strong options. To obtain a higher level of accuracy (lower error tolerance), the Accel TianBS&RE is recommended in pricing Strong-In, Strong-Out and Weak-In options while Accel LRBS&RE should be used for valuing Weak-Out options. This should have practical relevance both in academia and trading platforms where real-time metrics like Implied Volatility are reported to traders at high very frequency.

Chapter 4: Accounting for Employee Stock Options: Accelerating Convergence

Abstract

The Hull-White binomial approach to pricing Employee Stock Options (ESOs) pricing makes explicit reference to parameters that are not available in Black-Scholes model yet are generally understood to be important for the valuation. Previous literature point out that a key weakness of the lattice approach, when applied to valuing ESOs, is the sluggish convergence behaviour not generally experienced in trees configured to estimate plain vanilla options. In this paper, we propose adjustments to the Hull-White model, based on insights developed by Boyle-Lau and Tian specifications. These ensure faster convergence in lattice estimation when barriers occur. The Hull-White-Boyle-Lau and Hull-White-Tian revamps expand the practicable menu choice available to accountants and other stakeholders in an ever evolving regulatory framework. The proposed models also provide efficient testing frameworks for validating a newer generation of closed-form solutions. We also adapt truncation and dynamic memory technologies considered in earlier chapters to the modify Hull-White model and further improve the efficiency in the appendix.

Literature Review

Employee Stock Options (ESOs) enhance the way business can craft payrolls to reward performance, encourage innovation and retain personnel with strategic skills and know-how. They permit staff to enjoy windfall gains when corporations are best placed to leverage stock market gains. Hall and Murphy (2002) reported that 94% of S&P 500 companies granted options to the top executives. Frydman and Saks (2007) stated that ESOs constitute over 40%

of their total compensation for the same cohort of executives. Murphy (2013) noted that the FASB 123R substantially levelled the playing field between stock and options by virtue of imposing a fair value accounting regime. S&P 500 corporations from 2004 typically lowered the amount of options granted to top executives and to rank and file employees. Chang, Fu, Low and Zhang (2015) however provide empirical evidence regarding the positive impact of non-executive employee stock options on corporate innovation. Lam and Chng (2006) also show that ESO grants have value implications for company performance by decreasing the cost of agency. Many factors contribute to cause and effect but ESOs appear likely to still have staying power and potentially will exert powerful financial effects where innovation is considered to be an important catalyst of economic growth.

Rubinstein (1995) identify key hallmark features that distinguish ESOs from other plain vanilla options. ESOs usually have longer maturities ranging from 5 to 10 years. They have vesting periods when the option cannot be exercised. Departure of staff from the company prior to vesting leads to forfeiture.[30] After the vesting period, the exercise of the option is possible even though in some instances suboptimal. These characteristics indicate that the option structure of ESOs is typically a hybrid with a blend of European and American exercise rights.[31] ESOs are necessarily by construction call options and in important ways align the interests of all stakeholders.[32] Recipients of ESOs are not permitted to hedge their option position using dynamic replication or put options. The non-transferability of the option to third parties also pertains. Recipients can however exercise the option and subsequently sell the company's shares to crystallise a cash pay-out. Hence, ESOs cannot be viewed as trading in "complete markets" as generally understood to exist in the classic Black and Scholes (1973) framework.

---

[30] The vesting period can run up to 4 years.
[31] We will argue that ESOs have significant features in common with barrier options.
[32] This is truer in the absence of debt.

In 1995, the Financial Accounting Standards Board (FASB) published the Financial Accounting Standard No. 123 (FASB 123) which recommends companies to recognise the cost of ESOs using fair value measurement instead of intrinsic value measurement. Furthermore, in 2004, the International Financial Reporting Standard No. 2 (IFRS 2) and the Revised Financial Accounting Standard No. 123 (FAS 123R), published by the International Accounting Standard Board (IASB) and Financial Accounting Standard Board (FASB) respectively, both require companies to report the fair value of ESOs. . In addition, Securities and Exchange Commission (SEC) issued Staff Accounting Bulletin No. 107 and 110 (SAB 107, SAB 110) in 2005 and 2007 respectively, which provided further interpretation of FAS 123R. In 2009, FASB integrated FAS 123R into the system of Accounting Standards Codification as Topic 718 (ASC 718) and issued an updated version in 2018 to expand the share based compensation for both employees and non-employees. IFRS 2, FAS 123R, SAB 107/110 and ASC 718 all suggest applying either an modified Black-Scholes model tuned for the expected early exercise or explicitly a binomial model to estimate the fair value of ESOs. In terms of estimating the expected term of ESOs, lattice models permit suboptimal exercise and calculate the expected term as a model output while the Black-Scholes model uses the expected term as an input. The Chinese Accounting Standards No. 11 (CAS 11)[33] issued by Ministry of Finance of the People's Republic of China in 2006 similarly require fair value measurement and permitted the use of the Black-Scholes formula to estimate the value of ESOs. The Black-Scholes model is overwhelmingly favoured by industry participants while academics favour binomial model. The Black-Scholes model is simpler to apply with less computational cost while lattice models have the potential to be more accurate and less restrictive in terms of limiting early exercise.

---

[33] http://english.mofcom.gov.cn/article/policyrelease/domesticpolicy/200606/20060602419802.shtml

The Black and Scholes (1973) model is typically applied to value ESOs with the estimated life of ESOs. The "expected life" strategy characteristic of the Black-Scholes model ESO implementation lacks any solid theoretical foundation and the estimation of the ESOs may not be accurate (Leung, 2009). The adjusted Black-Scholes model also overstates the cost of ESOs, which becomes more severe when the underlying stocks have high volatility (Hemmer et al., 1994; Marquardt, 2002; Finnerty, 2005). The modified Black-Scholes model proposed in Finnerty (2014), however, avoids the overpricing bias of Black-Sholes model and can be as accurate as but simpler to apply than trinomial lattice models. Binomial models are also developed to capture the special features and estimate the fair value of ESOs. Huddart (1994), Carpenter (1998) and Bettis et al. (2005) develop binomial models for pricing ESOs and measure the early exercise behaviour with the utility-based framework. The unobservable risk avoidance variable of utility-maximizing models, however, requires an adjustment contingent on availability of empirical data (Chendra and Sidarto, 2020).

The Hull and White (2004) ESO pricing model, based on Cox, Ross, and Rubinstein (1979) binomial model makes explicit an early-exercise barrier. Hull and White (2004) assumes that early exercise triggers when the stock price reaches a certain multiple of the strike price. This model has been widely accepted in the literature as an appropriate valuation method for ESOs. Not surprisingly many variants of it have been proposed. Ammann and Seiz (2004) similarly use the adjusted strike price to determine the exercise policy. Brisley and Anderson (2008) assume that employees exercise voluntarily when the intrinsic value of the option is greater than a certain proportion of its remaining Black-Scholes value. Chendra and Sidarto (2020) reconstruct Hull–White ESO model based on a bino-trinomial tree and replace the single exercise barrier with a moving barrier. In addition, Cvitanic, Wiener and Zapatero (2008)

propose a closed form solution for ESO pricing with the exercise policy also is modelled with a barrier trigger.

Lattice models are generally considered to provide a sounder theoretical treatment. Accountants nevertheless commonly rely on Black-Scholes model because considerable costs are incurred in switching to a binomial framework.[34] Binomial trees that embed Hull-White parameters inputs are typically slow to converge. This imposes nontrivial costs on analysts who wish to expedite valuation. Cvitanic, Wiener and Zapatero (2008) estimated a Hull-White tree and found that even after 40,000 steps the tree had not converged to "True".[35] The absence of a stable lattice may stymie the elaboration of new approaches to estimate ESOs. Standard setters and regulatory actors require a stable and efficient framework to test the robustness of a new generation of closed form solutions. In this paper, we focus on solving the sluggish convergence problem of Hull-White model. Since Hull-White ESOs pricing model incorporates an exercise barrier, several numerical methods for accelerating barrier options pricing could be applied to the Hull-White model. Boyle and Lau (1994) pointed out that the sluggish saw-tooth shaped convergence owes to the fact that the barrier rarely coincide with nodes generated within Cox Ross and Rubinstein tree. They proposed a simple solution that determines the appropriate step-size that co-ordinates best with where the barrier likely falls consistent with nodes. Other prominent solutions include Derman, Kani, Ergener and Bardhan (1995) who propose an interpolation method to generate the option value obtained in conjunction with an effective and modified barrier.[36] Figlewski and Gao (1999) decrease

---

[34] Companies are permitted to choose between Black and Scholes (1973) and lattice techniques.

[35] The execution speed of 3,000 steps with 2011 Mac Book pro Core i7 was clocking less than one minute. 40,000 steps was clocking close to one hour using C++ and Apple's proprietary Xcode IDE. In comparison, Black Scholes (1973) is not computationally intensive and executes generally without any observable lag. In the Appendix, we outline some techniques to truncate the ESO lattice and we dynamically define the binomial array.

[36] The effective and modified barrier is immediately higher and immediately lower than the true exercise barrier respectively.

approximation error by building a fine-mesh tree around the exercise price at maturity and embed highly granular mesh into the substrate lattice. Tian (1999) innovatively builds a two-stage flexible binomial tree that contains a tilted tree in the first stage and a conventional binomial tree in the second stage. In this paper, we elect to make improvements in the Hull-White lattice by incorporating Boyle and Lau (1994) and then Tian (1999) and demonstrate in each instance that convergence can be effectively accelerated. Moreover, consistent with Shang and Byrne (2019), we apply acceleration technologies including truncation and dynamic memory to the further improve Hull-White model in the appendix.

The remaining paper is organized as follows. In Section 2, we introduce the original Hull-White ESO pricing model and show how it can capture the special features of ESOs. In Section 3, we illustrate with examples the sluggish convergence behaviour of the Hull-White model and we also discuss the likely sources of error generated by lattice models. In Section 4 and 5, we propose Hull-White-Boyle-Lau and Hull-White-Tian models respectively. The Hull-White model is combined with Boyle-Lau and Tian specifications separately to accelerate the convergence. In Section 6, we compare the performance of original Hull-White model with Hull-White-Boyle-Lau and Hull-White-Tian models in pricing ESOs. In Section 7, we conclude. In the Appendix, we outline how the Hull-White-Boyle-Lau lattice can be strengthened by making practical use of truncation and dynamic memory techniques.

Methodology

Hull and White (2004) ESOs Pricing Model

The original pricing model proposed by Hull and White (2004) was conceived as an "*Enhanced FASB 123*" model. Adhering to the notation developed by Hull and White (2004), *e* was explicitly applied as an estimate of the employee turnover (exit) rate which was used in

both the vesting and the post-vesting period. The "Enhanced FASB 123" model incorporated early exercise by assuming that this would occur automatically when the stock price was a certain multiple M of the exercise price K. The Hull and White (2004) model reconfigured the Cox, Ross and Rubinstein (1979) binomial tree to incorporate added conditions for calculating the value of the option at each node.[37] v denotes the vesting period expressed in years and r represents the risk-free rate. There are N steps of δt length.[38] Hull and White (2004) specified that:

(i)   the option can be exercised in the post vesting period ($i\delta t > v$)

(ii)   the vested option is exercised when the stock price $S \geq M*K$ ($i\delta t > v$)

(iii)   the product of the time step δt and e can be used to estimate the probability of forfeiture in the vesting period moving from node to node ($i\delta t < v$)

(iv)   in the post vesting period the probability $e\delta t$ captures the probability that the option is forfeited if $S < K$ or exercised early to realise $S - K$ if $S \geq K$

These conditions are imposed on the conventional Cox, Ross and Rubinstein (1979) tree. See Figure 4.1. Here again, we follow the notation set out by Hull and White (2004). At each node of the tree, there is a specific array reference (i, j) for the respective stock price $S_{i,j}$ in the recombining tree. The value of the option at time period $i\delta t$ generated by backward induction is given by $f_{i,j}$.

$$f_{N,j} = \max(S_{N,j} - K, 0) \tag{4.1}$$

The "Enhanced FASB 123" model specifies:

---

[37] We recommend Hull (2011) to get an overview of the ESO and lattices techniques. Haug (2007) provides VBA code to estimate a variety of binomial and trinomial trees. Benninga (2008) also provides an overview of ESO estimation and VBA code for estimation.
[38] $S_{i,j}$ denotes the stock price at node (i, j). $S_{i,j} = Su^j d^{i-j}$ can be used to estimate iteratively the node values on the tree grid. $\delta t = T/N$ where T is the full ESO maturity.

If $(i\delta t > v)$ and $S_{N,j} \geq KM$ then

$$f_{i,j} = S_{i,j} - K \qquad (4.2)$$

If $(i\delta t > v)$ and $S_{N,j} < KM$ then

$$f_{i,j} = (1 - e\delta t)e^{-r\delta t}\left[pf_{i+1,j+1} + (1-p)f_{i+1,j}\right] + e\delta tmax(S_{i,j} - K, 0) \qquad (4.3)$$

If $(i\delta t \leq v)$ then

$$f_{i,j} = (1 - e\delta t)e^{-r\delta t}\left[pf_{i+1,j+1} + (1-p)f_{i+1,j}\right] \qquad (4.4)$$

The Hull and White (2004) model nests Equation (4.1) – (4.4) in a Cox, Ross and Rubinstein (1979) framework. Figure 4.1 deconstructs the Hull and White (2004) model, into varying domains mapped to a standard binomial tree. Equation (4.1) – (4.4) can be mapped to vesting and post vesting periods with an early exercise boundary **K*M** posited here as a barrier. We proceed by estimating plain vanilla trees to largely gauge the typical speeds of convergence that can be expected. In the absence of Equation (4.1) – (4.4), backward induction can be handled comfortably even within a basic spreadsheet. In what follows, plain vanilla European options are juxtaposed against a standard implementation of Hull and White (2004). A routine execution of Hull and White (2004) can however be slow to converge (see Figure 4.2), so much so, the binomial tree dimensions must be rescaled to provide a dramatically finer mesh with substantial computational costs incurred that are likely to deter extensive use of lattice techniques.[39]

---

[39] We make use of Boyle Lau (1994) and Tian (1999) specifications separately to improve the feasibility of ESO lattice estimation. A straight application of Hull and White (2004) in spreadsheets can be largely impractical when a high level of accuracy is required. In the Appendix, we also add some additional improvement in terms of estimation using truncation and by redefining the two-dimensional tree to being one-dimensional. We elaborate somewhat on how some computer science concepts find useful application in estimating the ESO lattice.
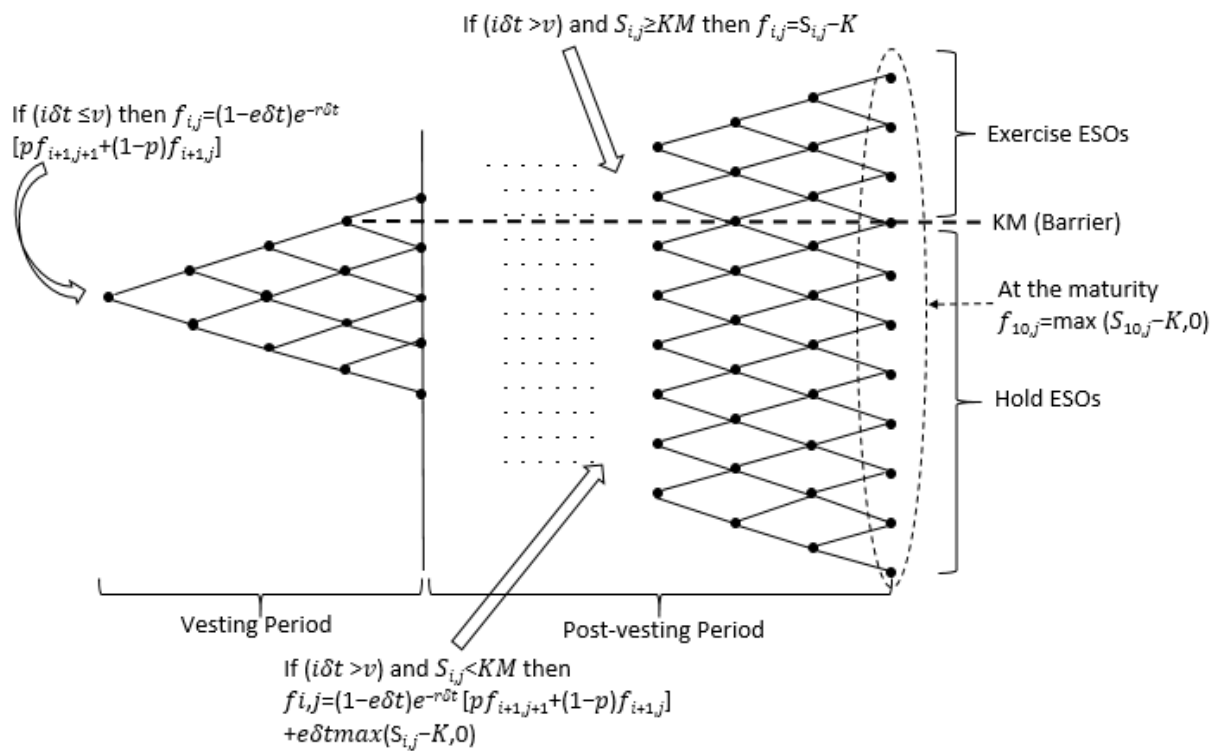
If ($i\delta t \leq v$) then $f_{i,j}=(1-e\delta t)e^{-r\delta t}$
$[pf_{i+1,j+1}+(1-p)f_{i+1,j}]$

If ($i\delta t > v$) and $S_{i,j} \geq KM$ then $f_{i,j}=S_{i,j}-K$

Exercise ESOs

KM (Barrier)

At the maturity
$f_{10,j}=max\,(S_{10,j}-K,0)$

Hold ESOs

Vesting Period

Post-vesting Period

If ($i\delta t > v$) and $S_{i,j} < KM$ then
$fi,j=(1-e\delta t)e^{-r\delta t}[pf_{i+1,j+1}+(1-p)f_{i+1,j}]$
$+e\delta t max(S_{i,j}-K,0)$

Figure 4.1: Hull and White (2004) model

Sluggish Convergence of Hull-White Model

Cvitanic, Wiener and Zapatero (2008) pointed out that Hull and White (2004) relied on CRR numerical methods to compute the value of the Employee Stock Option. They found that the Hull and White (2004) model converges very gradually and non-monotonically (which also creates problems for hedging computations). Cvitanic, Wiener and Zapatero (2008) reveal that convergence to "True" of a standard binomial implementation of Hull and White (2004) to be remarkably slow. In fact, at 40,000 steps the binomial ESO estimate Cvitanic et al (2008) obtained was 27.9291, when the "True" value was 27.8551. This finding may help explain why in practice accountants and other actors subjected to current American and European regulatory regimes may be slow to switch from using a Black Scholes type framework despite the latter being acknowledged as flawed for the purposes of estimating the value of ESOs. To give some scale of the estimation: when the number of time steps is equal to N (counting from 0), the

number of nodes in the CRR tree is equal to $(N+1)(N+2)/2$.[40] Estimation beyond 20,000

steps becomes computationally intensive for regular desktop computing resources and likely

unrealistic for those who have multiple option issues to frequently synthesize into Financial

Reports within short time frames.[41] This may, in part, explain the reluctance of the FASB and

SEC to exclusively impose a standard lattice framework on professional practitioners when

valuing ESOs. The SEC point to some theoretical evidence when providing this latitude.[42]

Steps	HW	True Value
10	31.8244	27.8551
20	28.1415	27.8551
50	28.9496	27.8551
70	29.2895	27.8551
100	29.8886	27.8551
200	29.0914	27.8551
500	28.4026	27.8551
1,000	28.28239	27.8551
1,500	27.9354	27.8551
2,000	27.9920	27.8551
2,500	28.2092	27.8551
3,000	28.1615	27.8551
4,000	27.9895	27.8551
5,000	28.0345	27.8551
7,500	27.9605	27.8551
10,000	28.0228	27.8551
20,000	27.8997	27.8551
100,000	27.8711	27.8551

Table 4.1: Sluggish convergence of Hull-White Model

In what follows, we graft on to the Cox, Ross and Rubinstein (1979) tree the Hull and White

(2004) conditions (i) – (iv) outlined previously. In Table 4.1, we report the value of the Hull

and White (2004) tree using the Cvitanic, Wiener and Zapatero (2008) parameters $S = 100$, K

---

[40] See Haug (2007) p.286. $N = 40,000$ steps would imply in excess of 800 million (i, j) nodes!!

[41] In our version of Microsoft Visual Studio Community 2017, C++ code did not execute at 30,000 steps without requests to reconfigure software specifications. Excel VBA crashed at 20,000 steps however some additional tweaking of code would have allowed higher specifications.

[42] SAB Topic 14 provides some guidance "The staff notes the existence of academic literature that supports the assertion that the Black-Scholes-Merton closed-form model, with expected term as an input, can produce reasonable estimates of fair value." See: https://www.sec.gov/interps/account/sabcodet14.htm

= 100, T = 10 years, *Sigma* = 0.2, r = 0.06, *Vesting* = 2 years, M = 1.5 and e = 0.04. The "True"

value of 27.8551 is only reached with a criterion of two decimal places accuracy when the

number of steps in the tree is raised to 100,000 steps. The large number of steps required to get

close to convergence is problematic.[43] We also believe that this type of erratic behaviour may

influence practitioners to rely more on a Black Scholes type framework despite the latter being

only reliable for valuing options with European expiry/exercise privileges. Cvitanic, Wiener

and Zapatero (2008) pointed out that convergence is not uniform, thus many estimations of the

same tree are typically required to confirm that convergence has actually taken place.[44] The

familiar saw tooth pattern commonly associated with Barrier Options is evidenced in Figure

4.2. The Hull and White (2004) ESO model is estimated from 10 to 10,000 steps in intervals

of 10. It is clear that convergence is difficult to attain, and a very large step magnitude tree

would be required to remove persistent error which manifests itself again and again. To

compound matters, a steadily growing number of steps does not manifestly yield a consistent

reduction in error.

---

[43] An accountant or financial analyst would not easily devote the necessary computing resources and time to extract fair value if the requisite tree step size was of the order of 100,000.

[44] We don't quite know when we have got there and analysts could be conceivably forced to repeat a number of implausibly elevated step magnitude trees to confirm actual convergence. Accountants and other practitioners understandably may feel obliged to accept Black Scholes (1973) based methodologies to estimate the fair value of option compensation even though the model is not entirely appropriate for ESOs. Perhaps, "better the devil you know".
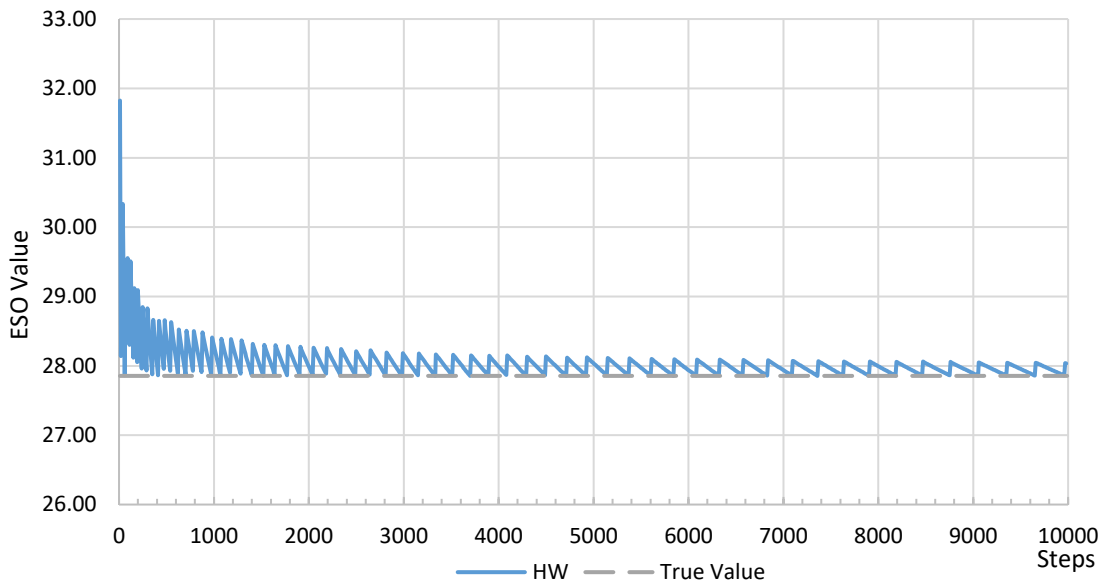
Figure 4.2: Sluggish convergence of Hull-White Model

Derman et al (1995) pointed out that there at least two sources of inaccuracy when modelling options using a lattice.[45] The first derives from the fact that the lattice is a discrete time representation of a continuous time process. As the mesh becomes finer, the level of inaccuracy can be reduced. The second source of bias may be explained by a failure of the lattice to precisely capture the terms of the option.[46] The Hull and White (2004) model embeds a specification for early exercise. As mentioned previously, in Section 2: when the stock price $S \geq M^*K$, the vested option is exercised ($i\delta t > v$).

We argue here that M^*K represents a form of barrier in the Hull and White (2004) model which can be triggered at any time in the post vesting period.[47] Derman et al (1995) pointed out that if a barrier does not coincide with a series of horizontal nodes on the recombining tree then systematic biases emerge in estimation. This type of 'specification error' is identified in Figure

---

[45] Derman et al (1995) referred to "quantization error" and "specification error". We address the latter using the Boyle and Lau (1994) procedure.

[46] Derman et al (1995) referred to this as specification error.

[47] We interpret the ESO as configured by Hull and White (2004) to embed *American* barrier exercise privileges.

4.3. One solution proposed by Boyle and Lau (1994) would involve aligning the tree or a series of horizontal nodes to **M*K**. This necessitates some manipulation of the array mapping to produce recombined equivalent values that plot close to or at the barrier. To correct for tree misalignment Boyle and Lau (1994) recommend that the lattice is designed so that the barrier falls on a series of successive recombining nodes, see Figure 4.4. This has the effect of placing the barrier on the nodes and 'specification error' as described by Derman et al (1995) can be mitigated for estimation purposes.
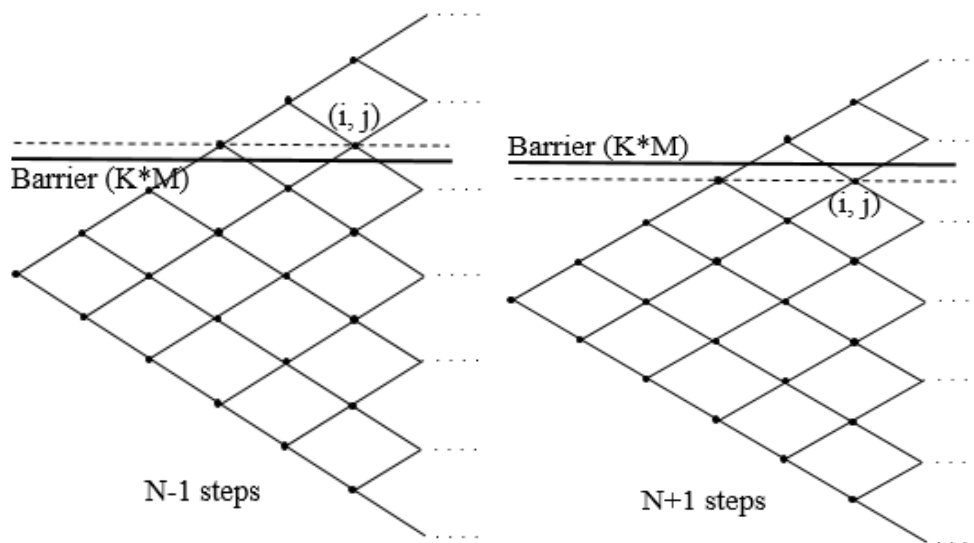
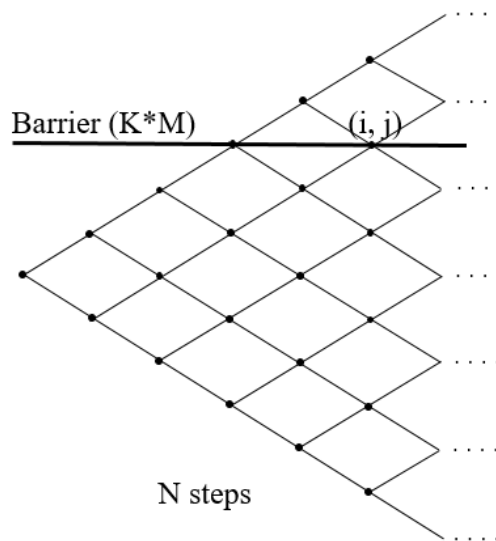Figure 4.3: Discordance between the barrier and horizontal nodes of the tree

Figure 4.4: The barrier falling on a series of horizontal nodes of the tree

Hull-White-Boyle-Lau Model

Boyle and Lau (1994) found that the effect of the barrier was to render convergence to be saw tooth shaped. Typically for barrier options: a consistently increasing step size does not produce a consistent diminution in error. Boyle and Lau (1994) advocated that the step size be restricted to values that produce nodes that match the barrier. In other words, the number of steps, should be selected so that an array of recombining nodes, in the tree, line up or nearly line up with the barrier. The Hull and White (2004) model is nested in a Cox, Ross and Rubinstein (1979) tree and has an early exercise boundary of *K*M*. Applying Boyle and Lau (1994) to Hull and White (2004) model, we make the barrier *K*M* falls on any one node of the tree, defined as:

$$S_{0,0} * u^i = K * M \tag{4.5}$$

where $S_{0,0}$ is the initial stock price and *u* is the magnitude of upward factors defined as:

$$u = e^{\sigma\sqrt{\Delta t}} = e^{\sigma\sqrt{\frac{T}{N}}} \tag{4.6}$$

where σ is volatility, *T* is time to maturity, *N* is the step number and Δt is the time step. Rearranging Equation (4.5), we obtain the function for selecting the appropriate *N*:

$$N = \left\{ \frac{(i^2\sigma^2 T)}{\left[\ln(\frac{KM}{S_{0,0}})\right]^2} \right\}, \text{ for } i = 1, 2, 3\ldots \tag{4.7}$$

where { . } means the closest integer to its argument and the step number *N* of Hull-White model can be calibrated by iterating *i*. The effect of Boyle-Lau method is visualized in Figure 4.5. The solid line represents the estimated value of an ESO produced by Hull-White model as the number of steps increases while the dashed horizontal line represents the true value of this ESO. The dots represent the value produced by Hull-White-Boyle-Lau model in which the

number of steps is selected to fuse the barrier $K*M$ as close as possible to a horizontal locus of tree nodes. Compared with the original Hull-White model whose estimations are fluctuating, the values produced by Hull-White-Boyle-Lau (dots) are visually consistent with the true values. Boyle and Lau (1994) iteratively re-estimated successive values of i to yield the desired N or integer closest to the desired N using Equation (4.7) so that the 'specification error' as explained by Derman et al (1995) can be reduced. If the barrier does not coincide with a specified locus of nodes, monitoring will in the American option's framework be repetitively out of sync. Equation (4.7) applies most effectively where there is a single barrier and volatility is constant. If M varies or if the tree is not consistent with Cox, Ross and Rubinstein (1979) then Boyle and Lau (1994) may not be appropriate.[48]
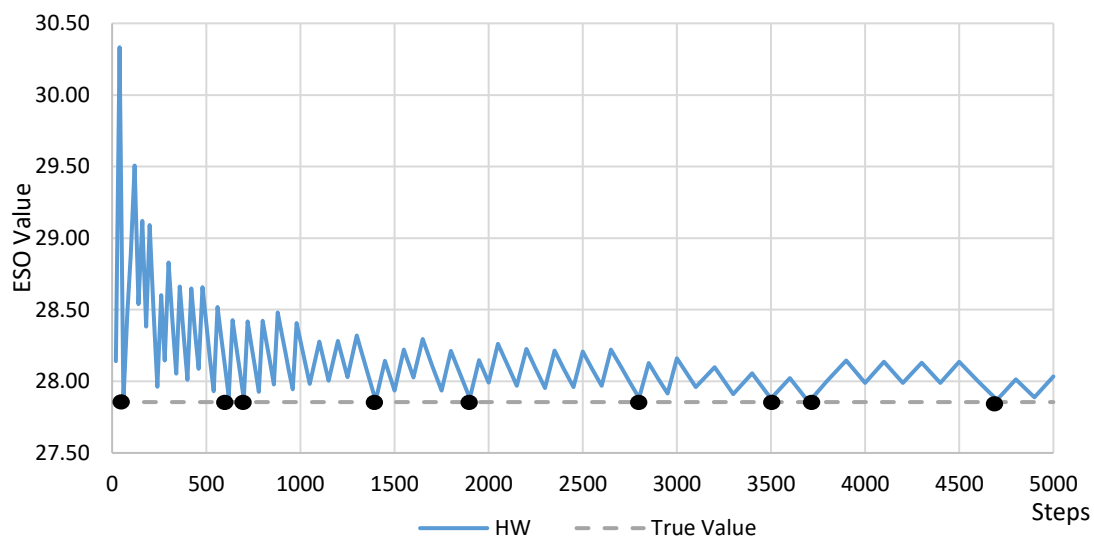

Figure 4.5: Hull-White-Boyle-Lau methods

---

[48]The Leisen Reimer (1996) tree does not recombine iteratively to a fixed range of values. Their model tilts the lattice so that the terminal stock prices centre towards the exercise. Placing the barrier in the Leisen Reimer (1996) tree to rest on the recombined nodes is not straightforward.

Hull-White-Tian Model

Tian (1999) proposed a two-stage flexible binomial methods for continuous barrier options, which ensures a layer of nodes coincides with the barrier. In our paper, we adapt two-stage flexible binomial method to the original Hull and White (2004) model, which eliminates the second source of bias and leads to the accurate estimations. As is shown in Figure 4.6, the Hull-White-Tian binomial tree consists of two different stages, separating by the vertical solid line. The first stage is defined as the first N_0 steps tree ($n \leq N_0$) where N_0 is a particular number of steps. In this stage, the binomial tree is tilted (solid tilted tree) with a non-zero tilt parameter λ, twisted from a normal binomial tree (dashed non-tilted tree). This is design to make the highest node (N_0, N_0) of the tilted tree fall exactly on the "early exercise barrier" $K*M$ (horizontal solid line). The second stage is the rest of the tree with the step number $n > N_0$. In this stage, the tree returns back to the normal non-tilted binomial tree with zero tilt parameter so that a horizontal layer of nodes where node (N_0, N_0) coincides with the early exercise barrier.
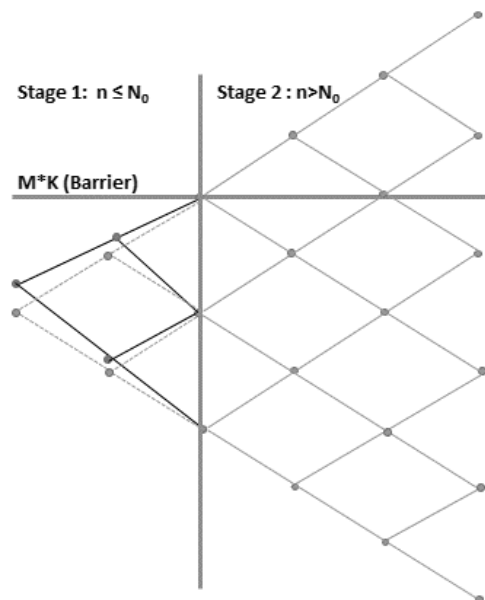

Figure 4.6: Two-stage Hull-White-Tian model

The stock price and option price of each node in these two stages are governed by two different sets of parameters. In the first stage, the degree of stock price moving up and down, u and d, are influenced by the tilt parameter λ:

$$u = e^{\sigma\sqrt{\Delta t} + \lambda\sigma^2\Delta t}, \tag{4.8}$$

$$d = e^{-\sigma\sqrt{\Delta t} + \lambda\sigma^2\Delta t}, \tag{4.9}$$

where λ is the tilt parameter. Taking the dividends q into account, the possibility of upward and downward movement p and 1-p is defined as:

$$p = \frac{e^{(r-q)\Delta t} - d}{u - d}, \tag{4.10}$$

$$1 - p = \frac{u - e^{(r-q)\Delta t}}{u - d}. \tag{4.11}$$

In addition, non-negative possibilities of upward and downward movement require:

$$d \leq e^{(r-q)\Delta t} \leq u, \tag{4.12}$$

which can be rewritten as:

$$\lambda - \frac{r - q}{\sigma^2} \leq \frac{1}{\sigma\sqrt{\Delta t}}. \tag{4.13}$$

It is not hard to observe that Equation (4.13) always establishes when Δt is sufficiently small. In the second stage, the tilted tree returns back to the normal non-tilted binomial tree. The upward and downward movement of stock price, u_0 and d_0, and the corresponding possibility, p_0 and 1-p_0, are given by:

$$u_0 = e^{\sigma\sqrt{\Delta t}}, \tag{4.14}$$

$$d_0 = e^{-\sigma\sqrt{\Delta t}}, \tag{4.15}$$

$$p_0 = \frac{e^{(r-q)\Delta t} - d_0}{u_0 - d_0},$$

(4.16)

$$1 - p_0 = \frac{u_0 - e^{(r-q)\Delta t}}{u_0 - d_0}.$$

(4.17)

The λ plays an important role here since it determines the shape of the tilted tree in the first stage and further determines whether the barrier can fall on a layer of nodes in the second stage. The value of λ is determined as follows: We start from the second stage, normal non-tilted binomial tree ($n > N_0$ and $\lambda = 0$), and determine the particular step number N_0. A horizontal layer of nodes in the second stage coincides with the "early exercise barrier" which can be represented as:

$$S_{0,0}u_0^{\,j} = MK,$$

(4.18)

Therefore, the solution for j is given by:

$$j = \frac{ln(\frac{MK}{S_{0,0}})}{ln(u_0)}.$$

(4.19)

Since j is barely an integer, N_0 is determined by:

$$N_0 = [\,j\,] = \left[\frac{\ln\left(\frac{MK}{S_{0,0}}\right)}{\ln(u_0)} \right],$$

(4.20)

where [.] means the closest integer to its argument. Moving backward to the first stage, the stock price of the highest node (N_0, N_0) in this stage is given by:

$$S_{N_0,N_0} = S_0 u^{N_0},$$

(4.21)

To ensure node (N_0, N_0) falls exactly on the barrier, the following Equation should establish:

$$S_{0,0}u^{N_0} = MK .$$ (4.22)

Therefore, by Equation (4.8), λ is solved as:

$$\lambda = \frac{ln\left(\dfrac{MK}{S_{0,0}}\right) - N_0\sigma\sqrt{\Delta t}}{N_0\sigma^2\Delta t} .$$ (4.23)

With the appropriate determination of λ, we produce a layer of nodes in the second stage consistent with the highest node in the first stage (N_0, N_0) coincides with the barrier. Moreover, Tian (1999) confirmed that the flexible binomial tree permits the second stage to recombine even though the set of parameters change between the two stages. By eliminating the 'specification error' stated in Derman et al (1995), Hull-White-Tian model can obtain the accurate value of ESOs with much less steps relative to the original Hull-White model.

Numerical Results

We first re-estimate the Hull and White (2004) model by incorporating Boyle and Lau (1994) and Tian (1999) respectively. The results are shown in Table 4.2. The same Cvitanic, Wiener and Zapatero (2008) parameters are used as Table 4.1: $S = 100$, $K = 100$, $T = 10$ years, *Sigma* $= 0.2$, $r = 0.06$, *Vesting* $= 2$ years, $M = 1.5$ and $e = 0.04$. Convergence is substantively improved relative to the original Hull-White model shown in Table 4.1. At a step size of 1,500, there would appear to be a relatively stable pattern of convergence already established. The saw tooth pattern of convergence identified in Figure 4.2 seems to have been largely resolved. In Figure 4.7, the relatively parsimonious refinement introduced by Boyle and Lau (1994) and Tian (1999) seems to mostly mitigate the sluggish convergence issues identified by Cvitanic, Wiener and Zapatero (2008) when implementing Hull and White (2004) in a traditional Cox, Ross and Rubinstein (1979) setting. More specifically, Hull-White-Boyle-Lau possesses the

smoothest convergence while Hull-White-Tian provides the best accuracy level. Hull-White-Boyle-Lau and Hull-White-Tian models offers some scope particularly for actors in the professional and regulatory bodies who default to the Black and Scholes (1973) framework by virtue of lower computational cost.

Steps	HW	HWBL	HWTian	True Value
50	28.9496	27.8632	27.7599	27.8551
100	29.8886	27.8675	27.9468	27.8551
200	29.0914	27.8715	27.8365	27.8551
300	28.8294	27.8782	27.8436	27.8551
400	28.0118	27.8480	27.8464	27.8551
500	28.4026	27.8675	27.8713	27.8551
1,000	28.2824	27.8530	27.8627	27.8551
1,500	27.9354	27.8549	27.8566	27.8551
2,000	27.9920	27.8576	27.8581	27.8551
2,500	28.2092	27.8565	27.8527	27.8551
3,000	28.1615	27.8562	27.8553	27.8551
3,500	27.8799	27.8563	27.8569	27.8551
4,000	27.9895	27.8566	27.8549	27.8551
4,500	28.1356	27.8569	27.8542	27.8551
5,000	28.0345	27.8535	27.8540	27.8551

Table 4.2: Fast convergence of Hull-White-Boyle-Lau and Hull-White-Tian models
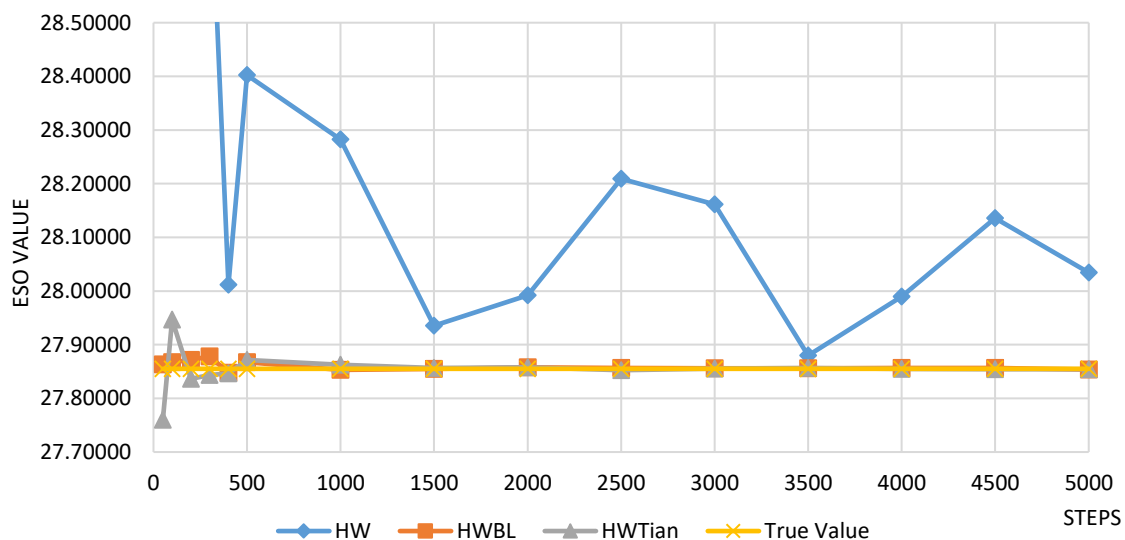

Figure 4.7: Comparing Hull-White with Hull-White-Boyle-Lau and Hull-White-Tian models

In Table 4.3, we alter the standard values we used so far: $S = 100$, $K = 100$, $T = 10$ years, *Sigma* $= 0.2$, $r = 0.06$, *Vesting* $= 2$ years, $M = 1.5$ and $e = 0.04$. The third column contains the estimations for original values. We compare Hull-White-Boyle-Lau (HWBL) and Hull-White-Tian (HWTian) relative to the Hull White estimation (HW). In each column, we can observe convergence as we increase the step size. For each successive column we re-compute these values amending a single variable e.g. S' denotes that we amend just a single variable in this case S moves from 100 to 120. The true values we estimated using Cvitanic, Wiener and Zapatero (2008).[49] Again it is clear that HWBL and HWTian values are clearly closer to True relative to the unadjusted Hull and White (2004).

In Table 4.4, we measure the bias based on the estimations in Table 4.3. Referring to Broadie and Detemple (1996), root mean squared relative error (RMSRE) is defined as the estimation error measurement. First, relative deviation e_i is defined by:

$$e_i = \frac{\widehat{ESO_i} - ESO_i}{ESO_i}, \qquad\qquad (4.24)$$

where $\widehat{ESO_i}$ is the estimated ESO value while ESO_i is the true value of ESO. Therefore, by Equation (4.24), RMSRE is determined by:

$$\text{RMSRE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n} e_i^2}, \qquad\qquad (4.25)$$

where n is the number of samples which is equal to 100 here. As it is shown in Table 4.4, the relative error of HWBL and the HWTian model is around 50 times less than that of original HW model. More specifically, HWTian achieved a slightly higher accuracy level than HWBL since the RMSRE of HWTian is a little lower than that of HWBL.

---

[49] We thank Professor Zvi Wiener for confirming the true values.

Methods	Steps	Initial	Multiple'=2	ExitRate'=0.08	DivRate'=0.03	Sigma'=0.4	Interest'=0.03	Vest'=4	T'=8	K'=120	S'=120
						ESO price					
HW		28.4026	35.2359	25.1287	21.7337	36.6282	24.5140	30.3561	27.4778	23.7446	39.6460
HWBL	500	27.8675	34.9800	24.7002	21.4729	35.4646	24.1361	30.0290	26.8391	23.5734	39.5246
HWTian		27.8713	39.5549	23.5637	26.8407	30.0587	24.1350	35.4351	21.4704	24.6991	34.9812
HW		28.2824	35.1153	25.0408	21.6802	36.2675	24.4317	30.2862	26.9971	23.9294	40.0232
HWBL	1000	27.8530	35.0000	24.6945	21.4661	35.4434	24.1243	30.0340	26.8557	23.5789	39.5137
HWTian		27.8627	39.5116	23.5712	26.8462	30.0398	24.1251	35.4342	21.4677	24.6965	34.9886
HW		27.9354	35.1598	24.7621	21.5080	35.8913	24.1836	30.0867	26.9350	23.5752	39.6849
HWBL	1500	27.8549	34.9970	24.6967	21.4670	35.4100	24.1254	30.0406	26.8425	23.5729	39.5163
HWTian		27.8566	39.5167	23.5710	26.8468	30.0407	24.1267	35.4266	21.4685	24.6987	34.9913
HW		27.9920	35.2537	24.8097	21.5377	35.8827	24.2250	30.1190	27.1516	23.7541	39.6131
HWBL	2000	27.8576	34.9959	24.7013	21.4698	35.4265	24.1280	30.0377	26.8496	23.5728	39.5125
HWTian		27.8581	39.5115	23.5741	26.8455	30.0431	24.1280	35.4255	21.4697	24.7009	34.9960
HW		28.2092	35.0412	24.9871	21.6473	36.0236	24.3813	30.2437	26.8929	23.7677	39.6518
HWBL	2500	27.8565	34.9956	24.7024	21.4712	35.4174	24.1290	30.0423	26.8486	23.5731	39.5176
HWTian		27.8527	39.5156	23.5729	26.8464	30.0391	24.1253	35.4212	21.4686	24.6989	34.9953
HW		28.1615	35.2041	24.9492	21.6239	35.6817	24.3473	30.2173	27.0451	23.6028	39.7450
HWBL	3000	27.8562	34.9978	24.7021	21.4709	35.4251	24.1285	30.0400	26.8479	23.5715	39.5147
HWTian		27.8553	39.5097	23.5733	26.8474	30.0401	24.1267	35.4165	21.4693	24.7003	34.9962
HW		27.8799	35.0269	24.7207	21.4818	35.4391	24.1445	30.0555	26.9948	23.5764	39.5519
HWBL	3500	27.8563	34.9970	24.7023	21.4709	35.4160	24.1285	30.0401	26.8488	23.5710	39.5109
HWTian		27.8549	39.5106	23.5730	26.8466	30.0390	24.1267	35.4224	21.4695	24.7007	34.9956
HW		27.9895	35.1241	24.8100	21.5378	35.7770	24.2240	30.1174	27.0018	23.6359	39.7116
HWBL	4000	27.8566	34.9969	24.7031	21.4713	35.4199	24.1289	30.0419	26.8463	23.5710	39.5129
HWTian		27.8549	39.5107	23.5714	26.8464	30.0392	24.1268	35.4169	21.4697	24.7010	34.9963
HW		28.1356	35.0808	24.9292	21.6115	35.6567	24.3291	30.2020	27.0469	23.7530	39.6070
HWBL	4500	27.8569	34.9963	24.7026	21.4705	35.4263	24.1281	30.0421	26.8471	23.5724	39.5106
HWTian		27.8553	39.5110	23.5723	26.8482	30.0402	24.1270	35.4199	21.4698	24.7013	34.9959
HW		28.0345	35.0806	24.8472	21.5608	35.5666	24.2565	30.1438	26.9118	23.6468	39.5255
HWBL	5000	27.8535	34.9960	24.7006	21.4694	35.4221	24.1262	30.0397	26.8480	23.5720	39.5126
HWTian		27.8540	39.5136	23.5717	26.8471	30.0397	24.1265	35.4212	21.4697	24.7011	34.9965
True Value		27.8551	34.9973	24.7031	21.4707	35.4181	24.1276	30.0402	26.8472	23.5720	39.5111

Table 4.3: Comparing Hull-White, Hull-White-Boyle-Lau and Hull-White-Tian model

	HW	HWBL	HWTian
Average deviation(*10^{-3})	6.8355	0.1093	0.1041
largest deviation (*10^{-3})	34.1672	1.3129	1.1082
Number of deviation > 1%	23	0	0
RMSRE (*10^{-3})	8.9274	0.2034	0.1892

Table 4.4: Comparing Hull-White, Hull-White-Boyle-Lau and Hull-White-Tian model

Conclusion

Standard setters by and large have not been implacably rigid in asserting a specific valuation framework for those tasked with preparing company accounts and expensing Employee Stock Options. Choice of model and parameter inputs however have nontrivial effects and varying estimates of valuation can be asserted. This may be a source of disquiet for the accounting profession and to regulatory actors such as the SEC. [50] The ability of shareholders and bondholders to appraise the incentives propelling senior executives can also be easily circumscribed by arbitrary choices within the valuation process. In this paper, we consider the impact of valuation and convergence of lattice models, in particular Hull and White (2004) model. We address the problem of Hull-White binomial model (HW) that the inability to reach convergence within a feasible number of steps may render lattice techniques impracticable for most practitioners. Regarding the Employee Stock Option as an American Barrier Options, we borrow established techniques in the literature including Boyle and Lau (1994) and Tian (1999) to improve implementation. We develop an efficient framework Hull-White-Boyle-Lau (HWBL) and Hull-White-Tian (HWTian) models to make the valuation process achieve greater accuracy with smaller computational costs. With the same number of steps, the accuracy level of HWBL and HWTian model is around 50 times higher than the original HW model, which indicates the convergence can be effectively accelerated. In the Appendix, we further improve

---

[50] The margin of error produced may be acceptable for producing Financial Statement in the same way that depreciation of asset estimations also necessarily must reside with a margin of error.

the HWBL algorithm by making better use of computer memory and truncating redundant

region, which can also be extended to Hull-White-Tian model.[51] The software implementation

can be made more efficient by defining the array mapping differently for the lattice.

---

[51] Some readers may prefer to read the Appendix separately from the rest of the paper or not at all. The biggest improvement in estimation comes from adapting the Boyle Lau (1996) specification to Hull White (2004). We attempt in the Appendix, to squeeze out some additional efficiencies in terms of run-time by making better use of memory. Some practitioners may appreciate these additional gains in efficiency particularly where the humble spreadsheet is the main pricing engine.

Chapter 5 - Conclusion

Summary

This thesis focuses on the valuation of American options and employee stock options, especially lattice-based valuation models. We begin in Chapter 1 with the introduction to the role options play and some background on American options, pricing and hedging. We then extend the analysis to include valuation methods and approaches likely countenanced to evaluate employee stock options.

In Chapter 2, an intelligent lattice search algorithm is introduced to efficiently locate the optimal exercise boundary for American options. An accelerated CRR binomial model is proposed by incorporating intelligent lattice search, truncation and dynamic memory. The computational runtime can be reduced from over 18 minutes down to less than 3 seconds to estimate a 15,000-step binomial tree without disturbing the accuracy. Delta and Implied Volatility can also be accelerated relative to standard models. The standard binomial trees, usually considered to be slow and less practical, are modified to be comparable with analytical formulae in terms of speed. More importantly, intelligent lattice search can be tweaked to reach varying levels of accuracy corresponding to different step size. Conventional analytical formulae are less flexible and offer less scope for improving accuracy.

Chapter 3 extends and improves the accuracy-speed trade-off, introduced by intelligent lattice search, for American Option valuation. Lattice techniques are further developed to accelerate computational time and accuracy for LR and Tian models. The improved LR and Tian models coupled with intelligent lattice search can achieve a level of accuracy and efficiency combined that surpass analytical analogues prominent in the literature. The performance of efficient lattice models is benchmarked against analytical formulae for a large number of different parameters, randomly generated. Results reveal that different pricing techniques are more

appropriate for out of-the-money, in-the-money, strong and weak options. The choice of model can be optimally selected to harness the greatest level of efficiency. It was found that lattices fused with intelligent lattice search affords an explicit trade-off locus between accuracy and speed that can be navigated according to predetermined precision tolerance levels. These models, on the whole, surpassed leading analytical techniques for both speed and accuracy. This improvement should have practical relevance to computationally expensive trading platforms that require real-time estimates of implied volatility.

In Chapter 4, we extend this analysis to the valuation of non-traded employee stock options. The well documented sluggish convergence of the celebrated Hull-White ESO pricing model can be substantially mitigated by combining Boyle-Lau and Tian specifications with the techniques introduced in Chapters 2 and 3. It has been found that the relative error of Hull-White-Boyle-Lau and Hull-White-Tian models are around 50 times less than the original Hull-White model using the same number of steps. The Hull-White model can also be further accelerated using the lattice techniques previously designed for American options. Hull-White-Boyle-Lau and Hull-White-Tian models extend the practicable menu choice available to stakeholders in developed and emerging market regimes who are varyingly subjected to accounting standards that increasingly reflect fair value imperatives. These techniques also provide an efficient testing framework for validating a newer generation of closed-form solutions also now appearing in the literature.

Future Research

Machine learning paradigms potentially offer a promising pathway to traverse the performance of the option pricing models and to elaborating more effectively hedging strategies. Predictive modelling, in principle, could be useful for classifying options as being some combination of

in/at/out-of-the-money and strong/weak. Knowing in advance the appropriate classification should improve model selection and performance. Strong options while relatively rare, account for a substantial proportion of error when pricing. Developing an efficient algorithm that bifurcates strong and weak options should help when pricing large books of options. In addition, the lattice-based ESOs pricing models presented by Brisley and Anderson (2008) and Abudy and Benninga (2013) are likely candidates that feasibly can be enhanced by intelligent lattice search.

References

Amin, K., & Khanna, A. (1994). Convergence of American option values from discrete-to continuous-time financial models. *Mathematical Finance*, 4(4), 289-304.

Ammann, M., & Seiz, R. (2004). Valuing employee stock options: Does the model matter? *Financial Analysts Journal*, 60(5), 21–37.

Areal, N., & Rodrigues, A. (2013). Fast trees for options with discrete dividends. *Journal of Derivatives*, 21(1), 49-63.

Barone-Adesi, G., & Whaley, R.E. (1987). Efficient analytic approximation of American option values. *Journal of Finance*, 42(2), 301-320.

Basso, A., Nardon, M., & Pianca, P. (2002). Discrete and continuous time approximations of the optimal exercise boundary of American options. *Quaderni del Dipartimento di Matematica Applicata*, 105.

Basso, A., Nardon, M., & Pianca, P. (2004). A two-step simulation procedure to analyze the exercise features of American options. *Decisions in Economics and Finance*, 27(1), 35-56.

Benninga, S. (2008). *Financial modelling* (3rd ed.): MIT press.

Bettis, J.C., Bizjak, J.M., & Lemmon, M.L. (2005). Exercise behavior, valuation, and the incentive effects of employee stock options. *Journal of Financial Economics*, 76 (2), 445–470.

Bjerksund, P., & Stensland, G. (1993). Closed-form approximation of American options. *Scandinavian Journal of Management*, 9, 87-99.

Bjerksund, P., & Stensland, G. (2002). Closed form valuation of American options. *Technical Report*. Norwegian School of Economics and Business Administration, Department of Finance and Management Science.

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

Boyle, P. P. (1986). Option valuation using a three-jump process. *International Options Journal*, 3, 7-12.

Boyle, P.P., & Lau, S.H. (1994). Bumping up against the barrier with the binomial method. *Journal of Derivatives*, 1(4), 6-14.

Brennan, M.J., & Schwartz, E.S. (1977). The valuation of American put options. *Journal of Finance*, 32(2), 449-462.

Brisley, N., & Anderson, C. K. (2008). Employee stock option valuation with an early exercise boundary. *Financial Analysts Journal*, 64(5), 88-100.

Broadie, M., & Detemple, J. (1996). American option valuation: new bounds, approximations, and a comparison of existing methods. *The Review of Financial Studies*, 9(4), 1211-1250.

Bunch, D.S., & Johnson, H. (1992). A simple and numerically efficient valuation method for American puts using a modified Geske–Johnson approach. *Journal of Finance*, 47(2), 809-816.

Byun, S. J., & Kim, I. J. (1994). Optimal exercise boundary in a binomial option pricing model. *Journal of Financial Engineering*, 3(2), 137-158.

Carpenter, J. N. (1998). The exercise and valuation of executive stock options. *Journal of Financial Economics*, 48(2), 127-158.

Carr, P. (1998). Randomization and the American put. *Review of Financial Studies*, 11(3), 597-626.

Chang, X., Fu, K., Low, A., & Zhang, W. (2015). Non-executive employee stock options and corporate innovation. *Journal of Financial Economics*, 115(1), 168-188.

Chen, T., & Joshi, M. (2012). Truncation and acceleration of the Tian tree for the pricing of American put options. *Quantitative Finance*, 12(11), 1695-1708.

Chendra, E., & Sidarto, K. A. (2020). An improved of Hull–White model for valuing employee stock options. *Review of Quantitative Finance and Accounting*, 54(2), 651-669.

Cox, J. C., Ross, S.A., & Rubinstein, M. (1979). Option pricing: a simplified approach. *Journal of Financial Economics*, 7(3), 229-263.

Curran, M. (1995). Accelerating American option pricing in lattices. *Journal of Derivatives*, 3(2), 8-18.

Cvitanic, J., Wiener, Z., & Zapatero, F. (2008). Analytic pricing of employee stock options. *Review of Financial Studies*, 21(2), 683-724.

Dempster, M. A. H., & Hutton, J. P. (1999). Pricing American stock options by linear programming. *Mathematical Finance*, 9(3), 229-254.

Derman, E., Kani, I., Ergener, D., & Bardhan, I. (1995). Enhanced numerical methods for options with barriers. *Financial Analysts Journal*, 51(6), 65-74.

Fabozzi, F.J., Paletta, T., Stanescu, S., & Tunaru, R. (2016). An improved method for pricing and hedging long dated American options. *European Journal of Operational Research*, 254(2), 656-666.

Figlewski S., & Gao, B. (1999). The adaptive mesh model: a new approach to efficient option pricing. *Journal of Financial Economics*, 53(3), 313-351.

Financial Accounting Standard Board (1995). Statement of financial accounting standard no. 123: Accounting for stock-based compensation.

Financial Accounting Standard Board (2004). Statement of financial accounting standard no. 123 (revised 2004): Share-based payments.

Financial Accounting Standard Board (2009). Accounting Standards Codification Topic 718: Compensation - Stock Compensation.

Financial Accounting Standard Board (2018). Accounting Standards Update No. 2018-07: Improvements to Nonemployee Share-Based Payment Accounting.

Finnerty, J. (2005). Extending the Black-Scholes-Merton model to value employee stock options. *Journal of Applied Finance*, 15(2), 25-54.

Finnerty, J. D. (2014). Modifying the Black-Scholes-Merton model to calculate the cost of employee stock options. *Managerial Finance*, 40(1), 2-32.

Frydman, C., & Saks, R. (2007). Executive compensation: a new view from a long-term perspective, 1936–2005. *The Review of Financial Studies*, 23(5), 2099-2138.

Gaudenzi, M., & Pressacco, F. (2003). An efficient binomial method for pricing American options. *Decisions in Economics and Finance*, 26(1), 1-17.

Geske, R., & Johnson, H. E. (1984). The American put option valued analytically. *Journal of Finance*, 39(5). 1511-1524.

Guo, S., & Liu, Q. (2019). A simple accurate binomial tree for pricing options on stocks with known dollar dividends. *Journal of Derivatives*, 26(4), 54-70.

Hall, B. J., & Murphy, K. J. (2002). Stock options for undiversified executives. *Journal of Accounting and Economics*, 33(1), 3-42.

Haug, E. (2007). Trees and finite difference methods. *The Complete Guide to Option Pricing Formulas* (2nd ed.): McGraw-Hill.

Hemmer, T., Matsunaga, S. & Shevlin, T. (1994) Estimating the 'fair value' of employee stock options with expected early exercise, *Account Horizons*, 8(4), 23-42.

Huang, J. Z., Subrahmanyam, M.G., & Yu, G.G. (1996). Pricing and hedging American options: a recursive integration method. *Review of Financial Studies*, 9(1), 277-300.

Huddart, S. (1994). Employee stock options. *Journal of Accounting and Economics*, 18(2), 207-231.

Hull, J., & White, A. (2004). How to value employee stock options. *Financial Analysts Journal*, 60(4), 114-119.

Hull, J.C. (2011). *Options, Futures, and Other Derivatives* (8th ed.): Prentice Hall.

International Accounting Standard Board (2004). International financial reporting standard no. 2: Share-based payments.

Joshi, M. S. (2009). The convergence of binomial trees for pricing the American put. *Journal of Risk*, 11(4), 87-108.

Ju, N. (1998). Pricing an American option by approximating its early exercise boundary as a multipiece exponential function. *Review of Financial Studies*, 11(3), 627-646.

Ju, N., & Zhong, R. (1999). An approximate formula for pricing American options. *Journal of Derivatives*, 7(2), 31-40.

Kim, I. J., & Byun, S. J. (1994). Optimal exercise boundary in a binomial option pricing model. *Journal of Financial Engineering*, 3(2), 137-158.

Kim, I. J., Jang, B. G., & Kim, K. T. (2013). A simple iterative method for the valuation of American options. *Quantitative Finance*, 13(6), 885-895.

Lam, S. S., & Chng, B. F. (2006). Do executive stock option grants have value implications for firm performance? *Review of Quantitative Finance and Accounting*, 26(3), 249-274.

Leisen, D. P., & Reimer, M. (1996). Binomial models for option valuation-examining and improving convergence. *Applied Mathematical Finance*, 3(4), 319-346.

Leung, T., & Sircar, R. (2009). Accounting for risk aversion, vesting, job termination risk and multiple exercises in valuation of employee stock options. *Mathematical Finance: An International Journal of Mathematics, Statistics and Financial Economics*, 19(1), 99-128.

Li, M. (2010). Analytical approximations for the critical stock prices of American options: a performance comparison. *Review of Derivatives Research*, 13(1), 75-99.

Marquardt, C. A. (2002). The cost of employee stock option grants: An empirical analysis. *Journal of Accounting Research*, 40(4), 1191-1217.

McDonald, R., & Schroder, M. (1998). A parity result for American option. *Journal of Computational Finance*, 1(3), 5-13.

McKean, H. P. (1965). Appendix: A free boundary problem for the heat equation arising from a problem in mathematical economics. *Industrial Management Review*, 6, 32-39.

Medvedev, A., & Scaillet, O. (2010). Pricing American options under stochastic volatility and stochastic interest rates. *Journal of Financial Economics*, 98(1), 145-159.

Merton, R. C. (1973). An intertemporal capital asset pricing model. *Journal of the Econometric Society*, 41(5), 867-887.

Ministry of Finance of the People's Republic of China (2006). Chinese accounting standards no. 11: Share-based payments.

Murphy, K. J. (2013). Executive compensation: Where we are, and how we got there. *Handbook of the Economics of Finance*: Elsevier.

Nielsen, B. F., Skavhaug, O., & Tveito, A. (2002). Penalty and front-fixing methods for the numerical solution of American option problems. *Journal of Computational Finance*, 5(4), 69-97.

Pressacco, F., Gaudenzi, M., Zanette, A., & Ziani, L. (2008). New insights on testing the efficiency of methods of pricing and hedging American options. *European Journal of Operational Research*, 185(1), 235-254.

Rouah, F. D., & Vainberg, G. (2007). *Option pricing models and volatility using Excel-VBA*: John Wiley & Sons.

Rubinstein, M. (1995). On the accounting valuation of employee stock options. Institute of Business and Economic Research, University of California, Berkeley.

Securities and Exchange Commission (2005). Staff accounting bulletin no. 107: Shared-Based Payment.

Securities and Exchange Commission (2007). Staff accounting bulletin no. 110: Shared-Based Payment.

Shang, Q., & Byrne, B. (2019). American option pricing: an accelerated lattice model with intelligent lattice search. *Journal of Derivatives*, 27(1), 92-108.

Staunton, M. (2005). Efficient estimates for valuing American options. *The Best of Wilmott*: John Wiley & Sons.

Tian, Y. (1993). A modified lattice approach to option pricing. *Journal of Futures Markets*, 13(5), 563-577.

Tian, Y. S. (1999). A flexible binomial option pricing model. *Journal of Futures Markets: Futures, Options, and Other Derivative Products*, 19(7), 817-843.

Wu, L. X., & Kwok, Y. K. (1997). A front-fixing finite difference method for the valuation of American options. *Journal of Financial Engineering*, 6(2), 83-97.

Zhu, S.P. (2006). An exact and explicit solution for the valuation of American put options. *Quantitative Finance*, 6(3), 229-242.

Appendix A

In this appendix, three Propositions and two Theorems of the optimal exercise boundary mentioned in Chapter 2 for an American put option are advanced. The Kim and Byun (1994) framework is augmented to incorporate unrestricted dividends y. Conditions for a continuous early exercise boundary are accordingly set out where the seed node is revealed in the penultimate column. Consider an American put option with an initial stock price S, strike price X, time to maturity T, risk-free interest rate r, continuous dividend yield y and volatility σ, priced by an n-step binomial model (we adhere to a standard notation). The magnitude of the stock price moving up and the stock price moving down are denoted by u and d. The probability of an upward and a downward movement respectively p and 1 - p are defined as:

$$u = exp\left(\sigma\sqrt{\Delta t}\right) \tag{A.1}$$

$$d = \exp\left(-\sigma\sqrt{\Delta t}\right) \tag{A.2}$$

$$p = (R/Y - d)/(u - d) \tag{A.3}$$

$$1 - p = (u - R/Y)/(u - d) \tag{A.4}$$

where $\Delta t = T/n$, $R = exp(r\Delta t)$, $Y = exp(y\Delta t)$. The stock price is given by:

$$S_{(i,j)} = Su^j d^{(i-j)} \tag{A.5}$$

Table 2.2 - 2.7 reported results from accelerated binomial option pricing process. This was largely accomplished by introducing an intelligent lattice search algorithm to efficiently locate the optimal exercise boundary. The optimal exercise node (n-1, B(n-1)) must be intuited by checking the exercise condition at *each* node along the penultimate column until the optimal boundary exercise is established. After detecting and confirming the optimal exercise node ❷ at column 9, in Figure 2.5, the seed value of the optimal exercise boundary is asserted. We then

proceeded inductively from the penultimate to the antepenultimate column and so on. To find a pathway through the lattice that delineates the stopping and continuation regions, a number of Propositions are framed from which two Theorems can be advanced and derived. In the derivation of the optimal exercise boundary robust to dividends, (n-1, B(n-1)) is regarded as a *known condition*. The node at which the first exercise value exceeds the holding value in the penultimate column can be numerically ascertained. This can be accomplished by simply iterating a lattice search algorithm specific to the penultimate column to locate the node at which the early exercise boundary emerges. To make the derivation more intuitive, the familiar nomenclature of a family is adapted here. Each node can be represented as a minute sub-tree of the entire binomial tree. In Figure A1, each node from the penultimate column back can be constituted as a "child" (i, j) with stock price $S_{(i, j)}$. The "mother" is defined here as (i + 1, j + 1) with stock price $uS_{(i, j)}$ and the "father" is (i + 1, j) with stock price $dS_{(i, j)}$. Nodes (i + 1, j + 1) and (i + 1, j) are "parents" in this tripartite "family" and these associations shift and are merely advanced to communicate lattice relativities and place-holding.[52] Consistent with this "family" nomenclature the "child" is the progeny of the "parents". We also try to add some intuition to the CRR tree by using a tilted j axis in Figure A2. This has the virtue of allowing the nodes to fall on the same diagonal when they share the same j (see the downward dashed line in Figure A2). Furthermore, the nodes on the same horizontal level have the same stock price (see the horizontal dashed line in Figure A2).

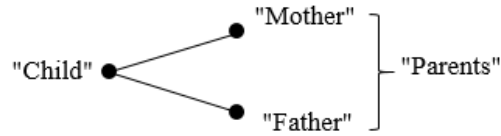[52] $uS_{(i, j)}$ implies multiplying $S_{(i, j)}$ by u and $dS_{(i, j)}$ implies multiplying $S_{(i, j)}$ by d.
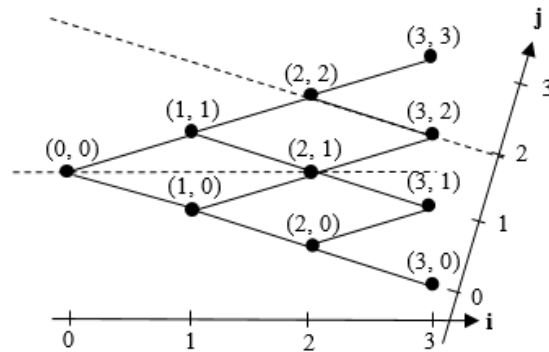
Figure A1: A "family" binomial tree

Figure A2: A binomial tree

Auxiliary Proposition. If $S_{(i, j)} > S_{(n - 1, B(n - 1) + 1)}$, then $(i, j) \in C$ for $1 \leq j \leq i \leq n - 1$ (the nodes above the horizontal solid line from the penultimate column back in Figure A3 belong to the continuation region).

As observed before, $(n - 1, B(n - 1))$, the optimal exercise node at the penultimate column, is a *known condition*.[53] By extension $(n - 1, B(n - 1) + 1)$ is the first continuation node from the bottom at the penultimate column that is *known* to be more valuable when the option is held rather than exercised. The $(n - 1, B(n - 1) + 1)$ node reference can be regarded as the "child" or apex node of $(n, B(n - 1) + 2)$ and $(n, B(n - 1) + 1)$. The proof of the Auxiliary Proposition can be divided into two parts relating to different properties of its "mother", $S_{((n, B(n - 1) + 2))}$. The value of put options at the maturity are unambiguously determined by:

---

[53] This can be easily established by testing each node until the optimal exercise node is found in the penultimate column. Is it more valuable to hold or to exercise?

$$V_{(n,j)} = \max(X - S_{(n,j)}, 0) \tag{A.6}$$

which means if $X \geq S_{(n, j)}$, then $V_{(n, j)} = (X - S_{(n, j)})$ and $(n, j) \in S$. Otherwise if $X < S_{(n, j)}$, $V_{(n, j)} = 0$ and $(n, j) \in C$.[54]

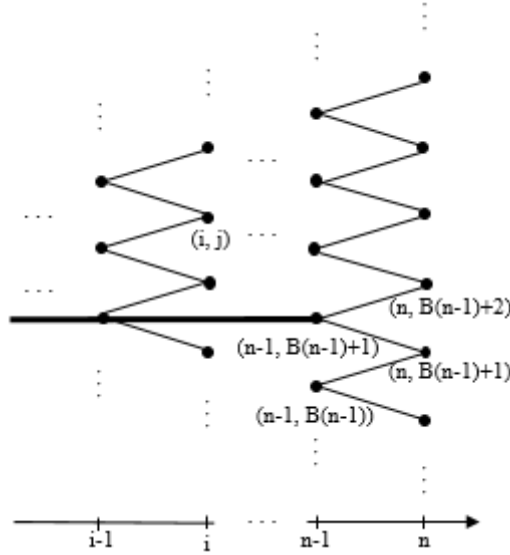Figure A3: Auxiliary proposition

Auxiliary Proposition (Part A): If $X < S_{(n, B(n - 1) + 2)}$ ("mother" $\in C$) and $S_{(i, j)} > S_{(n - 1, B(n - 1) + 1)}$, then $(i, j) \in C$ for $1 \leq j \leq i \leq n - 1$.

Proof of Auxiliary Proposition (Part A): Since $S_{(n, B(n - 1) + 2)}$ represents the smallest stock price $S_{(i, j)}$ which satisfies $S_{(i, j)} > S_{(n - 1, B(n - 1) + 1)}$ in the tree, $S_{(i, j)} > S_{(n - 1, B(n - 1) + 1)}$ is equivalent to $S_{(i, j)} \geq S_{(n, B(n - 1) + 2)}$. Combining with $X < S_{(n, B(n - 1) + 2)}$, $X < S_{(i, j)}$ can be obtained. Therefore, (i, j) belongs to the continuation region. The Auxiliary Proposition (Part A) can be proven.

Auxiliary Proposition (Part B): If $X \geq S_{(n, B(n - 1) + 2)}$ ("mother" $\in S$) and $S_{(i, j)} > S_{(n - 1, B(n - 1) + 1)}$, then $(i, j) \in C$ for $1 \leq j \leq i \leq n - 1$.

---

[54] **S** and **C** denote the stopping and continuation regions respectively.

Proof of Auxiliary Proposition (Part B): Since $X \geq S_{(n, B(n-1)+2)} > S_{(n, B(n-1)+1)}$, both "parents" ϵ S, which means the exercise values are assigned to "parents". Their "child" (n - 1, B(n - 1) + 1), as a continuation node which *has been already established (known)*, has the following property by Equation (2.1) - (2.3):

$$X - S_{(n-1,B(n-1)+1)} < [p(X - S_{(n,B(n-1)+2)}) + (1-p)(X - S_{(n,B(n-1)+1)})]R^{-1} \qquad \text{(A.7)}$$

which can be rewritten as:

$$X - S_{(n-1,B(n-1)+1)} < [p(X - uS_{(n-1,B(n-1)+1)}) + (1-p)(X - dS_{(n-1,B(n-1)+1)})]R^{-1} \quad \text{(A.8)}$$

According to Equation (A.1) - (A.4), $[pu + (1 - p)d]R^{-1} = Y^{-1}$ holds[55]. Therefore, Inequality (A.8) can be rewritten as:

$$S_{(n-1,B(n-1)+1)} > X(1 - R^{-1})/(1 - Y^{-1}) \qquad \text{(A.9)}$$

Inequality (A.9), *the known condition*, implies that as long as $S_{(i, j)} > S_{(n-1, B(n-1)+1)}$ for $1 \leq j \leq i \leq n - 1$ (the nodes above the horizontal solid line from the penultimate column back, see Figure A3), then Inequality (A.10) below holds:[56]

$$S_{(i,j)} > X(1 - R^{-1})/(1 - Y^{-1}) \qquad \text{(A.10)}$$

By reference to the reverse process of rearranging Inequality (A.9) - (A.7), Inequality (A.10) can be rewritten as:

$$X - S_{(i,j)} < [p(X - S_{(i+1,j+1)}) + (1-p)(X - S_{(i+1,j)})]R^{-1} \qquad \text{(A.11)}$$

---

[55] Since $Y = \exp(y^* \Delta t)$, $Y > 1$ and $Y^{-1} < 1$.
[56] The *known condition* has been established by testing the nodes in the penultimate column. Using a search algorithm, the location of the first continuation node at the penultimate column can be verified.

It follows from the definition of S and C that $V_{(i,j)} \geq X - S_{(i,j)}$ always holds. Therefore, from Inequality (A.11), so long as $S_{(i,j)} > S_{(n-1, B(n-1)+1)}$ for $1 \leq j \leq i \leq n-1$, Inequality (A.12) below can be advanced:

$$X - S_{(i,j)} < \left[pV_{(i+1,j+1)} + (1-p)V_{(i+1,j)} \right]R^{-1} \tag{A.12}$$

which implies $(i, j) \in C$. The proof of the Auxiliary Proposition (Part B) is therefore complete. Overall, the Auxiliary Proposition holds.

Proposition 1. If (i + 2, j + 1) ∈ C, then (i, j) ∈ C for 0 ≤ j ≤ i ≤ n - 2.
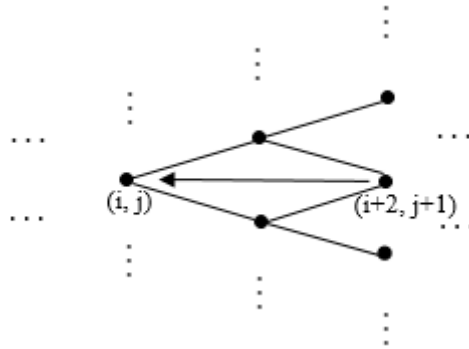
Figure A4: Proposition 1

Consistent with Kim and Byun (1994) Proposition 2, Proposition 1 can be asserted.

Proposition 2. If (i + 1, j + 1) ∈ S and (i + 1, j) ∈ S, then (i, j) ∈ S for 0 ≤ j ≤ i ≤ n - 2 (If the "parents" ∈ S then the "child" ∈ S from the penultimate column back).
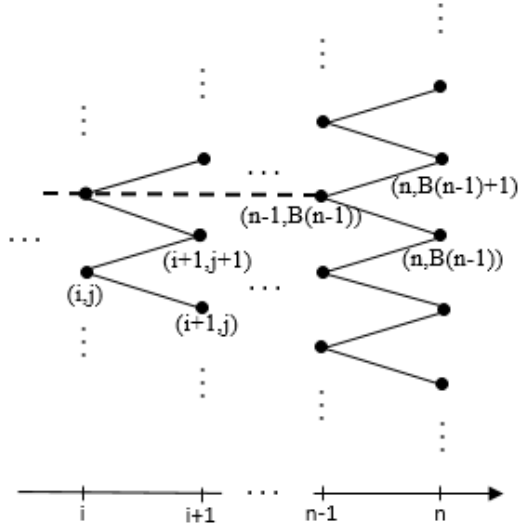
Figure A5: Proposition 2

Proof of Proposition 2: The optimal exercise node at the penultimate column is located (n - 1, B(n - 1)), which is a *known condition*. As a stopping node, (n - 1, B(n - 1)) adheres to the following property given Equation (2.1) - (2.3):

$$X - S_{(n-1,B(n-1))} \geq [pV_{(n,B(n-1)+1)} + (1-p)V_{(n,B(n-1))}]R^{-1} \qquad (A.13)$$

We posit here (n - 1, B(n - 1)) as a "child" and focus on the properties of its "parents". First, we prove that the "father" (n, B(n - 1)) ϵ S. By (1), $V_{(n - 1, B(n - 1))} = X - S_{(n - 1, B(n - 1))} \geq 0$. It is clear $S_{(n, B(n - 1))} < S_{(n - 1, B(n - 1))}$ and $X - S_{(n, B(n - 1))} > X - S_{(n - 1, B(n - 1))} \geq 0$. According to Equation (A.6), $V_{(n, B(n - 1))} = X - S_{(n, B(n - 1))}$ and (n, B(n - 1)) ϵ S. Two possibilities are feasible for the "mother" (n, B(n - 1) + 1): either (n, B(n - 1) + 1) ϵ S or (n, B(n - 1) + 1) ϵ C. If (n, B(n - 1) + 1) ϵ S, by Equation (A.6), $V_{(n, B(n - 1) + 1)} = X - S_{(n, B(n - 1) + 1)} \geq 0$. This implies that Inequality (A.13) can be rewritten as:

$$X - S_{(n-1,B(n-1))} \geq [p(X - S_{(n,B(n-1)+1)}) + (1-p)(X - S_{(n,B(n-1))})]R^{-1} \qquad (A.14)$$

Otherwise, if (n, B(n - 1) + 1) ϵ C, by Equation (A.6), X - $S_{(n, B(n - 1) + 1)}$ < 0 so that $V_{(n, B(n - 1) + 1)}$ = 0. Combining this with the properties linked to status of "father", Inequality (A.13) should be rewritten as:

$$X - S_{(n-1,B(n-1))} \geq [0 + (1 - p)(X - S_{(n,B(n-1))})]R^{-1} \tag{A.15}$$

This implies that even if (n, B(n - 1) + 1) ϵ C, Inequality (A.14) still holds. The optimal exercise node at the penultimate column (n - 1, B(n - 1) always satisfies Inequality (A.14), which is a *known condition*. By extension, rearranging inequalities pertaining to Inequality (A.7) - (A.9), permits Inequality (A.14) can be rewritten as:

$$S_{(n-1,B(n-1))} \leq X(1 - R^{-1})/(1 - Y^{-1}) \tag{A.16}$$

Inequality (A.16), can be equally postulated as a *known condition.* As long as $S_{(i, j)} \leq S_{(n - 1, B(n - 1))}$ for $0 \leq j \leq i \leq n - 2$ (the nodes on or below the horizontal dashed line from the penultimate column back, see Figure A5). Inequality (A.17) below holds:

$$S_{(i,j)} \leq X(1 - R^{-1})/(1 - Y^{-1}) \tag{A.17}$$

By reference to the reverse process of rearranging inequalities Inequality (A.9) – (A.7), Inequality (A.17) can be rewritten as:

$$X - S_{(i,j)} \geq [p(X - S_{(i+1,j+1)}) + (1 - p)(X - S_{(i+1,j)})]R^{-1} \tag{A.18}$$

Hitherto, if (i + 1, j + 1) ϵ S, (i + 1, j) ϵ S and $S_{(i, j)} \leq S_{(n - 1, B(n - 1))}$, then (i, j) ϵ S for $0 \leq j \leq i \leq n - 2$. The next step is to drop the assumption $S_{(i, j)} \leq S_{(n - 1, B(n - 1))}$, which is as follows:

According to Auxiliary Proposition and Proposition 1, since (n - 1, B(n - 1) + 1) is the first continuation node at the penultimate column, the nodes with stock price equal to or higher than $S_{(n - 1, B(n - 1) + 1)}$ should belong to the continuation region (the nodes on or above the horizontal heavy solid line in Figure A6). Then the highest possible stopping node should be on or below

the horizontal heavy dashed line. Since the "mother" ϵ *S* is assumed in Proposition 2, the highest stopping node "mother" in Proposition 2 can only possibly fall on or below the horizontal heavy dashed line. Its "child", apparently the highest "child" in proposition 2, can only possibly be on or below the horizontal light dashed line. Therefore, the assumption $S_{(i,\,j)}$ $\leq S_{(n-1,\,B(n-1))}$ (the "child" (i, j) with stock price equal to or lower than $S_{(n-1,\,B(n-1))}$) has already been satisfied when (i + 1, j + 1) ϵ *S* ("mother" ϵ *S*) is asserted in Proposition 2. After dropping the redundant assumption, the proof of Proposition 2 is complete.

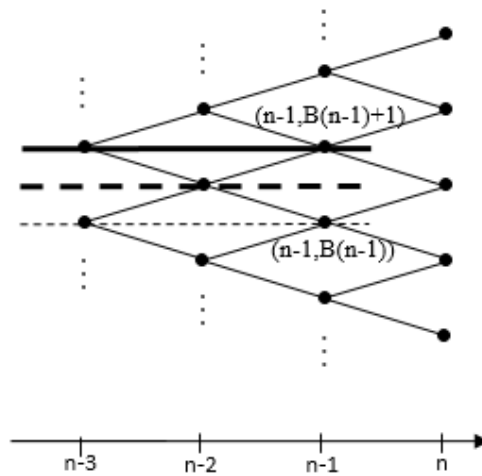Figure A6: Drop the redundant assumption

Proposition 3. If (i, j) ϵ C, then (i, j - 1) ϵ S and (i - 1, j - 1) ϵ S, and if (i, j) ϵ C, then (i, j + 1) ϵ

C and (i - 1, j) ϵ C for 1 $\leq i \leq n$ - 1, 1 $\leq j \leq i$ - 1. (If "mother" ϵ S then her "family" ϵ S and if the "father" ϵ C then his "family" ϵ C from the penultimate column back).
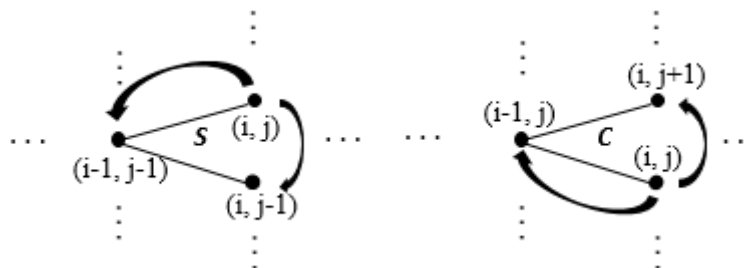

Figure A7: Proposition 3

122

Proof of Proposition 3: Proposition 3 can be proved using induction for $1 \leq i \leq n$ - 1. Firstly, from the Auxiliary Proposition and Proposition 2, the result obviously holds for i = n - 1. Secondly, suppose Proposition 3 holds for i = k when $1 \leq k \leq n$ - 1. We need to prove that Proposition 3 holds when i = k - 1. Propositions 1, 2 and 3 are consistent with Proposition 2, 1 and 3 from Kim and Byun (1994). One important difference however relates to the fact that our Propositions only apply from the penultimate column back. Since the proof for i = k - 1 in their Proposition 3 is wholly based on their Proposition 1 and 2, their proof can be equally used for our Proposition 3 based also on our Proposition 2 and 1.

Theorem 1. The optimal exercise boundary of an American put option with unrestricted continuous dividend yields is continuous from the penultimate column back. This implies that $B(i - 1) = B(i)$ or $B(i - 1) = B(i) - 1$ for $i, i - 1 \in \mathbf{I_0}$.

Theorem 2. The optimal exercise boundary of an American put option with unrestricted continuous dividend yields is non-increasing from the penultimate column back. This implies that if $B(i - 1) = B(i)$ then $B(i - 2) = B(i - 1) - 1$ for $i, i - 1, i - 2 \in \mathbf{I_0}$.

Proof of Theorem 1 and 2 follows the logic presented by Kim and Byun [1994]. Our Theorems 1, 2 and Propositions 1, 3 are consistent with their Theorem 1, 2 and Propositions 2, 3. The boundary from the penultimate column back is simply developed. Since the proof of Kim and Byun (1994) Theorem 1 and 2 is based on their Proposition 3 and Proposition 2 respectively, their proof can be used equally for Theorem 1 and 2 here based on our Proposition 3 and Proposition 1.

Appendix B

Convergence in the Hull White (2004) model is slow and follows an obstinate saw tooth erratic pattern. We found already that a Boyle Lau (1994) specification for the barrier option can produce faster convergence with lower RMSRE. In this Appendix, we add a few additional tweaks to make the lattice estimation more efficient. We follow in part the techniques outlined by Curran (1995) to define, demarcate and ultimately truncate regions within the binomial lattice. In particular, we identify a zero region and the barrier region of the Hull-White ESO lattice that can be used to leverage some efficiency. We also implement a technique suggested by Broadie and Detemple (1996) to optimise the mapping array. Most implementations of binomial trees including Benninga (2008) defined a two-dimensional lattice. Broadie and Detemple (1996) in their Appendices B.1 and B.2 stated however that it is not necessary to store the entire tree in memory. They stressed that only information related to the contemporaneous time-period/time-step is required marked out by the oval shapes in Figure B.1. It is possible to enhance the Hull-White-Boyle-Lau (HWBL) procedure by configuring the backward recursion such that at any one-time computer memory is only being used to estimate a single column of option time values. Consider the static binomial option tree below.
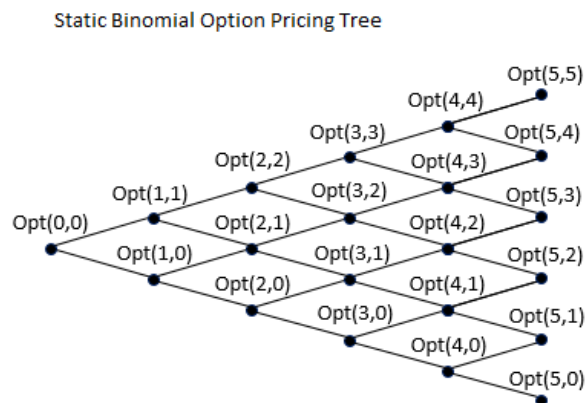

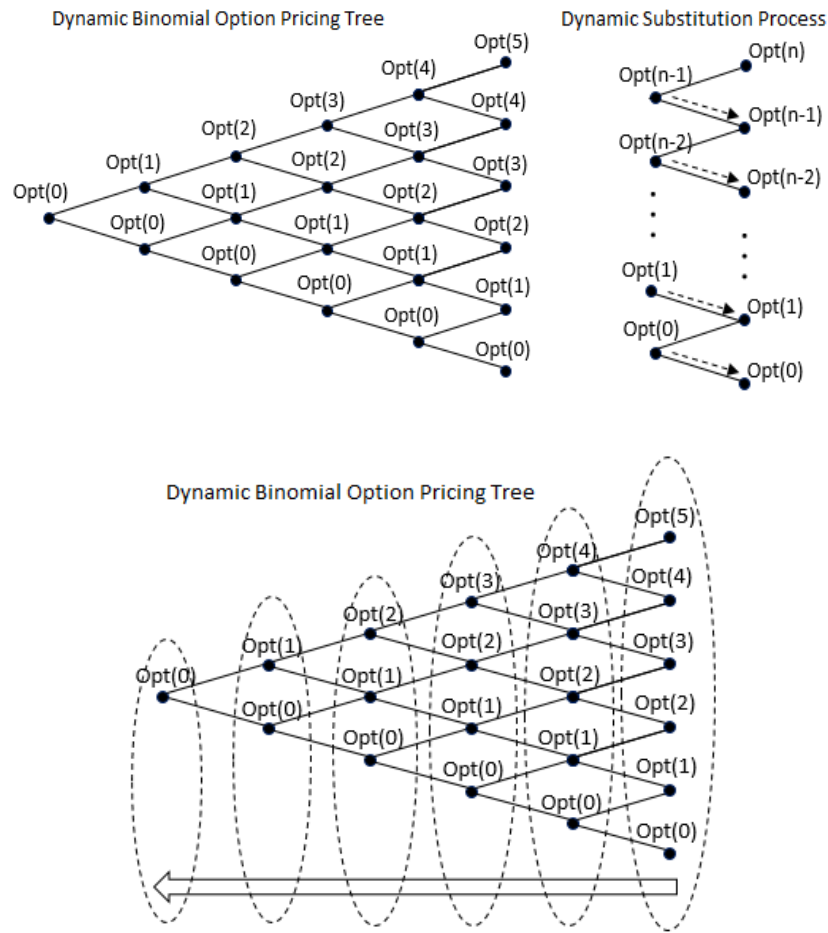Figure B.1: A two-dimensional static binomial option pricing tree

Figure B.2: A one-dimensional dynamic binomial option pricing tree and substitution process

The original HWBL model of Figure B.1 is a two-dimensional *Static Binomial Option Pricing Tree*, which allocates an independent storage space for the option time values of every node. The dynamic HWBL model in Figure B.2 is a one-dimensional *Dynamic Binomial Option Pricing Tree*, which exhibits a *Dynamic Substitution Process*. In the option pricing process of backward induction, the option values we calculate at any time step would substitute the option values at its *previous* time steps to be stored in the same space.[57] Consider a n-step binomial tree. The static binomial tree requires $(n + 1)(n + 2)/2$ storage spaces. The dynamic binomial

---

[57] Each step time period can be identified above by the passing from one ovoid to the previous.

tree only needs $n + 1$ storage spaces which reduces by decrements of one with each successive backward recursion. This optimization effectively reduces the unnecessary data storage.

A second type of optimization follows in line with insights provided by Curran (1995) where the lattice is effectively truncated. Regions that are unnecessary can be removed from the backward recursion estimation. In Figure B.3 below, we identify four regions for the Hull White (2004) lattice.
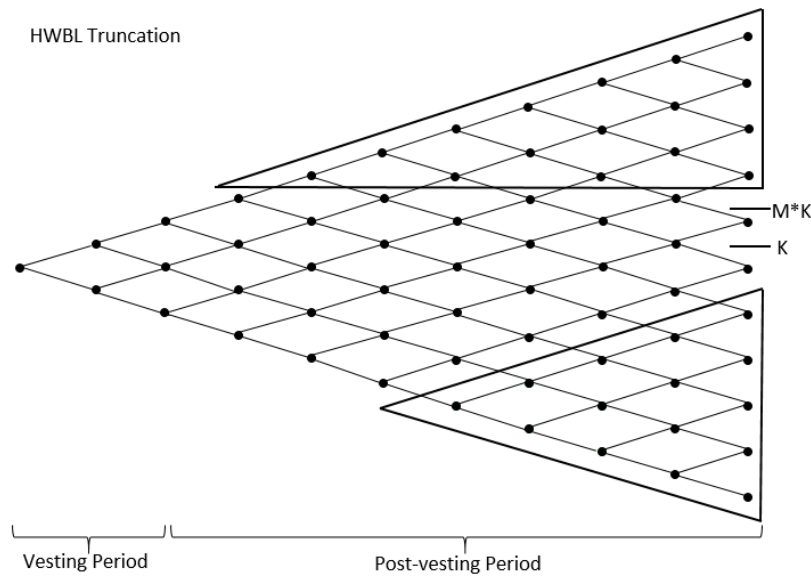

Figure B.3: Truncation technology for Hull-White model

We truncate the "Redundant early exercise" region (upper triangle) in which option values of nodes are equal to their intrinsic values and the "zero" region (lower triangle) in which the option value of each node is zero. The valuation in these two regions are unnecessary since they will not influence materially the backward induction or values passing back through to produce the present value of the option. In Figure B.3, we identify four different valuation regions. This similarly follows the logic proposed by Curran (1995). Curran (1995) proposed the Diagonal Method which draws on the Kim and Byun (1994) deduction that the early boundary of an American option can be identified in the binomial structure. Pinpointing exactly

126

the early exercise boundary speeds up estimation time because it removes the necessity to constantly test each node. The **stopping** region and **continuation** region can be defined around the boundary.[58] Curran notes that once the option is exercised, it cannot be exercised again. Importantly, here with Hull White (2004), we know the early exercise because this is made explicit in the model. Hull and White (2004) posited that early exercise occurs once the stock price passes through the **M*K** boundary where **M** is defined as the multiple parameter. This simplifies the analysis and allows the lattice to be segmented. The option value of the nodes in the upper triangle is captured by the intrinsic value defined by **M*K**. The option values of the nodes in the lower triangle are all zero. This leaves the rest of nodes to be valued by backward induction. The nodes after the vesting period still possess the passive early exercise condition applying by virtue of employees terminating employment. The vesting period does not permit early exercise: only forfeiture.

In the Table below, we provide some insight into how these optimisations speed up computation times in Excel™ for both the Dynamic and Truncation implementation relative to the original HWBL model.

---

[58] We borrow these terms form Kim and Byun (1994).

Steps	HWBL	Accelerated HWBL
500	27.8675	27.8675
	01:56.00	00:00.32
1000	27.8530	27.8530
	00:03.36	00:00.18
1500	27.8549	27.8549
	00:06.61	00:01.59
2000	27.8576	27.8576
	00:11.63	00:02.75
2500	27.8565	27.8565
	00:19.80	00:04.45
3000	27.8562	27.8562
	00:27.83	00:06.13
4000	27.8566	27.8566
	00:47.08	00:10.16
5000	27.8535	27.8535
	01:15.22	00:15.67

Table B.1: Comparing original HWBL and accelerated HWBL[59]

We take again parameters values for the ESO we used before. We set S = 100, K = 100, T = 10 years, Sigma = 0.2, r = 0.06, Vesting = 2 years, M = 1.5 and e = 0.04. The computation time generated in Excel™ at different number of steps is shown using the format mm:ss.00. The laptop we used for computation was a DELL Latitude E5470 with Intel's Core i3 processors. As can be seen from Table B.1, the accelerated HWBL (HWBL with Dynamic and Truncation implementation) is approximately four to five times faster than the original HWBL. The main gain in terms convergence is obtained by imposing Boyle Lau. Nevertheless, a significant reduction in runtime can be engineered by optimizing the code.

---

[59] Speed Comparison: Excel™ VBA Estimation

Publication and Conferences

Publication

Shang, Qianru, and Brian Byrne. "American Option Pricing: An Accelerated Lattice Model with Intelligent Lattice Search." *The Journal of Derivatives*, Vol 27, No. 1 (2019), pp. 92-108.

Shang, Qianru, and Brian Byrne. "American Option Pricing: Optimal Lattice Models and Multidimensional Efficiency Tests." *European Journal of Operational Research*, under reviewed.

Conferences

"An Efficient Search Routine for Uncovering the Early Exercise Boundary in American Option Lattice Structures." *INFINITI Conference on International Finance*, 2019

"Accounting for Employee Stock Options: Accelerating Convergence." 10th World Congress of *the Bachelier Finance Society*, 2018

"Accounting for Employee Stock Options: A Small Refinement to the Lattice Approach." *INFINITI Conference on International Finance*, 2017

"Valuation of Employee Stock Options." 8th Annual Postgraduate Research Symposium of Technological University Dublin, 2017

Employability Skills and Discipline Specific Skills Training

Employability Skills Modules

Introduction to Pedagogy for Postgraduates

Discipline Specific Modules

Derivatives 1 & 2 & 3

Financial Mathematics 1 & 2

Econometrics 1 & 2

Business Research Methods