Some innovative numerical approaches for pricing American options

Jin Zhang
University of Wollongong


This paper is posted at Research Online.
NOTE

This online version of the thesis may have different page formatting and pagination from the paper copy held in the University of Wollongong Library.

UNIVERSITY OF WOLLONGONG

COPYRIGHT WARNING

You may print or download ONE copy of this document for the purpose of your own research or study. The University does not authorise you to copy, communicate or otherwise make available electronically to any other person any copyright material contained on this site. You are reminded of the following:

Copyright owners are entitled to take legal action against persons who infringe their copyright. A reproduction of material that is protected by copyright may be a copyright infringement. A court may impose penalties and award damages in relation to offences and infringements relating to copyright material. Higher penalties may apply, and higher damages may be awarded, for offences and infringements involving the conversion of material into digital or electronic form.
Some innovative numerical approaches for pricing American options

A thesis submitted in (partial) fulfillment of the requirements for the award of the degree of

Master of Science

from

UNIVERSITY OF WOLLONGONG

by

Jin Zhang

Master of Financial Mathematics
University of Wollongong

Bachelor of Engineering
Beijing University of Posts and Telecommunications

2007
I, Jin Zhang, declare that this Thesis, submitted in fulfilment of the requirements for
the award of Master of Science, in the School of Mathematics and Applied Statistics,
University of Wollongong. This Thesis is my own work unless otherwise referenced.
The document has not been submitted for a higher degree to any other University
or Institution.

Jin Zhang

March, 2007
ACKNOWLEDGEMENTS

I gratefully acknowledge the people who provided assistance in preparing this Thesis. First of all, I would like to express my deep gratitude to my supervisor, Dr. Song-ping Zhu, without his advice and assistance, this Thesis would have never been completed. I would also like to thank all staff in the School of Mathematics and Applied Statistics, especially Carolyn Silveri for her help in Latex, Dr. Xiao-ping Lu for her help in the Laplace Transform part, Dr. Joanna Goard for her constant encouragement, my dear neighbor Dr. Keith Tognetti and his faithful fellow Jack for the every night we shared in the university. Last but not least, I must thank my mom for her support and encouragement; without her, it is impossible for me to come and study in Australia, this Thesis is dedicated to her.
ABSTRACT

With the well-known model of lognormal asset price, the option valuation problems can be implemented by using the Black-Scholes partial differential equation approach. However, for American option pricing problems, it is hard to find an analytical formula due to the moving boundary feature [23]. This thesis presents two innovative numerical methods [38, 39] to value American put options in terms of solving the Black-Scholes partial differential equation with a set of appropriate boundary conditions.

The first method is the Laplace Transform Method, which extends the pseudo-steady-state approximation idea for the American option pricing problems in non-dividend yield case [35] to the one in constant dividend yield case. The approach transfers the original partial differential equations system to an ordinary differential equations system, to derive the solutions of the option prices and the optimal exercise boundary in the Laplace space respectively. After that, numerical inversions are performed to restore their corresponding values in the original time space.

The second method promotes a new predictor-corrector idea that uses a hybrid finite difference scheme to tackle the nonlinear nature of American option pricing problems, which is explicitly exposed after applying the front-fixing technique [21] to the original Black-Scholes partial differential equation. The new predictor-corrector scheme implements the computation of the option prices and the optimal exercise boundary through solving a set of linearized difference equations at each time step, to achieve high computational efficiency and numerical accuracy.

Through the comparison with Zhu’s analytical solution [34], we found that, the Laplace Transform Method is highly efficient since numerical calculations are only
performed for the inversion part, whereas the calculations of the Laplace transform are done analytically. Although the Laplace Transform Method slightly undervalues the optimal exercise boundary due to the pseudo-steady-state approximation introduced to allow the Laplace transform to be performed on the moving boundary. The loss of the accuracy in this regard is greatly compensated by its high computational speed. For the second method, we have shown that the numerical results obtained from the predictor-corrector scheme converge uniformly to Zhu’s exact optimal exercise boundary and option values [34], provided a convergence criterion is imposed. Furthermore, the agreement between the numerical solutions from the second method, and those from the Grid Stretching Method [24] that is a fourth-order scheme for both the asset price and time discretizations, not only validates the second method once again but also demonstrates its accuracy in that a lower-order scheme has virtually achieved the same level of accuracy as a higher-order scheme does.
CONTENTS

1. Introduction .................................................. 1

2. Arbitrage-Free Pricing Model ................................. 5
   2.1 Stochastic Processes ................................. 6
   2.2 Itô’s Formula ........................................... 7
   2.3 The Black-Scholes Equation .......................... 9
   2.4 The Partial Differential Equation System ............ 11

3. The Laplace Transform Method .............................. 15
   3.1 Solutions of Optimal Exercise Price and Option Value ...... 16
   3.2 Numerical Laplace Inversion .......................... 21
      3.2.1 Stehfest Method ................................... 22
      3.2.2 Papoulis Legendre Polynomial Method ............ 23
      3.2.3 Kwok and Barthez’s Linear Combination Method .... 24
   3.3 Numerical Test of Standard Functions .................. 25
   3.4 Numerical Examples for American Puts .................. 31
      3.4.1 Numerical Accuracy .............................. 31
      3.4.2 Numerical Efficiency ............................ 33
      3.4.3 Accuracy of the Laplace Transform Method ...... 35
      3.4.4 Calculation of the Delta ......................... 36

4. The Predictor-Corrector Scheme ............................ 41
   4.1 The Front-Fixing Transform ........................... 43
   4.2 The Predictor-Corrector FDM Scheme .................. 45
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.3</td>
<td>Numerical Examples</td>
<td>50</td>
</tr>
<tr>
<td></td>
<td>4.3.1 Discussion on Validity</td>
<td>51</td>
</tr>
<tr>
<td></td>
<td>4.3.2 Discussion on Order of Convergence</td>
<td>53</td>
</tr>
<tr>
<td></td>
<td>4.3.3 Discussion on Accuracy and Efficiency</td>
<td>55</td>
</tr>
<tr>
<td></td>
<td>4.3.4 The Constant Dividend Yield Case</td>
<td>59</td>
</tr>
<tr>
<td>5.</td>
<td>Conclusion</td>
<td>63</td>
</tr>
<tr>
<td>6.</td>
<td>Appendix</td>
<td>65</td>
</tr>
</tbody>
</table>
LIST OF FIGURES

3.1 The Optimal Exercise Prices by Stehfest Method with $D_0 = 0$ . . . . 32
3.2 Comparison of Analytic Solution and Numerical Inversion . . . . . . 36
3.3 The Delta Values with $D_0 = 0\%$ . . . . . . . . . . . . . . . . . . . . 39
3.4 The Delta Values with $D_0 = 5\%$ . . . . . . . . . . . . . . . . . . . . 40

4.1 The Predictor-Corrector Procedure . . . . . . . . . . . . . . . . . . 49
4.2 Comparison of Analytic Solution and Numerical Solution . . . . . . 52
4.3 The Numerical Optimal Exercise Boundary in Perpetual Case . . . . 54
4.4 The Relative RMS, $N$ and $M$ . . . . . . . . . . . . . . . . . . . . . 57
4.5 The Accuracy and Efficiency of the New Scheme . . . . . . . . . . . 58
4.6 The Option Value with $D_0 = 5\%, T = 1$ year . . . . . . . . . . . 60
4.7 The Delta Value with $D_0 = 5\%, T = 1$ year . . . . . . . . . . . 61
4.8 The Optimal Exercise Price with $D_0 = 5\%$ . . . . . . . . . . . . 62
LIST OF TABLES

3.1 Results of the Stehfest Method ........................................... 28
3.2 Results of the Papoulis Method ............................................ 29
3.3 Results of the Linear Combination Method ............................ 30
3.4 Numerical Inversion of the Stehfest Method for Option Pricing . . . 33
3.5 Numerical Inversion of the Linear Combination Method at $\tau = 0.045$ 33
3.6 Comparison of Efficiency ..................................................... 35

4.1 Order of Convergence in the $\tau$ Direction ............................. 55
4.2 Order of Convergence in the $x$ Direction ............................... 55
1. INTRODUCTION

Securities are paper assets which are issued by a government or company in order to acquire capital financing; examples of securities include bonds, bills of exchange, promissory notes, certificates of deposit, shares and financial derivatives. Options are the most common derivative securities that are frequently bought and sold in today’s open exchanges. Nowadays, investment companies use options for their risk management through hedging against possible fluctuations of the underlying asset price. Hence, the valuation of options is an important field in financial research.

The most basic options are call options and put options, which are often referred to as ‘vanilla’ options. A call option gives the holder the right to buy the underlying asset, whereas a put option gives the holder the right to sell the underlying asset. These buying and selling activities are usually known as the exercise of the option. For example, the exercise of a call option refers to the process that the option holder buys the underlying asset at a prescribe price from the option writer, who issued the option, and took the premium for the risk of incurring possible loss. Also, options can be classified as American options and European options according to their exercise feature. An American option allows its holder to exercise the option at any time prior to the expiry day. On the other hand, if the exercise of an option is only allowed at the expiry day, then it is called a European option. The option holder can forgo the exercise right since the option is not an obligation for its holder. However, if the option holder decides to exercise the option, the option writer is obligated to deliver the underlying asset. In this thesis, we focus the study on American put option pricing problems.

The revolution on financial derivative securities, both in exchange markets and in
academic communities, began in the early 1970’s. How to rationally price an option was not clear until 1973, when a famous formula was given by Black and Scholes [4]. However, the Black-Scholes formula is only suitable for the pricing problems of European-style options. For American-style options, the Black and Scholes formula can not be applied due to the moving boundary feature [23]. In the past two decades, many researchers used different methods to tackle the moving boundary problem of American option pricing problems. For a long time, there were two kinds of approximation methods for American options pricing problems. One approach is the numerical method, such as the Finite Difference Method (FDM) [31], the Finite Element Method [1] and the Multi-period Binomial Method [11]. The other one is the analytical approximation method, like the Compound-Option Approximation Method [14], the Integral Representation Method [19] and the Laplace Transform Method [9, 22, 35]. Despite the recent progress made in finding an exact analytical solution by Zhu [34] in non-dividend yield cases, numerical methods are still preferred by market practitioners as they are usually efficient, easy to implement and have acceptable accuracy. However, among these numerical methods, most of them can not carry out an accurate solution without sacrificing the computational efficiency. For example, the Binomial Method [11] is a time-consuming method [5, 16], which generates more accurate results at the expense of greater computational time; the parameter estimation method [10] and some finite difference methods [3, 30] need iterations for more accurate results at each time step.

This thesis presents two numerical methods to value American put options on a constant dividend yield paying asset. We adopt the partial differential equation (PDE) that was initially derived by Merton [23], who extended the original PDE of Black and Scholes [4] to the one for the constant dividend yield case. We use the PDE and a set of appropriate boundary conditions to construct a partial differential equations system, which governs the American put option pricing problems. Then we solve the system by employing the following two methods.

The first method is called the Laplace Transform Method. Zhu promoted this
approach to value American put options on no dividend payment underlying asset [35], and he believe that the approach can be extended to price American puts for the constant dividend yield case as well. After the appropriate expressions of the optimal exercise boundary and the option prices are found in the Laplace space based on the pseudo-steady-state approximation [35] introduced to allow the Laplace transform to be performed on the moving boundary, numerical inversions are performed to restore their corresponding values in the original time space. Among many numerical inversion techniques, we have found that three are most suitable for the functions arising from option pricing problems. Then, out of these three methods, we found that, through numerical experiments, the Stehfest method is the best, in terms of both numerical accuracy and computational efficiency. A great advantage of this numerical approach is its robustness in calculating the Greeks of an option.

In the literature, there are many techniques to deal with the moving boundary problem. It has been noted [18] that the moving boundary problem can be regarded as a fixed boundary problem for a nonhomogeneous Black-Scholes equation. Being further motivated with this observation, we still use the front-fixing technique [21] to explicitly reveal the nonlinear nature of American options. After that, we promote a new predictor-corrector hybrid scheme, to tackle the nonlinearity of the Black-Scholes PDE, which is explicitly exposed after applying the front-fixing technique [21]. The new scheme is a one-step discretization in time, which advances the results of the next time step with using the information of the current time step. An advantage of the new predictor-corrector finite difference scheme, is its high efficiency due to no initialization nor iterations required by the scheme. Another advantage of the approach is its high numerical accuracy, due to reducing the numerical errors induced in the prediction-correction process.

This thesis is organized as follows. Chapter 2 introduces the governing partial differential equation and the boundary conditions, which are used to construct the partial differential equations system for American puts pricing problems. In Chapter 3, the Laplace Transform Method is used to develop the expressions of the optimal
exercise prices and the option values in the Laplace space; a most suitable numerical inversion method is selected for the functions arising from option pricing problems. Chapter 4 presents the new predictor-corrector finite difference scheme to tackle the nonlinear nature of American option pricing problems, which is explicitly exposed after applying the front-fixing technique [21] to the original Black-Scholes PDE. Our conclusion is given in Chapter 5.
2. ARBITRAGE-FREE PRICING MODEL

In 1973, Black and Scholes [4] presented the idea that a price of an option should be mainly depended on the price of its underlying asset. However, market practitioners hardly come to an agreement among on the option prices written on the underlying assets, since different market practitioner might have different opinions of the future value on the same underlying asset. However, Black and Scholes [4] promoted their pricing framework that guarantees only one fair price for an option in the market, if the market complies with some assumptions. The most important assumption in Black and Scholes [4] option pricing theory is no riskless arbitrage opportunities in the market. In other words, no player in the market is able to make a riskless profit through selling and buying financial securities. This key assumption assures the fair price of an option in the market, otherwise the arbitrage-free principle will be violated.

The arbitrage-free principle is very important in applied finance. The market practitioners apply this principle not only to price a single security but also to price a portfolio, which is a collection of different financial securities. A portfolio may contain long and short positions on different securities. A long position means that a liquidation of the securities creates a positive cash flow. On the other hand, a short position creates a potential negative cash flow on the liquidation day. In this chapter, we briefly review the procedure of deriving the celebrated Black-Scholes PDE, through applying the arbitrage-free principle to a portfolio, which is constructed by long position in option and short position in its underlying asset.

The procedure of deriving the extended Black-Scholes-Merton PDE [23], for the option on a constant dividend paying asset, is presented in this chapter as well. Fur-
2. Arbitrage-Free Pricing Model

Moreover, for American put option pricing problems, a partial differential equations system that is constructed by the PDE and a set of boundary conditions, is also presented at the end of the chapter.

2.1 Stochastic Processes

Before we start the option valuation problems, we should know how is the growth of underlying asset price since the value of an option mainly relates to the current value and future value of its underlying asset price. In the literature, the underlying asset price is assumed to follow the lognormal random walk \([30]\), and the mean and the variance of the asset return distribution are \(\mu t\) and \(\sigma^2 t\) respectively. Usually \(\mu\) refers to the drift rate of the underlying asset price, and \(\sigma\) refers to the volatility of the underlying asset with \(t\) being the time variable. Let us denote \(S_0\) as the current asset price and \(S_t\) as the future asset price at the time instant \(t\). The return of the asset over the time interval \(t\) is

\[
\frac{S_t - S_0}{S_0} = \mu t + \sigma \sqrt{t} \phi, \tag{2.1}
\]

in which \(\phi\) is a standard normal random variable with the mean being 0 and the variance being 1. The classical Black-Scholes pricing theory assumes both \(\mu\) and \(\sigma\) being the known functions of \(t\) or constants. Eq. (2.1) tells us that as time passes by an amount of \(t\), the expected return of the asset be \(\mu t\), and the deviation of the return be \(\sigma \sqrt{t} \phi\). Because the random term \(\phi\) varies during every small time interval \(t\), there is a sequence of \(\phi\) involved over the time periods. The sequence of the random variables is called random process or random walk.

Now if we take the limit of time interval \(t \to 0\), Eq. (2.1) becomes the stochastic differential equation

\[
dS = S\mu dt + S\sigma dW_t. \tag{2.2}
\]

The term \(dW_t\) contains the randomness that is known as a Wiener process, or
Brownian motion. Also, this term can be written as $dW_t = \phi \sqrt{dt}$, which has the following properties:

1. $dW_t$ is a random variable from the normal distribution;
2. The mean and variance of $dW_t$ are zero and $dt$ respectively.

If we set the volatility of the underlying asset $\sigma$ to 0, Eq. (2.2) turns to

$$
\frac{dS}{S} = \mu dt, \quad \text{or} \quad \frac{dS}{dt} = \mu S.
$$

(2.3)

Eq. (2.3) can be solved subject to $S = S_0$ when $t = 0$ to find $S = S_0 e^{\mu t}$, with $\mu$ being a constant. It means that the logarithm of $S$ follows the normal distribution, or the $S$ follows the lognormal distribution.

## 2.2 Itô’s Formula

Suppose $V(x)$ is a smooth function of $x$. Then using a Taylor series expansion,

$$
\Delta V = V'(x) \Delta x + \frac{V''(x)}{2!} \Delta x^2 + \frac{V'''(x)}{3!} \Delta x^3 + \cdots.
$$

(2.4)

In the limits as $\Delta x \to 0$, we have

$$
dV = V'(x)dx.
$$

(2.5)

Now suppose $V$ is a function of $S$, where $dS = S \mu dt + S \sigma \phi \sqrt{dt}$, with $\phi$ following standard normal distribution. Then

$$
\Delta V = V'(S) \Delta S + \frac{V''(S)}{2!} \Delta S^2 + \frac{V'''(S)}{3!} \Delta S^3 + \cdots.
$$

(2.6)

We can discretise the random walk followed by $S$, so that

$$
\Delta S = \sigma S \phi \sqrt{\Delta t} + \mu S \Delta t,
$$

(2.7)
\[ \Delta S^2 = (S \mu \Delta t + S \sigma \phi \sqrt{\Delta t})^2 = \sigma^2 S^2 \phi^2 \Delta t + o(\Delta t). \]  
(2.8)

As \( \Delta t \to 0 \) with the probability being 1, i.e., \( \phi = 1 \), \( \phi^2 \Delta t \to \Delta t \), so as \( \Delta t \to 0 \), the higher terms in \( \Delta t \), \( o(\Delta t) \to 0 \). Eq. (2.8) becomes

\[ dS^2 \to \sigma^2 S^2 dt. \]  
(2.9)

Substituting into our \( \Delta V \) Eq. (2.6) and taking the limits as \( \Delta t \to 0 \), we have Itô’s Formula for \( V \) a function of \( S \),

\[ dV = \sigma S \frac{dV}{dS} dW_t + (\mu S \frac{dV}{dS} + \frac{1}{2} \sigma^2 S^2 \frac{d^2V}{dS^2} + \frac{dV}{dt}) dt, \]  
(2.10)

which describes the random walk followed by a function \( V \) of \( S \). This can be generalized to give Itô’s Lemma for \( V \), \( V(S, t) \) is a smooth function depending on the underlying asset price \( S \) and the time \( t \). The Itô’s Lemma is

\[ dV = \frac{\partial V}{\partial t} dt + \frac{\partial V}{\partial S} dS + \frac{1}{2} \sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} dt, \quad \text{or} \]
\[ dV = \sigma S \frac{\partial V}{\partial S} dW_t + (\mu S \frac{\partial V}{\partial S} + \frac{1}{2} \sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + \frac{\partial V}{\partial t}) dt. \]  
(2.11)

If we consider \( V \) as the value of an option, the stochastic variable \( dW_t \) in Eq. (2.11) means that the option values also move randomly.

In the previous subsection, we mentioned that the randomness \( dW_t \) can be eliminated through having such a portfolio, which is constructed by the option and its underlying asset. If the randomness can be successfully eliminated in Eq. (2.11), the return of the portfolio is equivalent to a riskless return such as the interest rate. In the following section, we shall present the procedure of eliminating the random component in Eq. (2.11). This work leads to the famous Black-Scholes PDE, which only holds in the absence of arbitrage opportunities.
2.3 The Black-Scholes Equation

With the Black-Scholes [4] option pricing theory, a fair option price can be derived if the market complies with some assumptions. Black and Scholes [4] summarized these assumptions as follows.

1. The efficient market hypothesis is assumed to be satisfied. In other words, the market is assumed to be liquid, has fair price and provides all players with equal access to available information.
2. The assets are perfectly divisible.
3. The underlying asset price follows the lognormal random walk, or a geometric Brownian (Wiener) process.
4. The riskless interest rate \( r \) is a known function of time over the option life time.
5. The market does not have riskless arbitrage opportunities.
6. There are no transaction cost and tax in buying or selling the underlying asset or the option. Trading of the underlying asset and options can take place continuously.

According to the third assumption, the option value \( V \) is a stochastic process which mainly relates to another stochastic process \( S \) that is the underlying asset price. Now let us construct a portfolio consisting of long one option with value \( V \) and a short position of \( \Delta \) units on the stock, with the \( \Delta \) being unspecified yet. The value of the portfolio is

\[
\Pi = V - \Delta S. \tag{2.12}
\]

After one time step \( dt \), the value of the portfolio changes by

\[
d\Pi = dV - \Delta dS. \tag{2.13}
\]

After substituting Eq. (2.11) to Eq. (2.13), we obtain the PDE that describes the yield of the portfolio over the time step \( dt \),

\[
d\Pi = \frac{\partial V}{\partial t} dt + (\frac{\partial V}{\partial S} - \Delta) dS + \frac{1}{2} \sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} dt. \tag{2.14}
\]
In Eq. (2.14), obviously the random component \( dS \) can be eliminated through setting \( \Delta = \partial V / \partial S \). Once the random term \( dS \) is eliminated, \( d\Pi \) becomes fully deterministic,

\[
d\Pi = \left( \frac{\partial V}{\partial t} + \frac{1}{2} \sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} \right) dt. \tag{2.15}
\]

With imposing the arbitrage-free principle, the yield on the portfolio must be equal to the one of a riskless security over the same time period. Assume that the return of a riskless security is \( r \), then the portfolio value \( \Pi \) yields \( r\Pi dt \) after an infinitesimal time step \( dt \). The amount \( r\Pi dt \) must equal to the yield of the riskless portfolio \( d\Pi \) in Eq. (2.15) over the same time step \( dt \), thus we can have the expression

\[
r\Pi dt = \left( \frac{\partial V}{\partial t} + \frac{1}{2} \sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} \right) dt. \tag{2.16}
\]

With substituting Eq. (2.12) to Eq. (2.16) and dividing throughout by \( dt \), now we have the celebrated Black-Scholes PDE,

\[
\frac{\partial V}{\partial t} + \frac{1}{2} \sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + rS \frac{\partial V}{\partial S} - rV = 0. \tag{2.17}
\]

As Eq. (2.17) shows, the drift rate \( \mu \) does not present in the partial differential equation anymore. The arbitrage-free principle dictates that the drift rate of the underlying asset be exactly the return of the risk-free securities.

Now let us consider the case that the underlying asset pays a continuous dividend at a fixed rate \( D_0 \). After an infinitesimal time step \( dt \), the holder of the asset gains \( D_0 S dt \) due to the dividend payment. With complying the arbitrage-free principle, the underlying asset price must fall by the same amount. As the result, Eq. (2.2) becomes

\[
dS = S\mu dt + S\sigma dW_t - D_0 S dt. \tag{2.18}
\]

Doubtless, the dividend payment will affect the value of portfolio Eq. (2.12). The yield of the portfolio over the time step \( dt \), must drop by the amount being the decrease of the underlying asset price over the same time step, so Eq. (2.13)
becomes
\[ d\Pi = dV - \Delta dS - D_0 \Delta S dt. \]  \hfill (2.19)

After implementing the same process as deriving Eq. (2.17), the original Black-Scholes PDE becomes the one considering the constant dividend yield \( D_0 \),
\[
\frac{\partial V}{\partial t} + \frac{1}{2} \sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + (r - D_0)S \frac{\partial V}{\partial S} - rV = 0.
\]  \hfill (2.20)

In the following thesis, we employ Eq. (2.20) to study the pricing problems of American put options on a constant dividend yield paying asset. The constant dividend yield case is most suitable for options on foreign currencies; and it can be easily extended to the case for options on commodities as well. In the next section, we shall present a partial differential system, which governs American puts pricing problems.

2.4 The Partial Differential Equation System

To assist our readers for a better understanding, here we repeat the notations of the option parameters once again. The value of American put options also depends on the following parameters:

- \( \sigma \), the volatility of the underlying asset;
- \( T \), the contract life time;
- \( X \), the strike price;
- \( r \), the constant risk-free interest rate;
- \( D_0 \), the constant dividend yields;

Without loss of generality, we assume that both the risk-free interest rate and the dividend yield be constants. Our code can be easily modified for the case when they are some known functions of time and asset values.

It is well known [19] that American options can be decomposed into its European counterparts plus an early exercise premium. This early exercise premium is
associated with the extra right embedded in American options in comparison with its European counterparts. As a result of the additional freedom of being able to exercise prior to the expiry, a special set of boundary conditions need to be prescribed together with the Black-Scholes governing equation. For American put options, there exists an optimal exercise boundary $S_f(t)$ that separates the region of continue hold of the option from the region in which the option should be exercised. This optimal exercise boundary, sometimes refers to the critical asset price that is denoted as $S_f(t)$ in this thesis, and it is normally a function of time. In other words, when the underlying asset value reaches this optimal exercise boundary from the above for a put option (or from the below for a call option), the holder of the option should exercise his option, otherwise the option would be “too deeply” in-the-money. In fact, it can be shown that if the option is not exercised when the optimal exercise boundary is reached, the assumption of “arbitrage-free principle” in the Black-Scholes framework will be violated (see Wilmott et al. [30]). Therefore, to correctly and accurately compute this optimal exercise boundary, which is a function of time, is the main difficulty in the valuation of American put options. As shown in Wilmott et al. [30], there are two boundary conditions of the optimal exercise price $S = S_f(t)$,

$$\begin{align*}
V(S_f(t), t) &= X - S_f(t), \\
\frac{\partial V}{\partial S}(S_f(t), t) &= -1.
\end{align*}$$

To close the system, we need another boundary condition at the end of large asset value and a terminal condition, which is the payoff of the contract at the expiry day. The far-field boundary condition is

$$\lim_{S \to \infty} V(S, t) = 0,$$
and the terminal condition for a put option is

$$V(S, T) = \max\{X - S, 0\}. \quad (2.23)$$

In summary, the partial differential equations system for pricing American put options can be written as:

$$\begin{cases}
\frac{\partial V}{\partial t} + \frac{1}{2} \sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + (r - D_0)S \frac{\partial V}{\partial S} - rV = 0, \\
V(S_f(t), t) = X - S_f(t), \\
\frac{\partial V}{\partial S}(S_f(t), t) = -1, \\
\lim_{S \to \infty} V(S, t) = 0, \\
V(S, T) = \max\{X - S, 0\}.
\end{cases} \quad (2.24)$$

To solve the differential system Eq. (2.24) effectively, we normalize all variables in the system by introducing the following scale of variables,

$$V' = \frac{V}{X}, \quad S' = \frac{S}{X}, \quad \tau = (T - t)\frac{\sigma^2}{2}, \quad \gamma = \frac{2r}{\sigma^2}, \quad D = \frac{2D_0}{\sigma^2}, \quad S'_f(\tau) = \frac{S_f(\tau)}{X}.$$  

V, S and X are real numbers with the domain $\in (0, +\infty)$. r, D, and $\sigma$ are also real numbers with the domain $\in (0, 1)$. T is the maturity time in years, while $T - t$ is the time to maturity date in years, and $(T - t) \in [0, +\infty)$. As the result of normalization, the strike price becomes the unity in the normalized differential system. $\tau$ is the normalized time to expiry and this has changed the original problem into a forward diffusion problem. After dropping all primes, the normalized differential system becomes

$$\begin{cases}
\frac{\partial V}{\partial \tau} - S'^2 \frac{\partial^2 V}{\partial S'^2} - (\gamma - D)S' \frac{\partial V}{\partial S'} + \gamma V = 0, \\
V(S'_f(\tau), \tau) = 1 - S'_f(\tau), \\
\frac{\partial V}{\partial S'}(S'_f(\tau), \tau) = -1, \\
\lim_{\tau \to \infty} V(S, \tau) = 0, \\
V(S, 0) = \max\{1 - S, 0\}.
\end{cases} \quad (2.25)$$

So far, we have already presented the partial differential equations system for the pricing problems of American put options in the constant dividend yield case. In
the next two chapters, we shall solve the American option pricing problems through employing the Laplace Transform Method and a predictor-corrector scheme to solve the partial differential equations system Eq. (2.25).
3. THE LAPLACE TRANSFORM METHOD

This chapter presents an efficient numerical approach, based on the Laplace transform to price American puts. Several researchers applied the Laplace transform to option pricing problems in the past decade. For examples, Chesney et al. [9] applied the Laplace transform to price Parisian Barrier options. Mallier and Alobaidi [22] adopted the Laplace transform with asymptotic expansion for finding a solution of American options on a constant dividend yield asset. Unfortunately, Chesney et al. [9] and Mallier and Alobaidi [22] only gave out an integral equation in the Laplace space without performing the Laplace inversion. Recently, Zhu [35] presented a new method which uses the Laplace transform to find an analytical-approximation solution of the optimal exercise boundary of American put options. However, his analytical inversion appears to be limited to the non-dividend yield case only; extending this approach to the general case of pricing American-style options on the assets with constant dividend yield payment, would further justify the application of the Laplace transform as a practically useful numerical approach for pricing American options. In this chapter, we apply the Laplace Transform Method to solve a more general American put options pricing problem. After deriving the expressions of optimal exercise price and option value in the Laplace space, we obtain the corresponding values in the original time space by performing numerical inversion. The main advantage of the method is its high efficiency since there is no time marching nor iteration required.

In the context of option pricing, correctly calculating the Greeks is often as important as calculating the price itself for the purpose of hedging. But, some lattice-based approaches generate poor Delta values while they can produce very
accurate option prices. For example, the Finite Difference Method may produce some localized oscillations in option value, which result in significantly large errors in the calculation of the Delta and Gamma (see, Tavella and Randall [28]). Giles and Carter [15] even showed that the Delta of a vanilla European option calculated by the Crank-Nicolson method, which is a frequently-adopted approach in solving the Black-Scholes equation, exhibits convergence problems without some special treatments proposed by them. However, as will be shown in this chapter, the Greeks such as Delta can be easily computed by the Laplace Transform Method, because all the Greeks are calculated in the Laplace space analytically before the numerical inversion, and thus the order of the numerical errors is the same as that induced in the calculation of the option price itself.

The chapter is organized as follows. Section 3.1 discusses the procedure of deriving the optimal exercise price, and the option value in the Laplace space. Section 3.2 introduces three Laplace inversion algorithms which are potentially suitable for option pricing problems. Then, some test results for these algorithms applied to two standard test functions are presented in Section 3.3. Section 3.4 discusses the numerical accuracy and efficiency of the Laplace Transform Method being actually applied to option pricing problems. To demonstrate a great advantage of the current approach, the calculation of the Greeks (we used the calculated Delta as an example) is presented in the Section 3.4 as well.

3.1 Solutions of Optimal Exercise Price and Option Value

In this section, we present the Laplace Transform Method for solving the American put option pricing problems with employing the partial differential equations system in Chapter 2. For the easiness of reading, we repeat the normalized differential
system that determines the price of an American put option,

\[
-\frac{\partial V}{\partial \tau} + S^2 \frac{\partial^2 V}{\partial S^2} + (\gamma - D)S \frac{\partial V}{\partial S} - \gamma V = 0,
\]

\[
V(S_f(\tau), \tau) = 1 - S_f(\tau),
\]

\[
\frac{\partial V}{\partial S}(S_f(\tau), \tau) = -1,
\]

\[
\lim_{S \to \infty} V(S, \tau) = 0,
\]

\[
V(S, 0) = \max\{1 - S, 0\}. \tag{3.1}
\]

In Eq. (3.1), \(V\) is the option value, \(S\) is the underlying asset price, \(\tau\) is the time to expiry and \(S_f(\tau)\) is the optimal exercise price, which is to be determined as a part of the solution of the differential system. The unity in Eq. (3.1) is a result of normalizing the option price as well as the underlying asset price with respect to the strike price. The relationship between the normalized parameters and the original parameters is given in Chapter 2, and is thus not repeated here. It should be pointed out that the last equation of Eq. (3.1) is an initial condition, which corresponds to the terminal condition in the original dimensional system. The presence of the initial condition motivated us to take the advantage of the Laplace transform.

As well known, if the underlying asset price lies between the \(S_f\) and the strike price, the option is in the money and the option holder will have opportunities for an early exercise (see Zakamouline [32]). Naturally, we would introduce a new function \(U\) defined according to the value of the American put option in the following two scenarios as

\[
U = \begin{cases} 
V + S - 1, & \text{if } S_f \leq S < 1, \\
V, & \text{if } S \geq 1.
\end{cases} \tag{3.2}
\]

Then, in terms of \(U\), the governing differential systems are

\[
-\frac{\partial U}{\partial \tau} + S^2 \frac{\partial^2 U}{\partial S^2} + (\gamma - D)S \frac{\partial U}{\partial S} + DS = \gamma(U + 1),
\]

\[
U(S_f(\tau), \tau) = 0, \quad \text{if } S_f \leq S < 1,
\]

\[
\frac{\partial U}{\partial S}(S_f(\tau), \tau) = 0,
\]

\[
U(S, 0) = 0, \tag{3.3}
\]
The introduction of the new unknown function $U$ has made it more convenient to solve the system in the Laplace space as all the boundary conditions in Eq. (3.3) are homogeneous. In order to maintain the continuity of the option value as well as its Delta when the underlying asset price equals to the strike price (i.e. $S=1$), we need to impose two extra conditions on $U$:

\[
\lim_{S \rightarrow 1^-} U = \lim_{S \rightarrow 1^+} U, \\
\lim_{S \rightarrow 1^-} \frac{\partial U}{\partial S} = \lim_{S \rightarrow 1^+} \frac{\partial U}{\partial S} + 1.
\]

We solve the differential system Eq. (3.3) to Eq. (3.6) by means of the Laplace transform, which has been widely used in engineering and science in dealing with initial value problems. We define $\tilde{U}(S, p) = L(U(S, \tau))$, where $L(.)$ denotes the Laplace transform. Here we use it to temporarily remove the time variable $\tau$. After performing the Laplace transform, Eq. (3.3) to Eq. (3.6) become

\[
\begin{cases}
S^2 \frac{d^2 \tilde{U}}{dS^2} + (\gamma - D)S \frac{d\tilde{U}}{dS} - (\gamma + p)\tilde{U} = \frac{\gamma - DS}{p}, \\
\tilde{U}(p\tilde{S}_f, p) = 0, \\
\frac{d\tilde{U}}{dS}(p\tilde{S}_f, p) = 0,
\end{cases}
\]

\[
\begin{cases}
S^2 \frac{d^2 \tilde{U}}{dS^2} + (\gamma - D)S \frac{d\tilde{U}}{dS} - (\gamma + p)\tilde{U} = 0, \\
\lim_{S \rightarrow \infty} \tilde{U}(S, p) = 0,
\end{cases}
\]

\[
\begin{cases}
\tilde{U}(1^-, p) = \tilde{U}(1^+, p), \\
\frac{d\tilde{U}}{dS}(1^-, p) = \frac{d\tilde{U}}{dS}(1^+, p) + \frac{1}{p}.
\end{cases}
\]

The Laplace parameter $p$ is a complex variable, the values of which are restricted
on part of the complex plane. An important point of Eq. (3.7) should be noted; the Pseudo-Steady-State approximation has been used before the Laplace transform is performed.

This approximation usually allows researchers to model a small portion of an extremely complex system. If we are interested in the portion of the model that contains the slow, or rate limiting reactions, we can (sometimes) assume that the fast reactions are in a state of dynamic equilibrium, and the their derivatives are equal to zero, compared to the slow reactions. If we are interested in the portion of the model that has fast reactions, you can assume that the the slow portion does not change significantly (and thus, it’s derivative is zero) when compared to the fast. Here, we are interested in the optimal exercise price $S_f$, which has the slow reactions comparing with the asset price $S$. The optimal exercise price $S_f$ is held as a constant during the Laplace transform and replaced by $pS_f$ (see, Zhu [35]).

Now, by using some standard techniques to solve the ordinary differential equations (ODE), the solutions of Eq. (3.7) and Eq. (3.8) can be easily found as,

$$
\bar{U} = \begin{cases} 
C_1 S^{q_1} + C_2 S^{q_2} + \frac{(DS - \gamma)p + (S - 1)\gamma D}{p(p + \gamma)(p + D)}, & \text{if } S_f \leq S < 1, \\
C_3 S^{q_1} + C_4 S^{q_2}, & \text{if } S \geq 1,
\end{cases}
$$

(3.10)

where $q_1$ and $q_2$ are the roots of the corresponding characteristic equation for the non-homogeneous ODE in Eq. (3.7), and the characteristic equation of the homogeneous ODE in Eq. (3.8) itself,

$$
q_{1,2} = b \pm \sqrt{b^2 + (p + \gamma)}, \quad b = \frac{1 + D - \gamma}{2}.
$$

(3.11)

In Eq. (3.10), $C_1$ to $C_4$ are arbitrary complex functions of $p$, which are determined according to the boundary conditions. If we set the real part of $p$ large than $-\gamma$, then the real part of $q_1$ would always be positive, while the real part of $q_2$ would always be negative. To satisfy the boundary conditions in Eq. (3.8), $C_3$ must be set to zero (see Zhu [35]). After all the boundary conditions in Eq. (3.7) and Eq. (3.9)
are imposed, a set of algebraic equations is obtained,

\[
\begin{align*}
C_1(p \bar{S}_f)^q_1 + C_2(p \bar{S}_f)^q_2 + \frac{(Dp\bar{S}_f - \gamma)p + (p\bar{S}_f - 1)\gamma D}{p(p + \gamma)(p + D)} &= 0, \\
C_1q_1(p \bar{S}_f)^{q_1-1} + C_2q_2(p \bar{S}_f)^{q_2-1} + \frac{D}{p(p + D)} &= 0, \\
C_1(1)^{q_1} + C_2(1)^{q_2} - \frac{D - \gamma}{(p + D)(p + \gamma)} &= C_4(1)^{q_2}, \\
C_1q_1(1)^{q_1} + C_2q_2(1)^{q_2} + \frac{1}{p(p + D)} &= C_4q_2(1)^{q_2} + \frac{1}{p},
\end{align*}
\]

(3.12)

In Eq. (3.12), there are three \( C \)s complex functions which are dependent on \( \bar{S}_f \) and \( p \); also we have the \( \bar{S}_f \) that is only dependent on \( p \). Upon solving the algebraic equation system Eq. (3.12), we obtain an expression of the optimal exercise price in the Laplace space,

\[
\bar{S}_f(p)^{q_1} + \frac{\gamma + p + (D - \gamma)q_2}{q_2(p + \gamma)(p + D)} + \bar{S}_f(p)D\frac{1 - q_2}{p^1q_2(p + D)} = -\frac{\gamma}{p^{1+q_1}(p + \gamma)}. \quad (3.13)
\]

In order to find out the put options value, we have to work out the expressions of the other three \( C \)s functions, which at this stage can only be written in terms of \( \bar{S}_f \) as

\[
\begin{align*}
C_1 &= -\frac{D(p \bar{S}_f)^{1-q_1}}{q_1p(p + D)} - \frac{q_2}{q_1 - q_2} \cdot \left\{ \frac{D(p \bar{S}_f)^{1-q_1} + D(p \bar{S}_f)^{1-q_1} + p}{q_1p(p + D)} - \frac{D - \gamma}{(p + \gamma)(p + D)} \right\}, \\
C_2 &= -\frac{q_1}{q_1 - q_2} \cdot \left\{ \frac{D(p \bar{S}_f)^{1-q_2}}{q_1p(p + D)} - \frac{D(p \bar{S}_f)^{1-q_2} + p}{q_2p(p + D)} - \frac{D - \gamma}{(p + \gamma)(p + D)} \right\}, \\
C_4 &= -\frac{D(p \bar{S}_f)^{1-q_1} + p}{q_2p(p + D)} + C_2(1 - (p \bar{S}_f)^{q_2-q_1}). 
\end{align*}
\]

(3.14)

After these \( C \)s functions are substituted into Eq. (3.10), an expression of the option
3. The Laplace Transform Method

The value in the Laplace space can be found as,

\[
\bar{U} = \begin{cases} 
- \frac{D(p\bar{S}_f)^{1-q_1}}{q_1 p(p + D)} - \frac{q_2}{q_1 - q_2} \left\{ \frac{D(p\bar{S}_f)^{1-q_1}}{q_1 p(p + D)} - \frac{D(p\bar{S}_f)^{1-q_1} + p}{q_2 p(p + D)} - \frac{D - \gamma}{(p + \gamma)(p + D)} \right\} \cdot S^{q_1} \\
+ \frac{q_1}{q_1 - q_2} (p\bar{S}_f)^{q_1-q_2} \cdot \left\{ \frac{D(p\bar{S}_f)^{1-q_1}}{q_1 p(p + D)} - \frac{D(p\bar{S}_f)^{1-q_1} + p}{q_2 p(p + D)} - \frac{D - \gamma}{(p + \gamma)(p + D)} \right\} \cdot S^{q_2} \\
+ \frac{1 - (p\bar{S}_f)^{q_2-q_1}}{p(p + \gamma)(p + D)} \cdot S^{q_2} 
\end{cases} 
\]

\[
\text{if } S \leq S < 1, \\
\text{and } \bar{U} = - \frac{D(p\bar{S}_f)^{1-q_1} + p}{q_2 p(p + D)} S^{q_2} 
\]

\[
\text{if } S \geq 1. 
\]

(3.15)

So far, we obtain the expressions of \( \bar{S}_f \) and \( \bar{U} \) in the Laplace space. Unlike Zhu’s [35] analytical inversion case, an explicit expression of \( \bar{S}_f \) can not be easily found due to the nonlinear feature of Eq. (3.13). To obtain the value of \( \bar{S}_f \), we resort to Maple’s built-in solver to compute \( \bar{S}_f \) for each given \( p \) value. Then we perform the numerical Laplace inversion on \( \bar{S}_f \) to derive the value of \( S_f \) in the original time space.

### 3.2 Numerical Laplace Inversion

To find the original \( S_f(\tau) \) and \( V(S, \tau) \), the Laplace transform inversion needs to be performed. Ideally, we would like to perform the inversion of the Laplace transform analytically. However, unlike in a very special case Zhu [35] studied, i.e., American put options with no dividend yield, analytical inversion for options with continuous dividend yield cannot be readily found because one cannot solve for a clean expression of \( \bar{S}_f \) from Eq. (3.13); a nonlinear algebraic equation of \( \bar{S}_f \) would be obtained if one eliminates all the Cs in Eq. (3.12). Therefore, for more general cases, one has to resort to numerical methods for the Laplace inversion. This is also the case in many engineering applications, and it is for this reason a wide application of this very powerful approach for solving ODEs and PDEs has been limited. Thus, for the
interest of extending the approach in Zhu [35] to more general cases in option pricing, we need to investigate accuracy, easiness to use and reliability of the available numerical inversion techniques.

There are many numerical Laplace inversion algorithms, some of which are better for certain classes of functions but worse for others. For example, Cheng and Siduruk [8] reviewed several commonly-chosen inversion methods, which include the Stehfest method, the Papoulis method, the Durbin method, the Weeks method and the Piessens method. In the case of option pricing, because one may not be able to find an explicit dependence of $\bar{S}_f$ on $p$ in the Laplace space and consequently performing an analysis on the poles of $\bar{S}_f(p)$, only those methods that require no explicit location of poles of the to-be-inverted function are of interest to us. Moreover, we do not consider the algorithms that specially deal with the functions of periodic nature or spike shapes, because of the well known (see, Bunch and Johnson [6]) monotonicity property’s feature of the option prices as well as the optimal exercise prices of American puts. With these requirements in mind, we only focused on three algorithms, the Stehfest method, the Papoulis method, the linear combination method of Kwok and Barthez [20].

Although the detailed explanations of these algorithms have been given by Cheng and Siduruk [8] and Kowk and Barthez [20] already, for the completeness of this chapter and easiness to the readers, we briefly review each of the suitable inversion algorithms in the following subsections.

\subsection{Stehfest Method}

The Stehfest method, it is a favorable numerical inversion algorithm used in many engineering applications since it is simple to implement and reasonably reliable most of the time. For example, it has been used to numerically solve linear and non-linear thermal diffusion problems [36, 37].

In this approach, if $F(p)$ is the function to be inverted in the Laplace space, then the $f(t)$ in the original time space can be approximately calculated by
3. The Laplace Transform Method

\[
f(t) \approx \frac{ln(2)}{t} \sum_{n=1}^{N} c_n F\left(\frac{nln(2)}{t}\right),
\]

\[
c_n = (-1)^{n+\frac{N}{2}} \times \sum_{k=\lceil \frac{n+1}{2} \rceil}^{\min(n, \frac{N}{2})} \frac{\frac{\alpha}{2} (2k)!}{(n-k)!k!(k-1)!(n-k)!(2k-n)!},
\]

(3.16)

In Eq. (3.16), \( N \) is an even integer number and \( k \) is the greatest integer less than or equal to \((n + 1)/2\), \( n \) is an integer and \( n \in [1, N] \). The Stehfest method has only one parameter \( N \) that needs to be chosen. Theoretically, the larger the value of \( N \), the more accurate inversion results should be. However, in practice, the maximum of \( N \) is also limited by truncation errors; a larger value of \( N \) does not necessarily lead to a smaller relative error. Cheng and Siduruk [8] suggested the range of \( N \) be chosen between 6 and 20.

3.2.2 Papoulis Legendre Polynomial Method

In this method, the Laplace transform inversion is obtained as a series expansion in terms of Legendre polynomials with the exponential function \( e^{-\rho t} \) as their argument:

\[
f(t) \approx \sum_{n=0}^{N} a_n P_{2n}(e^{-\rho t}), \quad \rho F[(2k + 1)\rho] = \sum_{m=0}^{k} \frac{(k-m+1)\alpha}{2(k+\frac{1}{2})m+1} a_m,
\]

(3.17)

where \( P_n(x) \) is the Legendre polynomial of order \( n \), \( a_n \) is computed from the second recursion formula, and \( k = n \). \( (j)_m \) denotes as the Pochhammer symbol [8]. With this method, we need to choose two parameters \( N \) and \( \rho \), with \( N \) being a positive integer and \( \rho \) being a real number. A value of \( \rho = 0.2 \) is recommended by Davies and Martin in 1979.

In our numerical experiments, we had to perform enormous computation to identify the range of \( N \) and \( \rho \), in which insensitive results to the values of these parameters could be obtained. Obviously, using this method is not as convenient as using the Stehfest method due to an extra parameter needed in the calculation.
Kwok and Barthez [20] used the linear combination of a basis function $f_0$ with different coefficients to calculate the $f(t)$:

$$
\begin{align*}
f(t) & \approx \frac{1}{90} (f_0(t; a, 1) + f_0(t; a, -1)) + \frac{2}{9} (f_0(t; a, \frac{1}{2}) + f_0(t; a, -\frac{1}{2})) \\
& \quad + \frac{32}{45} (f_0(t; a, \frac{1}{4}) + f_0(t; a, -\frac{1}{4})), \\
f_0(t; a, \alpha) &= \frac{e^{\alpha t}}{t^{\sqrt{\alpha}} |\alpha|} S_N^{(m+1)}(t; a, \alpha), \\
S_N^{(0)}(t; a, \alpha) &= \sum_{n=0}^{N} F_n(t; a, \alpha), \\
S_N^{(m+1)}(t; a, \alpha) &= S_N^{(m)}(t; a, \alpha) + \varepsilon^{(m)} \\
F_n(t; a, \alpha) &= (-1)^n \text{Re} F \left( \frac{a - \ln \sqrt{\alpha} + n\pi i}{t} \right) \left( \frac{2 - \delta}{t} \right) \quad \text{when } \alpha > 0 \text{ and } n \geq 0, \\
F_n(t; a, \alpha) &= (-1)^n \text{Im} F \left( \frac{a - \ln \sqrt{-\alpha} + (n - \frac{1}{2})\pi i}{t} \right) \quad \text{when } \alpha < 0 \text{ and } n \geq 1,
\end{align*}
$$

where $\delta = 1$ if $n = 0$ and $\delta = 0$ if $n \neq 0$. $N$ is an odd number and $a$ is a real number that is greater than zero. In fact, Kwok and Barthez [20] suggested choosing $a$ is a function of $t$, in order to prevent $a/t$ becoming too large. However, we are only interested in small time range, because the normalized time variable $\tau$ is usually very small for ordinary options. For example, for a 1-year life time option, the maximum normalized $\tau$ value is 0.045 with $\sigma = 30\%$, and this is clearly much smaller than unity. Therefore it suffices to set $a$ as a constant in our numerical experiments.

To compute the inversion value efficiently, Kwok and Barthez [20] adopted an acceleration algorithm or extrapolation that is used to estimate the limit of a sequence from its several leading terms. This extrapolation is represented by $\varepsilon^{(m)}$ in Eq. (3.18). The number of extrapolation steps is denoted as $m$ here. Owing to the limited space, we do not present the details of the extrapolation here. In this method, we need to select two parameters $N$ and $m$. Kwok and Barthez [20] suggested that the extrapolation steps should not be too many. In our experiments, the values of $N$ were selected as 5, 7 and 9, while $m$ were selected from 1 to 3.
3.3 Numerical Test of Standard Functions

Ideally, a robust inversion method should depend very little on the parameters involved in the numerical inversion, i.e., the results of numerical inversion should not be sensitive to the variation of these parameters. However, due to the inherent unstable nature of the numerical Laplace inversion, parametric tests for the sensitivity of a method to different parameter values always need to be performed. We examine the sensitivity of three potentially suitable algorithms by running two standard tests. A range of least-sensible parameters is determined empirically for the numerical inversion involved in option pricing. To determine a suitable inversion method for our option pricing problems, again we restrict ourselves to the case with small time feature. All the results of numerical inversion in this section were obtained by using Maple 10 on an Intel Pentium 4, 3GHz machine.

The two test functions for Laplace inversion transforms are:

\[
\mathcal{L}^{-1}(\frac{\ln p}{p}) = -\gamma - \ln t, \tag{3.19}
\]

\[
\mathcal{L}^{-1}(\frac{2}{p}K_0(2\sqrt{p})) = -Ei\left(-\frac{1}{t}\right). \tag{3.20}
\]

The \(Ei\) part in Eq. (3.20) is the exponential integral. The criteria we use to select a good inversion method is that the relative errors of the method must be less than a prescribed error tolerance with a range of parameters. To evaluate accuracy of the three inversion methods, we apply them to obtain the inversion values on a set of discrete time instants. Then we take the algebraic average of the absolute and relative errors of the calculated values. The average absolute errors and average relative errors are defined, respectively, as:

\[
\text{AveAbsErr} = \frac{1}{n} \sum_{i=1}^{n} |\hat{a}_i - a_i|/n, \tag{3.21}
\]

\[
\text{AveRelErr} = \frac{1}{n} \sum_{i=1}^{n} |(\hat{a}_i - a_i)/a_i|/n, \tag{3.22}
\]
where \( \tilde{a}_i \) is the result of numerical inversion while \( a_i \) is the exact value at the \( ith \) time point, \( n \) is the total number of the calculated sample points. In the standard test examples presented here, \( n \) is set to be 5, and an error tolerance threshold is set to be 5\%. This means that if the average relative errors are above 5\% after trying all reasonable parameters, the inversion method is deemed to be of insufficient accuracy. A range of least-sensible parameters is selected in terms of the lowest average relative errors in standard tests for option pricing case. Table 3.1 to Table 3.3 show the selected parameter range and the corresponding relative errors of each method respectively.

To be able to examine a suitable parameter \( N \) of the Stehfest method, several different values of \( N \) are used in the algorithm. Table 3.1 shows that the average relative errors and average absolute errors are the smallest with \( N = 10 \). We can also see that the least-sensible parameters \( N \) are 6, 8 and 10 based on the lowest errors easily identified from Table 3.1. Moreover, we can tell that different \( N \) values do not affect the sensitivity of the results of numerical inversion too much since the average relative errors are not sensitive to the variation of \( N \). These three values of \( N \) were selected as candidate parameters in option pricing case.

The results of applying the Papoulis method to the standard test functions are presented in Table 3.2. The minimum average relative errors are around 4\% for Eq. (3.19) with \( \rho = 0.6 \) and \( N = 10 \), whereas for Eq. (3.20), the method gave a very poor performance. The minimum average relative errors are around 26\% with \( \rho \) being 0.5 and \( N \) being 9. Comparing directly with the average relative errors in Table 3.2, we find that average relative errors are too high to satisfy the accuracy requirement. Thus, we failed to identify the least-sensible parameters. Moreover, the results are too sensitive to the changing of \( \rho \). The method is clearly unsatisfactory for the Laplace inversion problems with small time characteristic.

Table 3.3 reports the results of the numerical inversion which were produced by employing the linear combination method of Kwok and Barthez [20]. The constant parameter \( a \) was set at 1. Average relative errors are all less than 1\% regardless
the change both in $N$ and $m$. The least-sensible parameters in next section were selected with $N$ being 5, 7, 9 and $m$ being 1, 2, 3 respectively.

In summary, among the three methods, the Stehfest method is the best algorithm judged by the computational efficiency and easiness to use, while the linear combination method of Kwok and Barthez [20] is the best one among the three in terms of accuracy.
### Tab. 3.1: Results of the Stehfest Method

<table>
<thead>
<tr>
<th>Eq. (3.19)</th>
<th>$t = 0.2$</th>
<th>$t = 0.4$</th>
<th>$t = 0.6$</th>
<th>$t = 0.8$</th>
<th>$t = 1$</th>
<th>AveAbsErr</th>
<th>AveRelErr</th>
</tr>
</thead>
<tbody>
<tr>
<td>$N = 4$</td>
<td>0.99380789</td>
<td>0.30066072</td>
<td>-0.10480434</td>
<td>-0.39248647</td>
<td>-0.61563003</td>
<td>0.03840</td>
<td>0.181000</td>
</tr>
<tr>
<td>$N = 6$</td>
<td>1.03114349</td>
<td>0.33799624</td>
<td>-0.06746862</td>
<td>-0.35515102</td>
<td>-0.57829427</td>
<td>0.00108</td>
<td>0.005080</td>
</tr>
<tr>
<td>$N = 8$</td>
<td>1.03239941</td>
<td>0.33925552</td>
<td>-0.06621092</td>
<td>-0.35389167</td>
<td>-0.57703512</td>
<td>0.00018</td>
<td>0.000845</td>
</tr>
<tr>
<td>$N = 10$</td>
<td>1.03236999</td>
<td>0.33907893</td>
<td>-0.06636999</td>
<td>-0.35405861</td>
<td>-0.57723010</td>
<td>0.00004</td>
<td>0.000104</td>
</tr>
<tr>
<td>Eq. (3.20)</td>
<td>$t = 0.2$</td>
<td>$t = 0.4$</td>
<td>$t = 0.6$</td>
<td>$t = 0.8$</td>
<td>$t = 1$</td>
<td>AveAbsErr</td>
<td>AveRelErr</td>
</tr>
<tr>
<td>$N = 4$</td>
<td>-0.01049148</td>
<td>0.02415162</td>
<td>0.09157978</td>
<td>0.16941534</td>
<td>0.24870620</td>
<td>0.01560</td>
<td>2.130000</td>
</tr>
<tr>
<td>$N = 6$</td>
<td>-0.00129469</td>
<td>0.02370021</td>
<td>0.07997027</td>
<td>0.14967877</td>
<td>0.22335859</td>
<td>0.00251</td>
<td>0.448000</td>
</tr>
<tr>
<td>$N = 8$</td>
<td>0.00073768</td>
<td>0.02483974</td>
<td>0.07892998</td>
<td>0.14720333</td>
<td>0.22011407</td>
<td>0.00052</td>
<td>0.075400</td>
</tr>
<tr>
<td>$N = 10$</td>
<td>0.00105910</td>
<td>0.02496171</td>
<td>0.07852346</td>
<td>0.14657537</td>
<td>0.21946922</td>
<td>0.00012</td>
<td>0.016700</td>
</tr>
<tr>
<td>$t$</td>
<td>Eq.(3.19)</td>
<td>$t = 0.2$</td>
<td>$t = 0.4$</td>
<td>$t = 0.6$</td>
<td>$t = 0.8$</td>
<td>$t = 1$</td>
<td>$\text{AveAbsErr}$</td>
</tr>
<tr>
<td>-----</td>
<td>-----------</td>
<td>-----------</td>
<td>-----------</td>
<td>-----------</td>
<td>-----------</td>
<td>--------</td>
<td>----------------</td>
</tr>
<tr>
<td>$N = 8$</td>
<td>$\rho = 0.6$</td>
<td>1.08816867</td>
<td>0.29870909</td>
<td>-0.07024339</td>
<td>-0.30752772</td>
<td>-0.57699443</td>
<td>0.0294</td>
</tr>
<tr>
<td>$\rho = 0.7$</td>
<td>1.10290447</td>
<td>0.28670061</td>
<td>-0.03689834</td>
<td>-0.33329208</td>
<td>-0.61855821</td>
<td>0.0429</td>
<td>0.1590</td>
</tr>
<tr>
<td>$\rho = 0.8$</td>
<td>1.09260008</td>
<td>0.30428511</td>
<td>-0.01984772</td>
<td>-0.37362296</td>
<td>-0.62446611</td>
<td>0.0417</td>
<td>0.2000</td>
</tr>
<tr>
<td>$\rho = 0.9$</td>
<td>1.06771864</td>
<td>0.33517997</td>
<td>-0.03635838</td>
<td>-0.39974296</td>
<td>-0.59875006</td>
<td>0.0273</td>
<td>0.1330</td>
</tr>
<tr>
<td>$N = 9$</td>
<td>$\rho = 0.6$</td>
<td>1.06399234</td>
<td>0.34410184</td>
<td>-0.05725692</td>
<td>-0.38639425</td>
<td>-0.55809729</td>
<td>0.0195</td>
</tr>
<tr>
<td>$\rho = 0.7$</td>
<td>1.02212825</td>
<td>0.37449921</td>
<td>-0.09892359</td>
<td>-0.32248770</td>
<td>-0.57155821</td>
<td>0.0284</td>
<td>0.1450</td>
</tr>
<tr>
<td>$\rho = 0.8$</td>
<td>0.98500983</td>
<td>0.38231086</td>
<td>-0.07589516</td>
<td>-0.32276504</td>
<td>-0.59875006</td>
<td>0.0253</td>
<td>0.0711</td>
</tr>
<tr>
<td>$\rho = 0.9$</td>
<td>0.98345576</td>
<td>0.3726157</td>
<td>-0.09389834</td>
<td>-0.35343705</td>
<td>-0.54283057</td>
<td>0.0216</td>
<td>0.1180</td>
</tr>
<tr>
<td>$N = 10$</td>
<td>$\rho = 0.6$</td>
<td>1.05192366</td>
<td>0.35489352</td>
<td>-0.06760611</td>
<td>-0.38201621</td>
<td>-0.55042565</td>
<td>0.0183</td>
</tr>
<tr>
<td>$\rho = 0.7$</td>
<td>1.0192366</td>
<td>0.35489352</td>
<td>-0.06760611</td>
<td>-0.38201621</td>
<td>-0.55042565</td>
<td>0.0183</td>
<td>0.0419</td>
</tr>
<tr>
<td>$\rho = 0.8$</td>
<td>0.98822492</td>
<td>0.36690184</td>
<td>-0.09501630</td>
<td>-0.31920423</td>
<td>-0.57982259</td>
<td>0.0276</td>
<td>0.1320</td>
</tr>
<tr>
<td>$\rho = 0.9$</td>
<td>0.98620682</td>
<td>0.34099778</td>
<td>-0.06917281</td>
<td>-0.32633251</td>
<td>-0.61093892</td>
<td>0.0224</td>
<td>0.0458</td>
</tr>
</tbody>
</table>

$\text{Eq.}(3.20)$

<table>
<thead>
<tr>
<th>$t$</th>
<th>Eq.(3.20)</th>
<th>$t = 0.2$</th>
<th>$t = 0.4$</th>
<th>$t = 0.6$</th>
<th>$t = 0.8$</th>
<th>$t = 1$</th>
<th>$\text{AveAbsErr}$</th>
<th>$\text{AveRelErr}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$N = 8$</td>
<td>$\rho = 0.3$</td>
<td>-0.0178204</td>
<td>0.05332689</td>
<td>0.08635381</td>
<td>0.12360071</td>
<td>0.19713217</td>
<td>0.0199</td>
<td>3.4200</td>
</tr>
<tr>
<td>$\rho = 0.4$</td>
<td>0.01465437</td>
<td>0.3057325</td>
<td>0.05323119</td>
<td>0.14410700</td>
<td>0.2428315</td>
<td>0.0140</td>
<td>2.4900</td>
<td></td>
</tr>
<tr>
<td>$\rho = 0.5$</td>
<td>0.02694786</td>
<td>0.34823108</td>
<td>-0.07589516</td>
<td>-0.32276504</td>
<td>-0.60664765</td>
<td>0.0253</td>
<td>0.0711</td>
<td></td>
</tr>
<tr>
<td>$N = 9$</td>
<td>$\rho = 0.3$</td>
<td>-0.02211289</td>
<td>0.04031751</td>
<td>0.08635381</td>
<td>0.12573538</td>
<td>0.20871684</td>
<td>0.0156</td>
<td>4.2300</td>
</tr>
<tr>
<td>$\rho = 0.4$</td>
<td>-0.01742541</td>
<td>0.04306660</td>
<td>0.05814799</td>
<td>0.14457110</td>
<td>0.24195949</td>
<td>0.0163</td>
<td>3.4600</td>
<td></td>
</tr>
<tr>
<td>$\rho = 0.5$</td>
<td>0.0012185</td>
<td>0.02119426</td>
<td>-0.06917281</td>
<td>-0.32633251</td>
<td>-0.61093892</td>
<td>0.0224</td>
<td>0.0458</td>
<td></td>
</tr>
<tr>
<td>$N = 10$</td>
<td>$\rho = 0.3$</td>
<td>0.02099634</td>
<td>0.00606768</td>
<td>0.09140005</td>
<td>0.15636123</td>
<td>0.20356919</td>
<td>0.0155</td>
<td>3.6700</td>
</tr>
<tr>
<td>$\rho = 0.4$</td>
<td>0.00421973</td>
<td>0.02637842</td>
<td>0.0889192</td>
<td>0.12713183</td>
<td>0.21856013</td>
<td>0.0070</td>
<td>0.6010</td>
<td></td>
</tr>
<tr>
<td>$\rho = 0.5$</td>
<td>-0.01295149</td>
<td>0.04273483</td>
<td>0.06316345</td>
<td>0.14599707</td>
<td>0.23695276</td>
<td>0.0130</td>
<td>2.6500</td>
<td></td>
</tr>
</tbody>
</table>
### Tab. 3.3: Results of the Linear Combination Method

<table>
<thead>
<tr>
<th>Eq. (3.19)</th>
<th>$t = 0.2$</th>
<th>$t = 0.4$</th>
<th>$t = 0.6$</th>
<th>$t = 0.8$</th>
<th>$t = 1$</th>
<th>AveAbsErr</th>
<th>AveRelErr</th>
</tr>
</thead>
<tbody>
<tr>
<td>$N = 5$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$m = 1$</td>
<td>1.03221127</td>
<td>0.33904976</td>
<td>-0.06642892</td>
<td>-0.35412736</td>
<td>-0.57729452</td>
<td>4.18E-05</td>
<td>1.93E-04</td>
</tr>
<tr>
<td>$m = 2$</td>
<td>1.03211239</td>
<td>0.33907379</td>
<td>-0.06639042</td>
<td>-0.35406989</td>
<td>-0.57720640</td>
<td>2.45E-05</td>
<td>2.77E-05</td>
</tr>
<tr>
<td>$m = 3$</td>
<td>1.03215319</td>
<td>0.33907546</td>
<td>-0.06638686</td>
<td>-0.35407171</td>
<td>-0.57721517</td>
<td>1.45E-05</td>
<td>2.32E-05</td>
</tr>
<tr>
<td>$N = 7$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$m = 1$</td>
<td>1.03223377</td>
<td>0.33907379</td>
<td>-0.06639304</td>
<td>-0.35407663</td>
<td>-0.57721666</td>
<td>5.43E-06</td>
<td>1.70E-05</td>
</tr>
<tr>
<td>$m = 2$</td>
<td>1.03222210</td>
<td>0.33907484</td>
<td>-0.06639030</td>
<td>-0.35407240</td>
<td>-0.57721598</td>
<td>3.56E-07</td>
<td>1.52E-06</td>
</tr>
<tr>
<td>$m = 3$</td>
<td>1.03222080</td>
<td>0.33907477</td>
<td>-0.06638841</td>
<td>-0.35407367</td>
<td>-0.57721592</td>
<td>1.45E-05</td>
<td>6.07E-06</td>
</tr>
<tr>
<td>$N = 9$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$m = 1$</td>
<td>1.03222302</td>
<td>0.33907511</td>
<td>-0.06639048</td>
<td>-0.35407294</td>
<td>-0.57721683</td>
<td>8.04E-07</td>
<td>2.69E-06</td>
</tr>
<tr>
<td>$m = 2$</td>
<td>1.03222098</td>
<td>0.33907486</td>
<td>-0.06639026</td>
<td>-0.35407234</td>
<td>-0.57721581</td>
<td>3.14E-07</td>
<td>1.31E-06</td>
</tr>
<tr>
<td>$m = 3$</td>
<td>1.03222209</td>
<td>0.33907485</td>
<td>-0.06390301</td>
<td>-0.35407241</td>
<td>-0.57721183</td>
<td>4.98E-04</td>
<td>7.49E-03</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$Eq.(3.20)$</th>
<th>$t = 0.2$</th>
<th>$t = 0.4$</th>
<th>$t = 0.6$</th>
<th>$t = 0.8$</th>
<th>$t = 1$</th>
<th>AveAbsErr</th>
<th>AveRelErr</th>
</tr>
</thead>
<tbody>
<tr>
<td>$N = 5$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$m = 1$</td>
<td>0.00114814</td>
<td>0.02553168</td>
<td>0.07833707</td>
<td>0.14645986</td>
<td>0.21929464</td>
<td>1.51E-04</td>
<td>5.13E-03</td>
</tr>
<tr>
<td>$m = 2$</td>
<td>0.00114827</td>
<td>0.02522278</td>
<td>0.07833421</td>
<td>0.14645820</td>
<td>0.21926920</td>
<td>9.36E-05</td>
<td>2.64E-03</td>
</tr>
<tr>
<td>$m = 3$</td>
<td>0.00114797</td>
<td>0.02489054</td>
<td>0.07833938</td>
<td>0.14645462</td>
<td>0.21928473</td>
<td>3.42E-05</td>
<td>4.13E-04</td>
</tr>
<tr>
<td>$N = 7$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$m = 1$</td>
<td>0.00114836</td>
<td>0.02491411</td>
<td>0.07833085</td>
<td>0.14642993</td>
<td>0.21938121</td>
<td>4.55E-06</td>
<td>5.04E-05</td>
</tr>
<tr>
<td>$m = 2$</td>
<td>0.00114836</td>
<td>0.02491483</td>
<td>0.07833165</td>
<td>0.14641348</td>
<td>0.21938418</td>
<td>4.60E-07</td>
<td>1.78E-05</td>
</tr>
<tr>
<td>$m = 3$</td>
<td>0.00114836</td>
<td>0.02491431</td>
<td>0.07833236</td>
<td>0.14643715</td>
<td>0.21938405</td>
<td>5.13E-06</td>
<td>5.24E-05</td>
</tr>
<tr>
<td>$N = 9$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$m = 1$</td>
<td>0.00114837</td>
<td>0.02491500</td>
<td>0.07833384</td>
<td>0.14641393</td>
<td>0.21938699</td>
<td>8.38E-07</td>
<td>1.92E-05</td>
</tr>
<tr>
<td>$m = 2$</td>
<td>0.00114837</td>
<td>0.02491502</td>
<td>0.07833360</td>
<td>0.14641354</td>
<td>0.21938709</td>
<td>7.36E-07</td>
<td>1.83E-05</td>
</tr>
<tr>
<td>$m = 3$</td>
<td>0.00114837</td>
<td>0.02491501</td>
<td>0.07833359</td>
<td>0.14641354</td>
<td>0.21938707</td>
<td>7.28E-07</td>
<td>1.82E-05</td>
</tr>
</tbody>
</table>
3. The Laplace Transform Method

3.4 Numerical Examples for American Puts

Upon completing the tests in previous section, we concluded that the Stehfest method in Cheng and Siduruk [8], the linear combination method in Kwok and Barthez [20] were the two most suitable approaches for the inversion needed in the option pricing problems, with the small time characteristic due to the normalization. But, it was still not clear which one of these methods would be more suitable for pricing options if the Laplace transform is to be used. In this section, we show the results that were carried out by the two inversion methods being applied to option pricing problems.

3.4.1 Numerical Accuracy

To test accuracy of a numerical Laplace inversion, let’s first compare our numerical results with those of a case in which the analytical inversion has been successfully performed by Zhu [35]; for a special case with no dividend payments, Zhu [35] worked out the inversion analytically. So, we first adopt the same example used by Zhu [35] for the purpose of determining accuracy of the numerical Laplace inversion.

The normalized option parameters in this example are: $\gamma = 2.222$ and $D = 0$ with $\sigma = 30\%$, $\tau_{\text{max}} = 0.045$, which corresponds to an option with 1-year life time. Zhu [35] compared his results with those of Wu and Kwok [31], and Bunch and Johnson [6]. He found that the three sets of results agree reasonably well. If we can demonstrate that our numerical results through the numerical Laplace inversion agree with the results of Zhu [35], we are confident that our results would compare favorably with the other two sets of numerical results as well.

To evaluate accuracy of the Stehfest method, we perform the algorithm on ten normalized time instants for the experiment. Table 3.4 shows when $N$ equals to 8, the Stehfest method yields the minimum average relative errors. In the next subsection, we also set $N$ at 8 in the case with dividend yields. The low errors indicate that the Stehfest method works perfectly in the option pricing problems. In Fig.
3.1, we show the plot of the optimal exercise price against the time to expiry. Apparently, our curve perfectly matches the analytical-approximation inversion curve in Zhu [35].

Surprisingly, Kwok and Barthez’s method does not work in this case at all. For example, from Fig. 3.1, we know that the value of $S_f$ is around 0.76 at $\tau = 0.045$. Table 3.5 shows that unreasonable results of $S_f$ at $\tau = 0.045$ were produced by their method no matter what combinations of $N$ and $m$ values we chose; some of these $S_f$ values even become negative. We have tested a wide range of parameters, however, unlike the luck we had when the method was applied to the standard test functions shown in the previous section where excellent results were achieved, their method does not seem to work when we apply it to option pricing problems.

After comparing these three methods, the Stehfest method is clearly the best inversion approach suitable for the option pricing problems. There are mainly four advantages for adopting the Stehfest method. Firstly, the Stehfest method computes
3. The Laplace Transform Method

Tab. 3.4: Numerical Inversion of the Stehfest Method for Option Pricing

<table>
<thead>
<tr>
<th>( \tau )</th>
<th>( N = 6 )</th>
<th>( N = 8 )</th>
<th>( N = 10 )</th>
<th>\textit{Analytic Inversion}</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.005</td>
<td>0.84854646</td>
<td>0.84863630</td>
<td>0.84833590</td>
<td>0.84830692</td>
</tr>
<tr>
<td>0.010</td>
<td>0.81914464</td>
<td>0.81923834</td>
<td>0.81926323</td>
<td>0.81923386</td>
</tr>
<tr>
<td>0.015</td>
<td>0.80149013</td>
<td>0.80158423</td>
<td>0.80160762</td>
<td>0.80187124</td>
</tr>
<tr>
<td>0.020</td>
<td>0.78889894</td>
<td>0.78899394</td>
<td>0.78900146</td>
<td>0.78790613</td>
</tr>
<tr>
<td>0.025</td>
<td>0.77919868</td>
<td>0.77928735</td>
<td>0.77929831</td>
<td>0.77928012</td>
</tr>
<tr>
<td>0.030</td>
<td>0.77137149</td>
<td>0.77146010</td>
<td>0.77145895</td>
<td>0.77144913</td>
</tr>
<tr>
<td>0.035</td>
<td>0.76485816</td>
<td>0.76494019</td>
<td>0.76497703</td>
<td>0.76493095</td>
</tr>
<tr>
<td>0.040</td>
<td>0.75931502</td>
<td>0.75939195</td>
<td>0.75939992</td>
<td>0.75938297</td>
</tr>
<tr>
<td>0.045</td>
<td>0.75451730</td>
<td>0.75458543</td>
<td>0.75462317</td>
<td>0.75458026</td>
</tr>
<tr>
<td>AveAbsErr</td>
<td>0.00022949</td>
<td>0.00019447</td>
<td>0.00032054</td>
<td></td>
</tr>
<tr>
<td>AveRelErr</td>
<td>0.00028900</td>
<td>0.00024300</td>
<td>0.00040800</td>
<td></td>
</tr>
</tbody>
</table>

Tab. 3.5: Numerical Inversion of the Linear Combination Method at \( \tau = 0.045 \)

<table>
<thead>
<tr>
<th>( m )</th>
<th>( N = 5 )</th>
<th>( N = 7 )</th>
<th>( N = 9 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-0.31425753</td>
<td>-0.65868629</td>
<td>-1.44905280</td>
</tr>
<tr>
<td>2</td>
<td>0.04847262</td>
<td>-0.24705103</td>
<td>-0.60737056</td>
</tr>
<tr>
<td>3</td>
<td>0.45440735</td>
<td>0.43772053</td>
<td>-0.55868147</td>
</tr>
</tbody>
</table>

the optimal exercise price independently for each single time value using a delta-convergent series. The values of each time step are independent to each other and thus potentially the inversion can be computed in parallel to further enhance the computational efficiency. Secondly, the method only needs one constant parameter, and it is much easier to determine the optimal value of one parameter. Thirdly, the Stehfest method is accurate when it is handling small-time range, which is ideal for our problem here. Finally, the whole idea of the Stehfest method is to keep the numerical inversion simple and efficient, which is of particular attraction to market practitioners.

3.4.2 Numerical Efficiency

In this subsection, numerical efficiency is explored through bench-marking the results obtained with the Laplace Transform Method against those obtained with the traditional Binomial Method in Cox et al. [11]. The reason we chose the Binomial
Method to compare with is because it is still widely used in finance industry today due to its reliability and simplicity. In addition, accurate results for American options can be obtained since the possibility of early exercise can be easily checked at each binomial node.

Table 3.6 shows a comparison of the CPU time used to calculate all $S_f$ values on an Intel Pentium 4, 3GHz machine, using the Binomial Method and the Laplace Transform Method, respectively. The parameters used for this example are: the strike price $X = $100, the risk-free interest rate $r = 5\%$, the dividend yield $D_0 = 5\%$, the volatility $\sigma = 30\%$ and 1-year lifetime of the option. The $S_f$ values shown in Table 3.6 were recorded at the time to expiry being $2/3$ year. Obviously the efficiency of the latter approach is much higher than that of the former. With the number of time steps, $M$, being increased from 1000 to 3000, the CPU time used by the Binomial Method has increased exponentially while the CPU time used by the numerical Laplace inversion method has only increased linearly. In fact, to produce a curve of $S_f$ as a function of time to expiry with the number of time steps $M = 3000$, the numerical Laplace inversion method implemented inside of Maple 10 were more than 60 times faster in comparison with the Binomial Method. One of the main reasons for the higher efficiency of the Laplace Transform Method is attributed to the fact that numerical calculations are only performed for the inversion part, whereas the calculation of the Laplace transform is done analytically and thus waste no computational resources at all.

Another advantage of using the Laplace Transform Method is that if only the optimal exercise price or the option price is needed at a particular time, there is no need to discretize the time axis to very fine grids. In other words, the results are not very sensitive to the grid resolution. For example, for the case discussed here, discretizing the time axis with 1000 steps produced the same results as the case with the time axis being discretized with 3000 steps. In fact, this nice property of the Laplace Transform Method has been discussed in Zhu et al. [37], Zhu and Satravaha [36] for a heat transfer problem.
3. The Laplace Transform Method

### Tab. 3.6: Comparison of Efficiency

<table>
<thead>
<tr>
<th>$M$</th>
<th>$S_f($)$</th>
<th>Bino CPU Time(s)</th>
<th>$S_f($)$ Lap.Inv.</th>
<th>CPU Time(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1000</td>
<td>72.78</td>
<td>264</td>
<td>70.31</td>
<td>84</td>
</tr>
<tr>
<td>1500</td>
<td>72.59</td>
<td>809</td>
<td>70.32</td>
<td>127</td>
</tr>
<tr>
<td>2000</td>
<td>72.46</td>
<td>3499</td>
<td>70.31</td>
<td>173</td>
</tr>
<tr>
<td>2500</td>
<td>72.31</td>
<td>8747</td>
<td>70.30</td>
<td>220</td>
</tr>
<tr>
<td>3000</td>
<td>72.26</td>
<td>17128</td>
<td>70.31</td>
<td>274</td>
</tr>
</tbody>
</table>

### 3.4.3 Accuracy of the Laplace Transform Method

So far, we have presented three numerical inversion methods to obtain the optimal exercise price. We have already shown the low average relative errors by employing the Stehfest method. However, as stated earlier the errors due to the pseudo-steady-state approximation introduced when the Laplace transform is performed are not part of the errors due to the numerical inversion. It would be interesting to examine the overall accuracy of the entire approach of the Laplace transform. In this subsection, we present some comparison between the results obtained using the current approach and those obtained with Zhu’s [34] exact solution for American put options.

The option parameters are the same as those in the discussion of numerical accuracy. We have plotted the $S_f$ values in Fig. 3.2 for comparison. From the figure, one can observe that the numerical solution computed by the Laplace Transform Method consistently produces a lower $S_f$ in comparison with that produced by the exactly analytic solution in Zhu [34]. The degree of the under-estimation varies with the time to expiry. Closer to expiry, the under-estimate is roughly 2%, whereas it quickly reduces down to less than 1% when the time to expiry becomes 1 year. Given that the errors introduced by the numerical inversion are far less than 1-2% as shown in the previous subsection, the source of error is clearly the pseudo-steady-state approximation introduced to allow the Laplace transform to be performed on the moving boundary. This approximation has greatly simplified the solution procedure at the expenses of the boundary conditions on the moving boundary not
being exactly satisfied. To achieve the fast speed offered by the current approach, the slightly reduction of accuracy appears to be a well-worth-paid price.

3.4.4 Calculation of the Delta

Calculating the Greeks can be easily performed with the Laplace Transform Method. In this subsection, we shall use the calculation of Delta as an example; the calculation of other Greeks is very similar.

A great advantage of this method is that an analytic expression of Delta can be derived in the Laplace space first. Then, like the calculation of option price, the value of Delta in the original time space can be restored by employing the same numerical inversion that is adopted to calculate the option price. Compared to frequently encountered problems of calculating the Delta associated with lattice-based approaches due to some localized oscillations in the option values resulting in significantly large errors in the Delta and gamma (see Tavella and Randall [28]),
the direct inversion of an analytic expression of the Delta in the Laplace Transform Method completely avoids this problem.

In the Laplace space, the expression of Delta before numerical inversion can be easily derived by taking the derivative of the option price against the underlying asset value,

\[
\frac{d\bar{U}}{dS} = \begin{cases} 
-q_1 \frac{D(p\bar{S}_f)^{1-q_1}}{q_1 p(p + D)} \cdot S^{q_1 - 1} \\
-\frac{q_1 q_2}{q_1 - q_2} \cdot \left\{ \frac{D(p\bar{S}_f)^{1-q_1}}{q_1 p(p + D)} - \frac{D(p\bar{S}_f)^{1-q_1} + p}{q_2 p(p + D)} - \frac{D - \gamma}{(p + \gamma)(p + D)} \right\} \cdot S^{q_1 - 1} \\
+ \frac{q_1 q_2}{q_1 - q_2} (p\bar{S}_f)^{q_1 - q_2} \cdot \left\{ \frac{D(p\bar{S}_f)^{1-q_2}}{q_1 p(p + D)} - \frac{D(p\bar{S}_f)^{1-q_1} + p}{q_2 p(p + D)} - \frac{D - \gamma}{(p + \gamma)(p + D)} \right\} \cdot S^{q_2 - 1} \\
+ \frac{Dp + \gamma D}{p(p + \gamma)(p + D)} \left( 1 - (p\bar{S}_f)^{q_2 - q_1} \right) S^{q_2 - 1} \\
\text{if } S_f \leq S < 1, \\
-q_2 \frac{D(p\bar{S}_f)^{1-q_1} + p}{q_2 p(p + D)} S^{q_2 - 1} \\
+ \frac{q_1}{q_1 - q_2} (p\bar{S}_f)^{q_1 - q_2} \cdot \left\{ \frac{D(p\bar{S}_f)^{1-q_2}}{q_1 p(p + D)} - \frac{D(p\bar{S}_f)^{1-q_1} + p}{q_2 p(p + D)} - \frac{D - \gamma}{(p + \gamma)(p + D)} \right\} \\
\cdot (1 - (p\bar{S}_f)^{q_2 - q_1}) S^{q_2 - 1} \\
\text{if } S \geq 1.
\end{cases}
\]

Then, this function of \( p \) is numerically inverted as the case for the option price itself.

Fig. 3.3 and Fig. 3.4 show the Delta values computed by the Laplace Transform Method for the case with dividend yield being 0% and 5%, respectively. Other parameters are the same as those in the discussion of numerical accuracy. With the dividend yield being zero, our results obtained through numerical inversion should be very close to those obtained with analytical inversion worked out by Zhu [35]. This is indeed the case as clearly shown in Fig. 3.3. We have also compared our results with those obtained by Wu and Kwok [31], who used a multilevel finite difference scheme to compute the option prices and Delta values for the non-dividend case, and the maximum relative difference between ours and theirs is about 2%. Furthermore, the two solutions become closer as the asset price stays further away from the optimal exercise price. A distinct advantage of our method over Wu and Kwok’s method [31] is that the latter still needs to impose a stability constraint on time and space
step sizes to avoid localized oscillations, whereas we never need to worry about this issue once a robust numerical Laplace inversion such as the Stehfest method has been identified and tested as a suitable method for numerically pricing options and their Deltas.

With the case of non-zero dividend yield, an analytical inversion is no longer possible because $\bar{S}_f(p)$ cannot be explicitly solved from Eq. (3.13). Therefore, we have now compared our results through the numerical Laplace inversion with those of Oosterlee et al. [24], who employed the so-called Grid Stretching Method. In Fig. 3.4, the Delta values with $D_0 = 5\%$ generated by the current method and the Grid Stretching Method are compared. The agreement between the two is excellent, which further reconfirms the reliability of the current method. In fact, as can be observed from Fig. 3.4, our results are actually closer to Oosterlee et al.’s results [24]. Considering that Oosterlee et al. [24] used a fourth-order scheme for both space and time discretization based on a set of stretched grids, we are very pleased to see that our results achieve the same order of accuracy as those of Oosterlee et al. [24], as far as the calculation of Delta is concerned.

So far, the Laplace Transform Method and its numerical inversion method for American put option pricing problems, have been presented. In the following chapter, we are going to introduce a new hybrid finite difference scheme, which promotes a predictor-corrector idea to tackle the moving boundary problem in the valuation of American options.
Fig. 3.3: The Delta Values with $D_0 = 0\%$
Fig. 3.4: The Delta Values with $D_0 = 5\%$
4. THE PREDICTOR-CORRECTOR SCHEME

In the previous chapter, we have introduced the Laplace Transform Method to American option pricing problems. We have shown that the Laplace Transform Method slightly undervalues the optimal exercise boundary, due to the pseudo-steady-state approximation introduced to allow the Laplace transform to be performed on the moving boundary. To find a more accurate and efficient approach for American option pricing problems, we focused our study on the lattice-based approaches in the remaining three months research time (The days of past one year fly on with full research). In this chapter, we shall promote an accurate and efficient scheme that is a new predictor-corrector finite difference scheme.

In the past decade, various numerical methods have been presented by using the finite difference method (FDM), to solve the pricing problems of American options. For instance, Wu and Kwok [31] used a multilevel FDM to solve the nonlinear Black-Scholes PDE, which is explicitly exposed after applying the front-fixing technique [21] to the original Black-Scholes partial differential equation. Pantazopoulos et al. [25] proposed the Front-Tracking FDM, which shows that the optimal exercise boundary can be approximately traced through solving a nonlinear equation at each time step. Halluni et al. [12] derived the option values through using a penalty method in conjunction with the Crank-Nicolson scheme. Ikonen and Toivanen [27] applied an operator splitting method with the finite discretization equations to solve the linear complementarily problem of American options. Zhao et al. [33] used a compact finite difference method to transform the nonlinear Black-Scholes PDE to a set of ordinary differential equations to cope with the movement of this highly nonlinear optimal exercise boundary. More recently, Benth et al. [3] proposed the
finite difference scheme of a predictor-corrector type technique to solve a semilinear Black-Scholes equation. Cho et al. [10] used a set of linearized equations with a function space parameter estimation FDM to solve the option values and optimal exercise prices. Among these numerical methods, most of them can not carry out an accurate solution without sacrificing the computational efficiency. For example, the penalty method [12], the parameter estimation method [10], and the predictor-corrector method [3] need iterations at each time step for more accurate results at the expenses of lower efficiency.

Wu and Kwok [31] adopted a so-called front-fixing technique or Landau transform [21] to fix the optimal exercise boundary on a vertical axis. To solve the nonlinearity of the Black-Scholes PDE, which is explicitly exposed after applying the front-fixing transformation [21] to the original Black-Scholes PDE, Wu and Kwok [31] adopted a two-level discretization scheme in time. However, since the scheme is a multilevel discretization, the information at more than one time step is needed at the beginning to start the computation. In the literature, this problem refers to the initialization for multilevel schemes [29]. Wu and Kwok [31] resorted to a special technique at the first time step to obtain the extra information for the initialization. Their multilevel scheme motivated us to consider a simpler version, which maintains the same level of computational accuracy.

To avoid the initialization problem, we propose a one-step scheme based on the new prediction-correction framework to solve the nonlinear PDE. In our new approach, we adopt a predictor-corrector finite difference scheme at each time step to convert the nonlinear PDE to two linearized difference equations associated with the prediction and correction phase, respectively. The predictor is constructed by an explicit Euler scheme, whereas the corrector is designed with the Crank-Nicolson scheme. The predictor is used only to calculate the optimal exercise price, as the literature shows that it is far more difficult to calculate the optimal exercise price with a high accuracy. With a predicted optimal exercise price, which is then corrected in the correction phase together with the calculation of the option price, our scheme
maximizes the use of computational resources, as high accuracy of the computed option price is easy to achieve as long as a high accuracy can be achieved in the computation of the optimal exercise price. The efficiency is the fact that only one set of linear algebraic equations needs to be solved at each time step in our scheme.

The chapter is organized as follows. Section 4.1 introduces the front-fixing transform to American option pricing problems. Section 4.2 presents the new predictor-corrector scheme used to obtain the optimal exercise prices and the option values. In Section 4.3, some numerical examples are given to demonstrate the validity as well as the order of convergence of the scheme. Furthermore, the issues of computational efficiency, numerical accuracy and the calculation of Greeks, are also discussed in Section 4.3, through showing our numerical solutions in the comparison with those from other option pricing methods.

4.1 The Front-Fixing Transform

If we directly employ the FDM to solve the differential system Eq. (2.25), some kind of iterative schemes such as the projected SOR method described in Wilmott et al. [30] must be used because the differential system Eq. (2.25) is a nonlinear system. To avoid using iterations, Wu and Kwok adopted the Landau transform [21] to convert the problem into a fixed-boundary problem before they employ a multilevel finite difference scheme to solve the resulting system. We followed the same idea before applying our predictor-corrector scheme. Hereafter, we introduce the Landau transform,

\[ x = \ln \frac{S}{S_f(\tau)} \]  

(4.1)
to our differential system Eq. (2.25), which leads to

$$\begin{align*}
\frac{\partial P}{\partial \tau} - \frac{\partial^2 P}{\partial x^2} - (\gamma - D - 1) \frac{\partial P}{\partial x} + \gamma P &= \frac{\partial P}{\partial x} \frac{1}{S_f(\tau)} \frac{dS_f(\tau)}{d\tau}, \\
P(0, \tau) &= 1 - S_f(\tau), \\
\frac{\partial P}{\partial x}(0, \tau) &= -S_f(\tau), \\
\lim_{x \to \infty} P(x, \tau) &= 0, \\
P(x, 0) &= 0.
\end{align*}$$

(4.2)

The domains of the variables in Eq. (4.2) are the same as those in Laplace Transform section, except $x \in [0, +\infty)$. After this rather simple manipulation, the nonlinear nature of the problem is explicitly exposed in the right hand side of the PDE in Eq. (4.2), which consists the product of the Delta of the unknown option price under the Landau transform, the time derivative of the unknown nonlinear optimal exercise boundary $S_f(\tau)$ and its reciprocal.

One should note that we have replaced the unknown function $V(S, \tau)$ in Eq. (2.25), with a new unknown function $P$, which is defined as $P(x, \tau) = V(S(x, (\tau)), \tau)$ through the transform defined in Eq. (4.1). This is to facilitate the introduction of a relation between $P(0, \tau)$ and the $S_f(\tau)$ on the boundary $x = 0$, which is used to design the predictor of our numerical scheme and will be described in the next section.

One should also note that the transform in Eq. (4.1) only holds if $S_f(\tau) > 0$. This condition poses no problem since Samuelson [26] showed that the $S_f(\tau)$ for an American put option is a monotonically decreasing function of $\tau$; the lowest boundary of $S_f(\tau)$ is the optimal exercise price of the corresponding perpetual contract. For a perpetual American put on a constant dividend yield paying asset, this value was shown by Kim [19] as

$$\lim_{\tau \to \infty} S_f(\tau) = \frac{\eta + \sqrt{\eta^2 + 4\gamma}}{2 + \eta + \sqrt{\eta^2 + 4\gamma}},$$

(4.3)

with $\eta = \gamma - D - 1$. It is then very trivial to show that $S_f(\tau) > 0$ for any $\eta$ values.
Therefore, the differential system Eq. (4.2) defines a well-posed problem, other than a well-known singular point at \( \tau = 0 \) (see Barles et al. [2]). We now propose an efficient and accurate numerical scheme to solve this system.

### 4.2 The Predictor-Corrector FDM Scheme

In this section, our predictor-corrector scheme is presented to solve the American option pricing problems. We propose to solve the nonlinear PDE in the differential system Eq. (4.2) in two phases within a time step; a prediction phase, in which a rough guess of the \( S_f(\tau) \) is worked out before its final value is calculated together with the option value \( P(x, \tau) \) in the correction phase of the scheme.

We begin with truncating the semi-infinite domain \( S \in [S_f, \infty) \) to a finite domain \( x \in [0, x_{\text{max}}] \) so that our scheme can be realized in a computer. Guided by Wilmott et al.’s estimate [30] that the upper bound \( S_{\text{max}} \) does not have to be too large, typically about three or four times of the value of the strike price, we should set \( x_{\text{max}} = \ln(6) \) so that the underlying asset price is 5 times of the optimal exercise price, in order to provide a safe guard to ensure that the underlying asset price is about three or four times of the strike price.

Then, the computational domain is discretized with uniformly spread \( M + 1 \) grid points placed in the \( x \) direction and \( N + 1 \) equal time steps in the \( \tau \) direction (\( M \) and \( N \) are two positive integers controlling the step sizes in these two directions respectively). For the easiness of presentation, we shall denote the step length in the \( x \) direction as \( \Delta x = \frac{x_{\text{max}}}{M} \) and that in the \( \tau \) direction as \( \Delta \tau = \frac{\tau_{\exp}}{N} \), in which \( \tau_{\exp} \) is the normalized tenor of the contract with respect to half of the variance of the underlying asset, i.e., \( \tau_{\exp} = T \sigma^2 / 2 \). Consequently, the value of unknown function \( P \) at a grid point is denoted by \( P^m_n \) with the superscript \( n \) denoting the \( n \)th time step and the subscript \( m \) denoting the \( m \)th log-transformed asset grid point.

To facilitate the numerical computation, we derive an additional boundary condition to construct our predictor-corrector scheme. This condition is not independent
from all those boundary conditions prescribed in Eq. (4.2). Rather, it is derived by making use of the PDE in Eq. (4.2) as well as the boundary conditions that have already made the system closed. Firstly, we take a partial derivative with respect to \( \tau \) on both sides of the first boundary condition in Eq. (4.2), which yields

\[
\frac{\partial P}{\partial \tau}(0, \tau) = -\frac{dS_f(\tau)}{d\tau}. \tag{4.4}
\]

In fact, one can easily show that Eq. (4.4) is consistent with the condition

\[
\frac{\partial V}{\partial \tau}(S_f(\tau), \tau) = 0 \text{ in Bunch and Johnson’s paper [6].}
\]

Then, if we evaluate the PDE in Eq. (4.2) at \( x = 0 \), utilizing Eq. (4.4) and the second boundary condition in Eq. (4.2), we obtain

\[
-\frac{\partial^2 P}{\partial x^2} |_{x=0} - (D + 1)S_f(\tau) + \gamma = 0, \quad \text{if } \tau > 0. \tag{4.5}
\]

Eq. (4.5) reveals a relation of the put option price and the optimal exercise price at any time, except on the expiry day. This relation is very useful in our scheme to eliminate the value of the unknown function defined on the fictitious grid point near the boundary \( x = 0 \). The reason that it is only valid for \( \tau > 0 \) is the inherent singular behavior of the Black-Scholes PDE at \( \tau = 0 \) (see Barles et al. [2]).

Now, if a second-order central difference scheme is adopted for the asset price discretization in the \( x \) direction, Eq. (4.5) and the boundary conditions in Eq. (4.2) can be written as

\[
-\frac{P_{n+1}^0 - 2P_{n+1}^0 + P_{n+1}^{-1}}{\Delta x^2} - (D + 1)S_f^{n+1} + \gamma = 0, \tag{4.6}
\]

and

\[
\begin{align*}
P_0^{n+1} &= 1 - S_f^{n+1}, \\
\frac{P_1^{n+1} - P_{n-1}^{n+1}}{2\Delta x} &= -S_f^{n+1}, \\
\frac{P_M^{n+1}}{} &= 0, \\
P_m^n &= 0,
\end{align*} \tag{4.7}
\]
respectively. Making use of the first two discretized boundary condition in Eq. (4.7), we can eliminate $p_0^{n+1}$ and $p_{-1}^{n+1}$ to obtain the following relationship between $S_f$ and $P_1$ at the $(n+1)th$ time step as

$$P_1^{n+1} = \alpha - \beta S_f^{n+1},$$

(4.8)

in which $\alpha = 1 + \frac{\gamma}{2} \Delta x^2$ and $\beta = 1 + \Delta x + \frac{D + 1}{2} \Delta x^2$. Eq. (4.8) will be used to construct both the predictor and the corrector.

**Predictor:**

Our predictor is constructed by using the explicit Euler scheme to calculate a guessed value of $S_f^{n+1}$, which is denoted by $\hat{S}_f^{n+1}$. In the following content, we use the symbol ‘hat’ to denote the value as predicted one. Applying the explicit Euler scheme to the PDE in Eq. (4.2) results in

$$\frac{\hat{P}_1^{n+1} - P_1^n}{\Delta \tau} = \frac{P_2^n - 2P_1^n + P_0^n}{\Delta x^2} - \frac{(\gamma - D - 1)P_2^n - P_0^n}{2 \Delta x} + \gamma P_1^n = \frac{P_2^n - P_0^n}{2 \Delta x} \frac{1}{S_f^n} \frac{\hat{S}_f^{n+1} - S_f^n}{\Delta \tau},$$

(4.9)

which is coupled with Eq. (4.8) to generate the $\hat{S}_f^{n+1}$ value.

The boundary condition of $\hat{P}_0^{n+1}$ used in the corrector is also predicted here; with the calculated $\hat{S}_f^{n+1}$ value, $\hat{P}_0^{n+1}$ is calculated from the first equation in Eq. (4.7), which is nothing but the payoff function. Like the predicted $\hat{S}_f^{n+1}$ value, this predicted boundary value of $\hat{P}_0^{n+1}$ will also be corrected once the $\hat{S}_f^{n+1}$ is corrected in the following corrector scheme.

**Corrector:**

Our corrector is based on the Crank-Nicolson scheme, applied to linearize the PDE in Eq. (4.2). The linearization is designed with an alternating term being valued at the current time step in comparison with that in the predictor. In the latter, we let the time derivative of the $S_f$ in the nonlinear inhomogeneous term be valued at the current time step, whereas now we let the $x$ derivative of $P$ be valued
at the current time step through the Crank-Nicolson scheme. This approach has an advantage of reducing the numerical errors induced in the prediction-correction process. The finite difference scheme used for the corrector is

\[
\frac{P_{m}^{n+1} - P_{m}^{n}}{\Delta \tau} - \frac{P_{m+1}^{n+1} - 2P_{m+1}^{n} + P_{m+1}^{n-1}}{2\Delta x^2} - (\gamma - D - 1) \frac{P_{m+1}^{n+1} - P_{m}^{n+1} + P_{m+1}^{n} - P_{m}^{n}}{4\Delta x} + \gamma \frac{P_{m+1}^{n+1} + P_{m}^{n}}{2} = \frac{\hat{S}_{f}^{n+1} - S_{f}^{n}}{\Delta \tau}.
\] (4.10)

In Eq. (4.10), \(m\) is from 1 to \(M - 1\), which means that \(M - 1\) equations are solved simultaneously to obtain the corrected option values at the \((n + 1)th\) time step. \(P_{1}^{n+1}\) is obtained upon solving Eq. (4.10). Then, by means of Eq. (4.8), the newly-obtained \(P_{1}^{n+1}\) is used to correct the \(S_{f}^{n+1}\), which is then used to correct the \(P_{0}^{n+1}\) value before it is used in the calculation of the next time step.

In a much more condensed way, Eq. (4.10) can be written in the following matrix form:

\[
AP_{m}^{n+1} = BP_{m}^{n} + e,
\] (4.11)

with

\[
P_{m}^{n+1} = (P_{1}^{n+1}, P_{2}^{n+1}, \cdots, P_{M-1}^{n+1})^T,
\]

\[
P_{m}^{n} = (P_{1}^{n}, P_{2}^{n}, \cdots, P_{M-1}^{n})^T,
\]

\[
e = (a(P_{0}^{n} + \hat{P}_{0}^{n+1}), 0, \cdots, 0, c(P_{M}^{n} + P_{M}^{n+1}))^T.
\]
The coefficients in the matrices are

\[
A = \begin{bmatrix}
  b & -c & 0 & 0 & 0 & \cdots & 0 \\
  -a & b & -c & 0 & 0 & \cdots & 0 \\
  0 & -a & b & -c & 0 & \cdots & 0 \\
  0 & \ddots & \ddots & \ddots & \ddots & \ddots & 0 \\
  \vdots & \hdots & -a & b & -c \\
  0 & \hdots & -a & b
\end{bmatrix}, \quad
B = \begin{bmatrix}
  b' & c & 0 & 0 & 0 & \cdots & 0 \\
  a & b' & c & 0 & 0 & \cdots & 0 \\
  0 & a & b' & c & 0 & \cdots & 0 \\
  0 & \hdots & a & b' & c & \cdots & 0 \\
  \vdots & \hdots & \hdots & \hdots & \hdots & \ddots & 0 \\
  0 & \hdots & \hdots & \hdots & \hdots & a & b' \\
  0 & \hdots & \hdots & \hdots & \hdots & a & b'
\end{bmatrix}.
\]
4. The Predictor-Corrector Scheme

\[
\begin{align*}
    a &= \frac{\Delta \tau}{2\Delta x^2} - \frac{\gamma - D - 1}{4} \frac{\Delta \tau}{\Delta x} - \frac{1}{2\Delta x} \hat{S}_{f}^{n+1} - S_{f}^{n}, \\
    b &= 1 + \frac{\Delta \tau}{\Delta x^2} + \frac{\gamma}{2} \Delta \tau, \\
    b' &= 1 - \frac{\Delta \tau}{\Delta x^2} - \frac{\gamma}{2} \Delta \tau, \\
    c &= \frac{\Delta \tau}{2\Delta x^2} + \frac{\gamma - D - 1}{4} \frac{\Delta \tau}{\Delta x} + \frac{1}{2\Delta x} \hat{S}_{f}^{n+1} - S_{f}^{n}.
\end{align*}
\]

We solve these matrix equations in Matlab, Version 6.

This predictor-corrector process is repeated until the expiry time is reached. In order to provide a comprehensive overview of the entire structure of the proposed predictor-corrector scheme, a schematic flow chart of the scheme is shown in Fig. 4.1. In the following section, we present some numerical study about the current difference scheme.

4.3 Numerical Examples

In the previous section, we have introduced our predictor-corrector finite difference scheme to the American option pricing problems. To characterize what the new predictor-corrector scheme does, we need to address the following issues. First of all, we concern whether the solution of the our scheme converges to the original PDE’s solution, since the PDE in Eq. (4.2) before applying the new scheme is nonlinear, whereas our predictor-corrector framework is linearized one. Secondly, it would be interesting to examine the order of convergence in both the \( \tau \) and \( x \) direction, since our difference scheme is a hybrid scheme which is constructed by the explicit Euler scheme and the Crank-Nicolson scheme. Moreover, as we mentioned before, most of the finite difference methods can not carry out an accurate solution without sacrificing the computational efficiency. We shall examine the computational efficiency and numerical accuracy of the current scheme. Furthermore, a criteria to evaluate a good finite difference scheme in option pricing problems, is whether the scheme can derive nice solutions of the option derivatives. We will compare the Delta values derived by the current scheme with those of other option pricing methods to show
the validity in computing the option derivatives. In the following subsections, we will address these issues through the discussion on a couple of examples.

To ensure the convergence of the current approach, a constraint \( \frac{\Delta \tau}{\Delta x^2} \leq 1 \) is established empirically through enormous numerical experiments. All the experiments for the above issues are imposed with the constraint.

We use an example which has been used by many researchers for the discussion of American puts on an asset without any dividend payment [7, 31, 34]. Firstly, to discuss the validity of the current method, we set the constant dividend yield \( D_0 \) to zero, through showing the agreement between the results obtained by the current approach and those obtained by Zhu’s analytical solution [34]. Then, we set \( D_0 \) to a constant for the discussion in the constant dividend yield case, with showing the numerical solutions of the optimal exercise boundary, the option values and the Delta values in the comparison with those from the Laplace Transform Method [40] and the Grid Stretching Method [24] respectively.

4.3.1 Discussion on Validity

To valid our new predictor-corrector scheme, we plot our numerical solution and Zhu’s analytical solution [34] in a same diagram to show the agreement between the two solutions. Zhu already proved his approach which is an exact analytical method for American put option pricing problems in non-dividend case [34]. With knowing the exact solution [34] in non-dividend case, we can directly compare our solution with this analytical solution, to show whether our numerical solution converges to Zhu’s analytical solution [34]. Moreover, the property that the minimum value of the \( S_f(\tau) \) for an American put option is the one of the corresponding perpetual contract [19], is verified in this subsection as well.

The validity discussion is based on the study of the optimal exercise boundary. The option parameters for computing the \( S_f(\tau) \) in this subsection are: the strike price \( X = $100 \), the interest rate \( r = 10\% \), the volatility of the underlying asset \( \sigma = 30\% \) and the tenor of the option being 1 year for the comparison with Zhu’s
analytical solution [34], and being 20 years in the perpetual case respectively. We set $x_{\text{max}}$ to 3 and 7 for the two different life time cases respective, to have a large enough underlying asset price to satisfy the fourth condition in Eq. (2.25). For the convenience of those readers who prefer to see the results in dimensional form, all results presented in this subsection are those associated with the original dimensional quantities before the normalization process was introduced.

To show that our numerical solution converges to Zhu’s analytical solution [34], we plot the solutions in Fig. 4.2, in which the optimal exercise boundaries are computed by using the current scheme with $N = 600$, $M = 200$ and Zhu’s analytical method [34] respectively. As it can be seen from Fig. 4.2, the two curves match nicely. However, the current approach seems to slightly underestimate the $S_f$ values when the time closes to expiry day. The magnitude of the under-estimate varies with the time to expiry. The under-estimate is roughly 2.16% (relative error) at the time to expiry $(T - t)$ being 0.0067 year, whereas it reduces down to 0.05% at the
time to expiry being 1 year. The reason why the magnitude of the under-estimate is much higher around $\tau = 0$, is the well-known singular point at $\tau = 0$ (see Barles et al. [2]). It is hard that numerical methods can carry out a precise value around the singular point. We are pleased by the result that such a small error 2.16% can be achieved through employing our new scheme while the time to expiry closes to the singular point $\tau = 0$.

Some approximate approaches can derive a correct solution for short-term options but exhibit problems when the lifespan of an option is long; for example, the optimal exercise price discussed by Barone and Elliott [13] becomes non-monotonic and does not asymptotically approach the perpetual optimal exercise price [19]. Therefore, we use a long-term put option with a lifetime of 20 years in this example, to demonstrate that the new scheme works as well when $\tau$ is large. Furthermore, through showing the example, we can verify whether the optimal exercise boundary from our approach satisfies Eq. (4.3) as well. We plot the $S_f(\tau)$, computed by the current scheme with $N = 102,400$, $M = 1,000$, and the perpetual optimal exercise price [19] in Fig. 4.3. It is trivial to show that Eq. (4.3) reduces to $\frac{\gamma}{1+\gamma}$ when the constant dividend yield is zero. The perpetual optimal exercise price is $68.97$ in this particular non-dividend case. As clearly shown in Fig. 4.3, the numerical solution asymptotically approaches the perpetual optimal exercise price; it reaches the perpetual optimal exercise price at the time to expiry being 14 years in this case, and the optimal exercise boundary never falls below the perpetual optimal exercise price which should provide a lower bound [19] for the $S_f$s of all corresponding short term options.

4.3.2 Discussion on Order of Convergence

It would be interesting to study the order of convergence for the current difference scheme in both the $\tau$ and $x$ direction, since the current difference scheme is a hybrid scheme, which is constructed by the explicit Euler scheme and the Crank-Nicolson scheme. Theoretically speaking, we should have a second-order convergence in the
4. The Predictor-Corrector Scheme

$x$ direction since we adopt the central finite difference scheme; for the $\tau$ direction, we should have a first-order convergence since the predictor uses the explicit Euler scheme. We study the order of convergence in time for the current scheme according to the following procedure. Firstly we calculate the difference of the $S_f$ values at the time to expiry being 1 year by halving the time step size and fixing the grid step size; the initial time step number for the time step refinement is 200, with the fixed grid intervals being 50. Then we calculate the ratios of the successive differences of the $S_f$s; the ‘difference’ refers to the absolute value of the change in the $S_f$s as the time step size is refined, while the ‘ratio’ refers to the ratio of the successive differences. Finally, the order of convergence can be addressed through the relation $ratio = 2^{(\text{order})}$.

The study of the order of convergence in the $x$ direction follows the same idea in the $\tau$ direction, we still compute the $S_f$ values, with halving the grid size in the $x$ direction and fixing the time step size. The initial grid intervals number for the

Fig. 4.3: The Numerical Optimal Exercise Boundary in Perpetual Case
grid size refinement is 50, with the fixed time steps number being 1,600. The data of the two experiments are listed in the following tables.

Table 4.1 and Table 4.2 list the \( S_f \) values, the differences and ratios of our study in the \( \tau \) and \( x \) direction respectively. The ratios in Table 4.1 and Table 4.2 are all around 2 and 3 respectively, which suggest a first order of convergence in both the \( x \) and \( \tau \) direction, or \( o(\Delta x^2) \) and \( o(\Delta \tau) \).

So far, we have discussed the validity and the convergence property for the current method through the study on the \( S_f(\tau) \). In the next subsection, we shall discuss the issues that are computational efficiency and the numerical accuracy of the current method.

### 4.3.3 Discussion on Accuracy and Efficiency

In finance industry, quite often, the speed of calculation is of equal importance to the accuracy, although it is generally a rule of thumb that efficiency is inversely proportional to accuracy. In other words, when one wishes to achieve a high computational efficiency, normally he has to sacrifice the accuracy to a certain degree. The key question is however if one can achieve a high efficiency at a reasonably satisfactory accuracy.
As we mentioned before, most of the finite difference schemes have to face a tradeoff between the computational efficiency and the numerical accuracy. An advantage of our scheme is that the current approach can achieve an accurate solution with using very few time steps and grid numbers. This feature indicates that our predictor-corrector scheme has a high computation efficiency. To exactly examine the numerical accuracy of the scheme, we use the relative Root Mean Squared (RMS) error to measure the difference between our numerical solution and Zhu’s analytical solution [34]. And we use the CPU time to measure the current approach’s computational efficiency. All the experiments are performed on an Intel Pentium 4, 3GHz machine with Matlab 6, adopting long numeric format (15-digit scaled fixed point).

In order to study the computational efficiency (relating to \( N \) and \( M \)) and the numerical accuracy for the current scheme, we measure the relative RMS of the \( S_f(\tau) \)s computed by our new scheme in the following procedure. Firstly we compute the numerical optimal exercise boundaries with the time step \( N \) and grid number \( M \) being successively increasing by 1 respectively, both the initial \( N \) and \( M \) being 10. Secondly we measure the differences between these numerical solutions and Zhu’s analytical solution [34], to know the behavior of the relative RMS with the increasing in \( N \) and \( M \). Thirdly, we plot the truncation errors with the corresponding \( N \)s and \( M \)s in a 3-D diagram, to have an idea that what scale of \( N \) and \( M \) are large enough to carry out a low relative RMS region.

In order to measure the relative RMS of the whole \( S_f(\tau) \) boundary rather than on a single time instant, we sample Zhu’s analytical solution and the numerical solutions, by taking the \( S_f \)s, with equal temporal interval size. For a numerical solution \( S_f \) computed by the current scheme with a particular \( N \) and \( M \), the definition of its relative RMS is

\[
RMS = \sqrt{\frac{1}{T} \sum_{i=1}^{T} (\tilde{a}_i - a_i)^2},
\]

in which \( a_i \)s are the samples from Zhu’s analytical solution [34], and \( \tilde{a}_i \)s are the samples from our numerical solutions computed with different \( N \)s and \( M \)s; the
range of $N$ and $M$ being from 10 to 200 and from 10 to 100, respectively. $I$ is the sample number, being 50 in our experiments. For each numerical solution, we use the first order interpolation to derive the sample value $\tilde{a}_i$ on a same time instant $\tau$ of the sample value $a_i$.

As Fig. 4.4 shows, the relative RMS is reduced steadily with the increasing $N$ and $M$. According to the data from our experiments, we found that the relative RMS has been reduced from 3.794% with $N = 10$ and $M = 10$ to 0.241% with $N = 200$ and $M = 100$. To achieve such a low relative RMS as 0.241%, which is far more less than 1%, we employed our scheme with only taking the time steps and the grid intervals $N = 200$ and $M = 100$ respectively; the consumed CPU time for implementing the whole calculation was only 0.79 seconds. From Fig. 4.4, we can know that there is a low relative RMS region from $N = 100$ and $M = 60$, i.e., that the current approach can guarantee a low truncation error with using very few time steps number $N$ and grid intervals number $M$.

Fig. 4.5 shows the relative RMS as a function of total CPU time used in executing
the code for each run of different $N$ and $M$. The CPU times and the relative RMS are corresponding to $N$ values being 10, 15, 30, 50, 100, 150 and 200, with $M$ being 60, 80, 100 respectively. As clearly shown in Fig. 4.5, the accuracy is in general inversely varying with the efficiency; a higher accuracy usually implies a lower efficiency for any grid resolution. If we concern the computational efficiency, i.e., let's only consider the runs which’s CPU times are all less than 0.1 seconds, among those tests, the relative RMS being 1.65% derived by $M = 60$ with $N = 50$, is the lowest one, with using only 0.09 seconds CPU time. On the other hand, if we concern the numerical accuracy, i.e., among the runs which’s relative RMS are less than 1%, the test with using $M = 100$ and $N = 150$ achieves the lowest relative RMS being 0.14% with only taking 0.56 seconds. The relative RMS reduced nearly 10 folds, with 6 times in the increase of the execution CPU time, which not only demonstrates the current scheme’s high efficiency but also confirms the inverse relationship of accuracy and efficiency in our scheme.
From Fig. 4.5, one might note that the relative RMS does not always decrease with the increase in $N$; we can easily examine this phenomenon while $M$ is 60 with $N$ being 100, 150 and 200; the relative RMS computed with using $N = 200$ is apparently higher than the one computed with using $N = 150$. This is because that the relative RMS becomes quite high while either $M$ or $N$ close to 0, but becomes very low while both $M$ and $N$ are very small. In fact, for each $M$ value, there exists a range of $N$ values (we do not care the range of these $N$ values), which ensures a lowest relative RMS of the numerical solution, if $N$ becomes larger or smaller than this specific range, the relative RMS becomes higher. However, the range of the $N$ for a lowest relative RMS, extends with the increase in $M$, we do not worry about the range as we only consider the large values in $M$, like $M = 200$. This characteristic of the scheme can be easily examined if we review the previous 3-D plot.

Last but not least, from Fig. 4.5, we can know that all the CPU times for carrying out the low relative RMS that are far more less than 1%, are all less than 1 second. Apparently the new predictor-corrector scheme is a highly efficient approach and it can achieve satisfactory accuracy.

So far, we have shown the computational efficiency and numerical accuracy of the current approach. In the following subsection, we shall briefly present the optimal exercise prices, option values and Delta values which are computed by the current scheme with $D_0$ being a constant, in the comparison with the solutions of other option pricing methods, to show the validity of the current approach in the constant dividend yield case.

### 4.3.4 The Constant Dividend Yield Case

So far, the discussion we have presented in the previous subsections, are all based on the optimal exercise boundary. The reason we have focused on the validity, accuracy and efficiency for the calculation of $S_f(\tau)$ rather than the option value, is that it is well understood that for the valuation of American options, the key issue
is the calculation of the optimal exercise price, $S_f(\tau)$; once the $S_f(\tau)$ is determined accurately, the calculation of the option price itself is straightforward (see Huang et al. [17]). However, for the completeness of the option pricing problems, we present the discussion of the option values, Delta values as well as the optimal exercise prices through showing a constant dividend case in this subsection. We use the same parameters in the discussion of non-dividend case, except $D_0$ being 5% now. The solutions are derived by the current scheme with $N = 200$ and $M = 100$. The comparisons of the option values and Delta values are presented in the following two figures.

We plot the option values in Fig. 4.6, which shows a comparison of the option values calculated by the current approach and those obtained by Oosterlee et al. [24], who employed the so-called Grid Stretching Method. Also, the calculation of the Greeks can be easily implemented once the option values, the underlying asset prices and the time to expiry are converted back to the dimensional quantities.
 Unlike the significant large errors in calculating the Greeks such as Delta associated with some lattice-based approaches due to the localized oscillations in the option values [28], our approach can carry out the accurate Delta values. Fig. 4.7 shows the comparison of the Delta values calculated by our scheme and those obtained by Oosterlee et al. [24].

As we can observe from Fig. 4.6 and Fig. 4.7, our results demonstrate the perfect agreement with those of Oosterlee et al. [24] in the computation of both option values and Delta values. Considering the Grid Stretching Method [24] is a fourth-order scheme for both the $\tau$ and $x$ discretizations based on a set of stretched grids, such excellent agreement not only validates the accuracy of our scheme but also demonstrates its efficiency; a lower-order scheme has virtually achieved the same level of accuracy as a higher-order scheme does.

We compare our numerical solution of the optimal exercise boundary, with the one of another American puts pricing method that is the Laplace Transform Method.
In Fig. 4.8, as Oosterlee et al. [24] did not present the discussion on the optimal exercise boundary. As Fig. 4.8 shows, the solution of our new predictor-corrector scheme is above the solution of the Laplace Transform Method [40]. The difference between the two solutions is reasonable as we have shown that the Laplace Transform Method underestimates the optimal exercise price due to the pseudo-steady-state approximation in Chapter 3. In Fig. 4.8, the difference between the two solutions is around 2% at the time to expiry \((T - t)\) being 1 year. The under-estimate of the Laplace Transform Method is also around 1-2% in the comparison with the analytical solution [34] as we have demonstrated in the paper [40]. According to this same under-estimate magnitude, we can judge that our numerical solutions are valid as well in the constant dividend yield case, although we have already shown the validity of the current scheme in the constant dividend case through the discussion on the option values and the Delta values.
5. CONCLUSION

In this thesis, we have discussed two different numerical methods to value American put option on a constant dividend yield paying asset, in terms of solving the Black-Scholes partial differential equation with a set of appropriate boundary conditions. According to the discussion, we have found that both approaches are accurate and efficient for pricing the financial derivative with American-style exercise feature.

The first method is the Laplace Transform Method, which extends the pseudo-steady-state approximation idea for the American option pricing problems in non-dividend yield case [35] to the one in constant dividend yield case. Numerical inversion becomes necessary since analytical expressions of the option value and the optimal exercise price can not be found in the Laplace space. We have thus tested three numerical Laplace inversion methods, and have found that the Stehfest method with parameter $N$ being 8 is recommended as the most suitable method among the three methods for the option pricing problems, according to its numerical accuracy, efficiency as well as the easiness of implementation. Although the Laplace Transform Method slightly undervalues the optimal exercise prices due to the Pseudo-Steady-State approximation, the loss of the accuracy in this regard is greatly compensated by its high computational speed.

The second method promotes a new predictor-corrector idea that uses a hybrid finite difference scheme to tackle the nonlinear nature of American option pricing problems, which is explicitly exposed after applying the front-fixing technique [21] to the original Black-Scholes partial differential equation. The key features of the current scheme are its high efficiency since there is no iteration nor initialization required, and its high accuracy as the scheme reduces the numerical errors induced
in the prediction-correction process. We have shown that the numerical results obtained from the predictor-corrector scheme converge uniformly to the exact optimal exercise boundary and option values. However, a constrain for the convergence has to be imposed on the scheme, due to the nature of the lattice-based approaches.
Definitions and Background Information

Liquidity: The ability of an asset to be converted into cash quickly and without any price discount.

Liquidation: In finance, liquidation is also sometimes used as convenient shorthand for converting an asset to cash.

In-the-money: Situation in which an option’s strike price is below the current market price of the underlier (for a call option) or above the current market price of the underlier (for a put option). Such an option has intrinsic value.

At-the-money: A condition in which the strike price of an option is equal to (or nearly equal to) the market price of the underlying security.

Out-of-the-money: A call option whose strike price is higher than the market price of the underlying security, or a put option whose strike price is lower than the market price of the underlying security.
Greeks: In mathematical finance, the Greeks are the quantities representing the market sensitivities of options or other derivatives. Each "Greek" measures a different aspect of the risk in an option position, and corresponds to a parameter on which the value of an instrument or portfolio of financial instruments is dependent. The name is used because the parameters are often denoted by Greek letters.

Option Delta: is the change in the price of an option for a one point moves in the underlying. Call options: $0 < \text{Option Delta} < 1$
Put options: $-1 < \text{Option Delta} < 0$
In-the-money options: Delta Option approaches 1 (call:+1, put:-1)
At-the-money options: Delta is about 0.5 (call:+0.5, put: -0.5)
Out-of-the-money options: Delta Option approaches 0
Call Option Delta can be interpreted as the probability that the option will finish in the money. An at-the-money option, which has a delta of approximately 0.5, has roughly a 50/50 chance of ending up in-the-money. Put Option Delta can be interpreted as $-1$ times the probability that the option will finish in the money.

Laplace Transform: The Laplace transform is an integral transform perhaps second only to the Fourier transform in its utility in solving physical problems. The Laplace transform is particularly useful in solving linear ordinary differential equations such as those arising in the analysis of electronic circuits.

Predictor-Corrector Method: A general set of methods for integrating ordinary differential equations. Predictor-corrector methods proceed by extrapolating a polynomial fit to the derivative from the
previous points to the new point (the predictor step), then using this to interpolate the derivative (the corrector step). Press et al. (1992) opine that predictor-corrector methods have been largely supplanted by the Bulirsch-Stoer and Runge-Kutta methods, but predictor-corrector schemes are still in common use.

Binomial Method: In finance, the binomial options pricing model (BOPM) provides a generalisable numerical method for the valuation of options. The binomial model was first proposed by Cox, Ross and Rubinstein (1979). Essentially, the model uses a "discrete-time" model of the varying price over time of the underlying financial instrument. Option valuation is then computed via application of the risk neutrality assumption over the life of the option, as the price of the underlying instrument evolves.

Pseudo steady state assumption: this assumption allows you to model a small portion of an extremely complex system. Simply put, without this assumption many models would not exist and it allows us to work with a system that has both fast and slow reactions. If you are interested in the portion of the model that contains the slow, or rate limiting reactions, you can (sometimes) assume that the fast reactions are in a state of dynamic equilibrium, and the their derivatives are equal to zero, compared to the slow reactions. If you are interested in the portion of the model that has fast reactions, you can assume that the the slow portion does not change significantly (and thus, it’s derivative is zero) when compared to the fast.
Maple Code: the Optimal Exercise Boundary

restart;
stt:=time():
n:=10;
r:=0.1;
d:=0;
sigma:=0.3;
rg:=2*r/(sigma^2);
dg:=2*d/(sigma^2);
b:=(1+dg-rg)/2;
b2:=b^2;
logtwo:=evalf(log(2));

ste:=proc(t) local ft,i,p,q1,q2,fs,cn,c,k,cnn,IV;
ft:=0;
for i from 1 to n do
  p:=i*logtwo/t;
  q1:=b+sqrt(b2+rg+p);
  q2:=b-sqrt(b2+rg+p);
  nonlin:=sf^q1*(rg+p-rg*q2+dg*q2)/q2/(rg+p)/(dg+p)+sf*dg*(1-q2)/p^(1+q1)/(rg+p);
  fs:=fsolve(nonlin,sf);
  cn:=0:
  c:=0:
  for k from floor(i/2+0.5) to min(i,n/2) do
    cnn:=evalf(k^(n/2)*((2*k)!)/(n/2-k)!*k!*(k-1)!
      (i-k)!*(2*k-i)!));
    cn:=cn+cnn;
end do;
end do;
return fs;
end proc:
st:=time():
st stutter:
for i from 1 to 10 do
  slider:=ste(i);
end do;
end;
c:=evalf((-1)^(i+n/2)*cn);
ft:=ft+c*fs;
end;
IV:=logtwo*ft/t;
return(IV);
end;

stepl:=0.00045;
num:=100;
value:=[seq([0.00001+j*stepl,ste(0.00001+j*stepl)],j=0..num)];
time()-stt;
PLOT(CURVES(evalf(value)));
Maple Code: the Option Value Part A

restart;
n:=6;
r:=0.1;
d:=0;
sigma:=0.3;
rg:=2*r/(sigma^2);
dg:=2*d/(sigma^2);
b:=(1+dg-rg)/2;
b2:=b^2;
logtwo:=evalf(log(2));

stev:=proc(t,s) local ft,i,p,q1,q2,fss,fs,cn,c,k,cnn,IV;
ft:=0;
for i from 1 to n do
  p:=i*logtwo/t;
  q1:=b+sqrt(b2+rg+p);
  q2:=b-sqrt(b2+rg+p);
  nonlin:=sf^q1*(rg+p-rg*q2+dg*q2)/q2/(rg+p)/(dg+p)+sf*dg*(1-q2)
  /p^q1/q2/(dg+p)=rg/p^(1+q1)/(rg+p);
  fss:=p*fsolve(nonlin);
  fa:=(dg*fss^(1-q1)/q1/p/(dg+p)-(dg*fss^(1-q1)+p)/q2/p/(dg+p)
  -(dg-rg)/(rg+p)/(dg+p));
  c1:=-dg*(fss)^(1-q1)/q1/p/(dg+p)-(dg*fss^(1-q1)+p)/q2/p/(dg+p)
  -(dg-rg)/(rg+p)/(dg+p));
  c2:=q1/(q1-q2)*fss^(q1-q2)*fa;
  fs:=c1*s^q1+c2*s^q2+((dg*s-rg)*p+(s-1)*rg*dg)/p/(rg+p)/(dg+p);
  cn:=0;
  c:=0:
end do;
for k from floor(i/2+0.5) to min(i,n/2) do
    cnn:=evalf(k^(n/2)*((2*k)!)/((n/2-k)!*k!*(k-1)!*(
        (i-k)!*(2*k-i)!)));
    cn:=cn+cnn;
end;

c:=evalf((-1)^(i+n/2)*cn);
ifo:=ft+c*fs;
end;
IV:=logtwo*ft/t;
v:=IV-s+1;
return(v)
end;
Maple Code: the Option Value Part B

restart;
n:=6;
r:=0.1;
d:=0;
sigma:=0.3;
rg:=2*r/(sigma^2);
dg:=2*d/(sigma^2);
b:=(1+dg-rg)/2;
b2:=b^2;
logtwo:=evalf(log(2));

stev:=proc(t,s) local ft,i,p,q1,q2,fss,fs,cn,c,k,cnn,IV;

ft:=0;
for i from 1 to n do
    p:=i*logtwo/t;
    q1:=b+sqrt(b2+rg+p);
    q2:=b-sqrt(b2+rg+p);
    nonlin:=sf^q1*(rg+p-rg*q2+dg*q2)/q2/(rg+p)/(dg+p)+sf*dg*(1-q2)/p^q1/q2/(dg+p);
    fss:=p*fsolve(nonlin);
    fa:=(dg*fss^(1-q1)/q1/p/(dg+p)-(dg*fss^(1-q1)+p)/q2/p/(dg+p)-(dg-rg)/(rg+p)/(dg+p))/p^q1/q2/(dg+p);
    c2:=q1/(q1-q2)*fss^(q1-q2)*fa;
    fs:=-(dg*fss^(1-q1)+p)/p/(dg+p)/q2+c2*(1-fss^(q2-q1))*s^q2;
    cn:=0:
    c:=0:
    for k from floor(i/2+0.5) to min(i,n/2) do
cnn := evalf(k^(n/2)*((2*k)!)/((n/2-k)!*k!*(k-1)!*(i-k)!*(2*k-i)!));
cn := cn + cnn;
end;
c := evalf((-1)^(i+n/2)*cn);
ft := ft + c*fs;
end;
IV := logtwo*ft/t;
v := IV;
return(v)
end;
Matlab Code: the Predictor-Corrector Scheme

format long
% 1 option parameters Initialization
r=0.1;
d=0;
T=20;
K=100;
sigma=0.3;
rnd=2*r/(sigma^2);
dnd=2*d/(sigma^2);
tau=T*(sigma^2)/2;
k=K/K;

% 2 Discretization parameters Initialization
xmin=0;
xmax=7;
N=3000;M=400;
dx=xmax/M;
dtau=tau/N;
courant=dtau/(dx^2)

% 3 Initial Condition & Boundary Condition
p=zeros(N+1,M+1);

% 4 critical price
apha=1+rnd*(dx)^2/2;
beta=1+dx+dnd+1*(dx)^2/2;
lp(1)=1;
% 5coefficients of Difference scheme
sita1=1/2/dx/beta;
sita2=dtau/(dx^2);
sita3=(rnd-dnd-1)*dtau/2/dx-1/2/dx;
sita4=apha/2/dx/beta;
sita5=1-2*dtau/(dx^2)-rnd*dtau;

deta1=1/2/dx+(dnd+1)/2;
deta2=1/2/(dx^2);

kesi1=dtau/2/(dx^2);
kesi2=(rnd-dnd-1)*dtau/4/dx;
kesi3=1/2/dx;
kesi4=1+dtau/(dx^2)+rnd*dtau/2;
kesi5=1-dtau/(dx^2)-rnd*dtau/2;

tic

for n=1:N

   % 6 preditor Scheme

   A1=(p(n,3)-p(n,1))*sita1/cp(n);
   A2=sita2+sita3+sita4/cp(n);
   A3=sita2-sita3-sita4/cp(n);
   pp(n+1,2)=(A2*p(n,3)+sita5*p(n,2)+A3*p(n,1))/(1+A1);
   cpp(n+1)=(apha-pp(n+1,2))/beta;
   pp(n+1,1)=k-cpp(n+1);
   cp(n+1)=cpp(n+1);
p(n+1,1)=k-cp(n+1);
\%

7 corrector scheme

B1=kesi1+kesi2+kesi3*(cp(n+1)-cp(n))/(cp(n+1)+cp(n));
B2=kesi1-kesi2-kesi3*(cp(n+1)-cp(n))/(cp(n+1)+cp(n));

for i=1:M-2
    MA1(i,i)=kesi4; MA1(i,i+1)=-B1; MA1(i+1,i)=-B2;
    MA2(i,i)=kesi5; MA2(i,i+1)=B1; MA2(i+1,i)=B2;
end;

MA1(M-1,M-1)=kesi4;
MA2(M-1,M-1)=kesi5;

for i=1:M-1 AP2(i)=p(n,i+1); end;

AP2=AP2’;

RMA=MA2*AP2;

RMA(1)=RMA(1)+B2*(p(n,1)+p(n+1,1));
RMA(M-1)=RMA(M-1)+B1*(p(n,M+1)+p(n+1,M+1));

AP1=inv(MA1)*RMA;

for m=2:M p(n+1,m)=AP1(m-1); end;

AP2=AP2’;

\%

8 Correcting critical price

cp(n+1)=(alpha-p(n+1,2))/beta;
p(n+1,1)=1-cp(n+1);

end;
toc


