

The University of Tartu
Department of Mathematics and Computer Science
Institute of Mathematical Statistics

BABALWA MEHLOMAKULU

ESTIMATING THE TRUNCATION ERROR IN
THE CASE OF SOLVING ONE-DIMENSIONAL
BLACK-SCHOLES EQUATION

Advisor: Assoc. Prof. Raul Kangro

Tartu 2013

Contents

Introduction	3
1 Option Pricing	4
1.1 Put and Call options	4
1.2 Black-Scholes Market Model	5
1.3 Black-Scholes equation	5
1.4 Truncated problem	6
2 Maximum Principle for backward parabolic equations	8
2.1 Proofs	8
2.2 Outline of further way of getting estimates	11
3 Obtaining estimates at the boundaries	16
3.1 Estimates at the boundaries	16
4 Obtaining estimates inside the region	21
4.1 Estimates inside the region	21
5 Sample applications	26
Kokkuvõte (summary in Estonian)	29
Appendix	31

In the early 1970s, Fischer Black and Myron Scholes made a breakthrough by deriving a differential equation that must be satisfied by the price of any derivative security dependent on a non-dividend-paying stock [1]. They used the equation to obtain the values for European call and put options on the stock. Options are now traded on many different exchanges throughout the world and are very popular instruments for both speculating and risk management.

There are several approaches to option pricing but however we only consider Partial Differential Equations(PDE) approach, where options are expressed as solutions to certain partial differential equations. These equations are specified over an infinite(unbounded) region and usually cannot be solved exactly. Most numerical methods for solving partial differential equations require the region to be finite, so before applying numerical methods the problem is changed from infinite to finite region. The aim of our thesis is to study the error caused by this change, will do that by estimating the error at the boundaries and use these estimates to get pointwise error inside the domain, followed by numerical verification.

The structure of the thesis is as follows: Chapter one provides a brief introduction of option pricing and includes necessary results.

In chapter two we give a definition of maximum principle for backward parabolic equations and prove some lemmas based on this principle which will be useful throughout this thesis. We further outline ways of getting estimates with the aid of the results we got in our lemmas.

In chapter three we will obtain estimates at the truncation boundaries for both call and put option.

In chapter four we use the estimates of the previous chapter to find the estimates inside the region.

In chapter five we demonstrate the process of using our estimates in the case of pricing concrete put and call options and show the validity of the estimates by finding numerically the values of the solution of this truncated problem.

Chapter 1

Option Pricing

1.1 Put and Call options

Options on stocks were first traded on an organized exchange in 1973 [1]. Since then there has been a dramatic growth in options markets. Options are now traded on many different exchanges throughout the world and are a very popular instrument for speculating and hedging. The underlying assets include stocks, foreign currencies, commodities, future contracts, stock indices and debt instruments. There are two basic types of options, call and put options.

An **option** is a contract between a buyer and a seller that gives the buyer of the option the right, but not the obligation to buy or to sell a specified asset (underlying) on or before the option's expiration time at an agreed price, the strike price. In return for granting the option, the seller collects the payment from the buyer which amount is determined by the behaviour of the stock market up to the moment of executing the contract [3]. An option, just like a stock or bond is a security. It is also a binding contract with strictly defined terms and properties.

If the buyer chooses to exercise this right, the seller is obliged to sell or buy the asset at the agreed price. If the option may only be exercised at expiration time T then it is called **European option**.

European call option gives the buyer of the option the right but not the obligation to buy the underlying at the strike price on or before the expiration date. **European put option** gives the buyer of the option the right but not the obligation to sell the underlying at the strike price on or before the expiration date.

1.2 Black-Scholes Market Model

The Black-Scholes model for calculating premium of an option was introduced in 1973 [1]. The model, developed by three economists-Fischer Black, Myron Scholes and Robert Merton is perhaps the world's most well known options pricing model and the corresponding formulas for the theoretical prices of the European put and call options are probably among the most useful formulas of financial mathematics.

In order to use mathematics in option pricing we have to begin by specifying a model for stock price evolution. Based on the model we can come up with a rule for calculating the price. One of the commonly used models is the Black-Scholes Market model, which assumes that the stock price changes according to the stochastic differential equation

$$dS(t) = S(t)(\mu dt + \sigma dB(t)),$$

where $S(t)$ is the stock price at time t , μ is the average growth of the stock price, σ is the volatility of the stock price (measures the risks of the instrument) and B is the standard brownian motion.

In addition to the market model we make several additional simplifying assumptions [2]:

- the risk free interest rate is a known constant r and is the same for lending and borrowing;
- it is possible to trade continuously and with arbitrarily small fractions of a stock;
- there are no transaction costs;
- it is possible to make riskless profit by trading on the market.

It is clear, that some of the additional assumptions do not hold in practise and that the Black-Scholes model, at least with constant parameters μ and σ , is often not in a very good accordance with the real market behaviour, but it is still a good starting point for mathematical modeling of the market behaviour.

1.3 Black-Scholes equation

The Black-Scholes equation is a partial differential equation describing how the value of the price of an option changes when time and the current stock-price change, provided the Black-Scholes market model holds.

Black-Scholes equation is defined on an infinite region $(s, t) \in (0, \infty) \times [0, T)$

$$\frac{\partial u}{\partial t}(s, t) + \frac{\sigma^2 s^2}{2} \frac{\partial^2 u}{\partial s^2}(s, t) + (r - D)s \frac{\partial u}{\partial s}(s, t) - ru(s, t) = 0, \quad (1.1)$$

with the final condition

$$u(s, T) = p(s), s \geq 0. \quad (1.2)$$

It is known that the price of the European option which is equivalent to the right of owner to get the payment $p(S(T))$ at the exercise time T is at any time $t < T$ equal to $u(s(t), t)$, where u is the solution of (1.1) and (1.2). The function p giving the value of the option at the final time T is called the payoff function. We want to solve (1.1) in order to compute the option price at time $t = 0$ that is $(u(s_0, 0))$.

1.4 Truncated problem

For many numerical methods the partial differential equation has to be defined on a finite domain. So we replace (1.1) and (1.2) with the problem

$$\frac{\partial \tilde{v}}{\partial t}(s, t) + \frac{\sigma^2 s^2}{2} \frac{\partial^2 \tilde{v}}{\partial s^2}(s, t) + (r - D)s \frac{\partial \tilde{v}}{\partial s}(s, t) - r\tilde{v}(s, t) = 0,$$

$$s_{min} < s < s_{max}, 0 \leq t \leq T$$

together with the final condition

$$\tilde{v}(s, T) = p(s), s_{min} < s < s_{max}$$

and boundary conditions

$$\tilde{v}(s_{min}, t) = \phi_1(t), \tilde{v}(s_{max}, t) = \phi_2(t),$$

where $\phi_1(t), \phi_2(t)$ are some functions we have to specify ourselves.

We want to know what is the difference between $u(s_0, 0)$ and $\tilde{v}(s_0, 0)$.

Note that the difference

$$\xi(s, t) = u(s, t) - \tilde{v}(s, t)$$

satisfies in the region $s_{min} < s < s_{max}, 0 \leq t \leq T$ the equation

$$\frac{\partial \xi}{\partial t}(s, t) + \frac{\sigma^2 s^2}{2} \frac{\partial^2 \xi}{\partial s^2}(s, t) + (r - D)s \frac{\partial \xi}{\partial s}(s, t) - r\xi(s, t) = 0, \quad (1.3)$$

and

$$\xi(s, T) = 0, s_{min} < s < s_{max}. \quad (1.4)$$

The main tool for getting estimates for the difference ξ is the maximum principle for the parabolic equations, that we will discuss in the next chapter.

Chapter 2

Maximum Principle for backward parabolic equations

In this chapter we define maximum principle for backward parabolic equation and prove some lemmas which will be useful later when applying this principle to our partial differential. Maximum principle is the property of solutions to certain PDE's but in our work we will focus our attention on backward parabolic equations. This principle says that the maximum of the function in a domain can be found on the boundary of that domain. We will give a broad meaning to this principle in terms of proofs which will be useful later.

2.1 Proofs

Denote

$$\Omega = (x_{min}, x_{max}) \times [0, T),$$

where x_{min} , x_{max} are some fixed real numbers and T is a number greater than zero.

Lemma 1. *If u is continuous and two times continuously differentiable in the region Ω and satisfies the inequality*

$$\frac{\partial u}{\partial t}(x, t) + a \frac{\partial^2 u}{\partial x^2}(x, t) + b \frac{\partial u}{\partial x}(x, t) - cu(x, t) \geq 0, \quad (x, t) \in \Omega, \quad (2.1)$$

where $a, c \geq 0$, then it cannot attain positive maximum in Ω .

Proof Case1

Suppose maximum of equation (2.1) is attained at (x_0, t_0) , $x_{min} < x_0 < x_{max}$,

$0 < t_0 < T$.

Then necessary conditions for maximum are

$$\frac{\partial u}{\partial t}(x_0, t_0) = 0$$

$$\frac{\partial u}{\partial x}(x_0, t_0) = 0$$

$$\frac{\partial^2 u}{\partial x^2}(x_0, t_0) \leq 0$$

This implies that

$$cu(x_0, t_0) \leq a \frac{\partial^2 u}{\partial x^2}(x_0, t_0).$$

If a and c are positive it follows that $u(x_0, t_0) \leq 0$, thus we cannot have positive maximum.

Case 2

Suppose maximum of equation (2.1) is attained at $(x_0, 0)$, $x_{min} < x_0 < x_{max}$. Then necessary conditions for maximum are

$$\frac{\partial u}{\partial t}(x_0, 0) \leq 0,$$

$$\frac{\partial u}{\partial x}(x_0, 0) = 0,$$

$$\frac{\partial^2 u}{\partial x^2}(x_0, 0) \leq 0.$$

This implies that

$$cu(x_0, 0) \leq a \frac{\partial^2 u}{\partial x^2}(x_0, 0) + \frac{\partial u}{\partial t}(x_0, 0).$$

Since a and c are positive, we have $u(x_0, 0) \leq 0$, thus we cannot have positive maximum.

Lemma 2. *If u is continuous and two times continuously differentiable in the region Ω and satisfies the inequality*

$$\frac{\partial u}{\partial t}(x, t) + a \frac{\partial^2 u}{\partial x^2}(x, t) + b \frac{\partial u}{\partial x}(x, t) - cu(x, t) \leq 0, \quad (x, t) \in \Omega, \quad (2.2)$$

where $a, c \geq 0$, then it cannot attain negative minimum in Ω .

Proof Case1

If the minimum of equation (2.2) is attained at (x_0, t_0) , $x_{min} < x_0 < x_{max}$, $0 < t_0 < T$.

Then necessary conditions for minimum are

$$\frac{\partial u}{\partial t}(x_0, t_0) = 0,$$

$$\frac{\partial u}{\partial x}(x_0, t_0) = 0,$$

$$\frac{\partial^2 u}{\partial x^2}(x_0, t_0) \geq 0.$$

Thus from (1) we get

$$cu(x_0, t_0) \geq a \frac{\partial^2 u}{\partial x^2}(x_0, t_0).$$

If a and c are positive, then $u(x_0, t_0) \geq 0$, therefore we cannot have negative minimum.

Case 2

Suppose the minimum of equation (2.1) is attained at $(x_0, 0)$, $x_{min} < x_0 < x_{max}$.

Then necessary conditions for minimum are

$$\frac{\partial u}{\partial t}(x_0, 0) \geq 0,$$

$$\frac{\partial u}{\partial x}(x_0, 0) = 0,$$

$$\frac{\partial^2 u}{\partial x^2}(x_0, 0) \geq 0.$$

From (2.1) we get

$$cu(x_0, 0) \geq a \frac{\partial^2 u}{\partial x^2}(x_0, 0) + \frac{\partial u}{\partial t}(x_0, 0).$$

Since a and c are positive, then $u(x_0, 0) \geq 0$, no negative minimum.

2.2 Outline of further way of getting estimates

Here we turn our attention to estimating the difference $\xi(s, t) = u(s, t) - \tilde{v}(s, t)$, where u is the solution of problem (1.1), (1.2) and $\tilde{v}(s, t)$ is the solution of the truncated problem. Our procedure for estimating the difference is as follows:

1. Get estimates for $|\xi(s_{min}, t)| \leq C_1$ and $|\xi(s_{min}, t)| \leq C_2$.
2. Use maximum principle to get $\bar{w}(s, t)$ (upper estimate) and $\underline{w}(s, t)$ (lower estimate) such that $\underline{w}(s, t) \leq \xi(s, t) \leq \bar{w}(s, t)$.

For the second step, we prove the following results

Lemma 3. *If $\bar{w}(s, t)$ satisfies*

$$\frac{\partial \bar{w}}{\partial t}(s, t) + \frac{\sigma^2 s^2}{2} \frac{\partial^2 \bar{w}}{\partial s^2}(s, t) + (r - D)s \frac{\partial \bar{w}}{\partial s}(s, t) - r\bar{w}(s, t) \geq 0,$$

$$\bar{w}(s_{min}, t) \geq C_1, \bar{w}(s_{min}, t) \geq C_2$$

and

$$\bar{w}(s, T) \geq 0,$$

then

$$\xi(s, t) \leq \bar{w}(s, t) \forall s, s_{min} \leq s \leq s_{max}, 0 \leq t \leq T.$$

Proof Consider the difference $\eta(s, t) = \xi(s, t) - \bar{w}(s, t)$, it satisfies

$$\begin{aligned} & \frac{\partial \eta}{\partial t}(s, t) + \frac{\sigma^2 s^2}{2} \frac{\partial^2 \eta}{\partial s^2}(s, t) + (r - D)s \frac{\partial \eta}{\partial s}(s, t) - r\eta(s, t) \\ &= \frac{\partial \xi}{\partial t}(s, t) + \frac{\sigma^2 s^2}{2} \frac{\partial^2 \xi}{\partial s^2}(s, t) + (r - D)s \frac{\partial \xi}{\partial s}(s, t) - r\xi(s, t) \\ & - \left(\frac{\partial \bar{w}}{\partial t}(s, t) + \frac{\sigma^2 s^2}{2} \frac{\partial^2 \bar{w}}{\partial s^2}(s, t) + (r - D)s \frac{\partial \bar{w}}{\partial s}(s, t) - r\bar{w}(s, t) \right) \\ &= 0 - \left(\frac{\partial \bar{w}}{\partial t}(s, t) + \frac{\sigma^2 s^2}{2} \frac{\partial^2 \bar{w}}{\partial s^2}(s, t) + (r - D)s \frac{\partial \bar{w}}{\partial s}(s, t) - r\bar{w}(s, t) \right) \\ &\leq 0 \end{aligned}$$

and

$$\begin{aligned}\eta(s_{min}, t) &= \xi(s_{min}, t) - \bar{w}(s_{min}, t) \leq \xi(s_{min}, t) - C_1 \leq 0, \\ \eta(s_{max}, t) &= \xi(s_{max}, t) - \bar{w}(s_{max}, t) \leq \xi(s_{max}, t) - C_2 \leq 0, \\ \xi(s, T) - \bar{w}(s, T) &\leq 0.\end{aligned}$$

We see that $\eta(s, t) \leq 0$ at the boundaries $s = s_{min}, s = s_{max}$ and at $t = T$ and by (1) it cannot have positive maximum at

$$(s_{min}, s_{max}) \times [0, T)$$

hence

$$\xi(s, t) \leq \bar{w}(s, t), \quad s_{min} < s < s_{max}, \quad 0 \leq t \leq T.$$

Lemma 4. *If $\underline{w}(s, t)$ satisfies*

$$\frac{\partial \underline{w}}{\partial t}(s, t) + \frac{\sigma^2 s^2}{2} \frac{\partial^2 \underline{w}}{\partial s^2}(s, t) + (r - D)s \frac{\partial \underline{w}}{\partial s}(s, t) - r \underline{w}(s, t) \leq 0,$$

and

$$\underline{w}(s_{min}, t) \leq -C_1, \underline{w}(s_{max}, t) \leq -C_2, \quad \underline{w}(s, T) \leq 0,$$

then

$$\xi(s, t) \geq \underline{w}(s, t) \quad \forall s, s_{min} \leq s \leq s_{max}, \quad 0 \leq t \leq T.$$

Proof The difference of $\eta(s, t) = \xi(s, t) - \underline{w}(s, t)$ satisfies

$$\begin{aligned}& \frac{\partial \eta}{\partial t}(s, t) + \frac{\sigma^2 s^2}{2} \frac{\partial^2 \eta}{\partial s^2}(s, t) + (r - D)s \frac{\partial \eta}{\partial s}(s, t) - r \eta(s, t) \\ &= \frac{\partial \xi}{\partial t}(s, t) + \frac{\sigma^2 s^2}{2} \frac{\partial^2 \xi}{\partial s^2}(s, t) + (r - D)s \frac{\partial \xi}{\partial s}(s, t) - r \xi(s, t) \\ &\quad - \left(\frac{\partial \underline{w}}{\partial t}(s, t) + \frac{\sigma^2 s^2}{2} \frac{\partial^2 \underline{w}}{\partial s^2}(s, t) + (r - D)s \frac{\partial \underline{w}}{\partial s}(s, t) - r \underline{w}(s, t) \right) \\ &= 0 - \left(\frac{\partial \underline{w}}{\partial t}(s, t) + \frac{\sigma^2 s^2}{2} \frac{\partial^2 \underline{w}}{\partial s^2}(s, t) + (r - D)s \frac{\partial \underline{w}}{\partial s}(s, t) - r \underline{w}(s, t) \right) \\ &\geq 0\end{aligned}$$

and

$$\begin{aligned}\eta(s_{min}, t) &= \xi(s_{min}, t) - \underline{w}(s_{min}, t) \geq \xi(s_{min}, t) + C_1 \geq 0, \\ \eta(s_{max}, t) &= \xi(s_{max}, t) - \underline{w}(s_{max}, t) \geq \xi(s_{max}, t) + C_2 \geq 0,\end{aligned}$$

$$\xi(s, T) - \underline{w}(s, T) \geq 0.$$

Since $\eta(s, t) \geq 0$ at the boundaries $s = s_{min}$, $s = s_{max}$ and at $t = T$ and by (2) we cannot have negative minimum at

$$(s_{min}, s_{max}) \times [0, T)$$

hence

$$\xi(s, t) \geq \underline{w}(s, t), \quad s_{min} < s < s_{max}, \quad 0 \leq t \leq T.$$

Next we are going to prove a lemma about the behaviour of the solution of the original Black-Scholes equation that enables us to estimate later the truncation error at the boundaries $s = s_{min}$ and $s = s_{max}$. In the proof we use the knowledge that the solution of the Black-Scholes equation can be expressed in the integral form as

$$u = \frac{e^{-r(T-t)}}{\sqrt{2\pi(T-t)}} \int e^{\frac{-(\ln s + (r-D-\frac{\sigma^2}{2})(T_1-t) - \xi)^2}{2\sigma^2(T_1-t)}} p(e^\xi) d\xi.$$

Lemma 5. *Let u be a solution of the Black-Scholes equation*

$$\frac{\partial u}{\partial t}(s, t) + \frac{\sigma^2 s^2}{2} \frac{\partial^2 u}{\partial s^2}(s, t) + (r - D)s \frac{\partial u}{\partial s}(s, t) - ru(s, t) = 0, \quad s \geq 0$$

with the final condition

$$u(s, T) = p(s), \quad 0 \leq t \leq T.$$

If

$$C_1 + C_2 s \leq p(s) \leq C_3 + C_4 s,$$

then

$$C_1 e^{-r(T-t)} + C_2 e^{-D(T-t)} s \leq u(s, t) \leq C_3 e^{-r(T-t)} + C_4 e^{-D(T-t)} s.$$

Proof Part 1

Let

$$v_1 = C_1 e^{-r(T-t)} + C_2 e^{-D(T-t)} s.$$

First we show that v_1 satisfies our Black-Scholes equation, taking partial derivatives we have

$$\begin{aligned} \frac{\partial v_1}{\partial t} &= rC_1 e^{-r(T-t)} + DC_2 e^{-D(T-t)} s, \\ \frac{\partial v_1}{\partial s} &= C_2 e^{-D(T-t)}, \\ \frac{\partial^2 v_1}{\partial s^2} &= 0. \end{aligned}$$

Substituting the partial derivatives to the original equation, we get

$$rC_1e^{-r(T-t)} + DC_2e^{-D(T-t)}s + 0 + (r - D)(C_2e^{-D(T-t)}) - r(C_1e^{-r(T-t)} + C_2e^{-D(T-t)}s) = 0,$$

therefore we can conclude that v_1 satisfies the Black-Scholes equation. The difference of

$$\Gamma(s, t) = v_1(s, t) - u(s, t)$$

satisfies

$$\begin{aligned} & \frac{\partial \Gamma}{\partial t}(s, t) + \frac{\sigma^2 s^2}{2} \frac{\partial^2 \Gamma}{\partial s^2}(s, t) + (r - D)s \frac{\partial \Gamma}{\partial s}(s, t) - r\Gamma(s, t) \\ &= rC_1e^{-r(T-t)} + DC_2e^{-D(T-t)}s + 0 + (r - D)(C_2e^{-D(T-t)}) - \\ & r(C_1e^{-r(T-t)} + C_2e^{-D(T-t)}s) - \\ & \left(\frac{\partial u}{\partial t}(s, t) + \frac{\sigma^2 s^2}{2} \frac{\partial^2 u}{\partial s^2}(s, t) + (r - D)s \frac{\partial u}{\partial s}(s, t) - ru(s, t) \right), \\ &= 0 - 0, \\ &= 0. \end{aligned}$$

At $t=T$, we get

$$\Gamma(s, T) = v_1(s, T) - u(s, T) = v_1(s, T) - p(s) \leq 0.$$

Using the integral representation for Γ we now get

$$\Gamma(s, t) \leq 0, \quad s \geq 0, \quad 0 \leq t \leq T,$$

thus we have

$$u(s, t) \geq v_1(s, t), \quad s \geq 0, \quad 0 \leq t \leq T.$$

Part 2

Let

$$v_2 = C_3e^{-r(T-t)} + C_4e^{-D(T-t)}s.$$

Similarly, we can show that v_2 satisfies the black-scholes equation. The difference of

$$\Gamma(s, t) = v_2(s, t) - u(s, t)$$

satisfies

$$\begin{aligned}
& \frac{\partial \Gamma}{\partial t}(s, t) + \frac{\sigma^2 s^2}{2} \frac{\partial^2 \Gamma}{\partial s^2}(s, t) + (r - D)s \frac{\partial \Gamma}{\partial s}(s, t) - r\Gamma(s, t) \\
&= rC_3 e^{-r(T-t)} + DC_4 e^{-D(T-t)} s + 0 + (r - D)(C_4 e^{-D(T-t)}) - \\
& r(C_3 e^{-r(T-t)} + C_4 e^{-D(T-t)} s) \\
&- \left(\frac{\partial u}{\partial t}(s, t) + \frac{\sigma^2 s^2}{2} \frac{\partial^2 u}{\partial s^2}(s, t) + (r - D)s \frac{\partial u}{\partial s}(s, t) - ru(s, t) \right), \\
&= 0 - 0, \\
&= 0.
\end{aligned}$$

At $t=T$, we have

$$\Gamma(s, T) = v_1(s, T) - u(s, T) = v_1(s, T) - p(s) \geq 0.$$

Using the integral representation for Γ we now get

$$\Gamma(s, t) \geq 0, \quad s \leq 0, \quad 0 \leq t \leq T,$$

hence

$$u(s, t) \leq v_1(s, t), \quad s \leq 0, \quad 0 \leq t \leq T.$$

Chapter 3

Obtaining estimates at the boundaries

In this chapter we consider two types of options, call and put options. Our aim is to construct estimates for upper and lower boundary for these options which will be useful for our computation later. We will start by constructing estimates for call option and later consider put option.

3.1 Estimates at the boundaries

Let us consider a call option, which gives the owner the right to buy a share of stock at time T for the price E .

The payoff function is given by $p(s) = \max(s - E, 0)$. Let us first consider the boundary $s = s_{max}(s_{max} > E)$, for this we use estimates

$$s - E \leq p(s) \leq s.$$

We see by (5) we can further write,

$$se^{-D(T-t)} - Ee^{-r(T-t)} \leq u(s, t) \leq se^{-D(T-t)}.$$

Therefore we have

$$\begin{aligned}
& \max_{t \in [0, T]} |u(s_{max}, t) - p(s_{max})| \\
&= \max_{t \in [0, T]} |u(s_{max}, t) - (s_{max} - E)| \\
&= \max_{t \in [0, T]} \max (|s_{max}e^{-D(T-t)} - s_{max} + E|, |s_{max}e^{-D(T-t)} - \\
&\quad Ee^{-r(T-t)} - s_{max} + E|) \\
&= \max (\max_{t \in [0, T]} (|s_{max}e^{-D(T-t)} - s_{max} + E|), \max_{t \in [0, T]} (|s_{max}e^{-D(T-t)} - \\
&\quad Ee^{-r(T-t)} - s_{max} + E|)).
\end{aligned}$$

Consider the first term

$$M_1 = \max_{t \in [0, T]} (|s_{max}e^{-DT}e^{Dt} - s_{max} + E|)$$

Note that for any function $f(t)$, we have

$$\begin{aligned}
\max_{t \in [0, T]} |f(t)| &= \max(\max_{t \in [0, T]} f(t), -\min_{t \in [0, T]} f(t)) = \max(|\max_{t \in [0, T]} f(t)|, \\
&\quad |\min_{t \in [0, T]} f(t)|).
\end{aligned}$$

Since the function inside the absolute value signs is increasing when t increases, its largest and smallest values are attained at $t=T$ and $t=0$.

Therefore

$$\begin{aligned}
M_1 &= \max(|s_{max}e^{-DT}e^{D0} - s_{max} + E|, |s_{max}e^{-DT}e^{DT} - s_{max} + E|), \\
&= \max(|s_{max}e^{-DT} - s_{max} + E|, E).
\end{aligned}$$

For the second term we have,

$$\begin{aligned}
M_2 &= \max_{t \in [0, T]} (|s_{max}e^{-D(T-t)} - Ee^{-r(T-t)} - s_{max} + E|), \\
&= \max_{t \in [0, T]} |(s_{max}e^{-DT}e^{Dt} - Ee^{-rT}e^{rt} - s_{max} + E)|.
\end{aligned}$$

Let

$$f(t) = s_{max}e^{-DT}e^{Dt} - Ee^{-rT}e^{rt} - s_{max} + E.$$

Note that $f(T) = 0$. Thus the extremal values of this function can occur at $t = 0$ and at the critical point $t = t_c$ if $t \in [0, T]$. Solving for critical points t_c of the function $f(t)$, we get from

$$f'(t_c) = DS_{max}e^{-D(T-t_c)} - rEe^{-r(T-t_c)} = 0,$$

that there is one critical point

$$t_c = \frac{1}{D-r} \ln\left(\frac{rE}{Ds_{max}}\right) + T,$$

if $D \neq 0, r \neq D$ and no critical points otherwise. After some algebra, we get

$$f(t_c) = E\left(\frac{r}{D} - 1\right)\left(\frac{rE}{Ds_{max}}\right)^{\frac{r}{D-r}} - s_{max} + E.$$

Therefore

$$M_2 = \max\left(|s_{max}e^{-DT} - Ee^{-rT} - s_{max} + E|, \left|E\left(\frac{r}{D} - 1\right)\left(\frac{rE}{Ds_{max}}\right)^{\frac{r}{D-r}} - s_{max} + E\right|\right)$$

if $0 < t < t_c$

and

$$M_2 = |s_{max}e^{-DT} - Ee^{-rT} - s_{max} + E|$$

otherwise. Finally, we have

$$|u(s_{max}, t) - p(s_{max})| \leq C_2, t \in [0, T]$$

where

$$C_2 = \max(M_1, M_2).$$

Getting estimates for the lower boundary C_1 at $s = s_{min}$ ($s_{min} \leq E$). Now it is better to use estimates

$$0 \leq p(s) \leq s.$$

By (5) we can further write,

$$0 \leq u(s, t) \leq se^{-D(T-t)}.$$

Note that $p(s_{min}) = 0$.

Thus

$$\begin{aligned} \max_{t \in (0, T)} |u(s_{min}, t) - p(s_{min})| &= \max_{t \in [0, T]} |u(s_{min}, t)| \\ &= \max_{t \in [0, T]} \max(|s_{min})e^{-D(T-t)}| \\ &= s_{min} \end{aligned}$$

Therefore

$$|u(s_{min}) - p(s_{min})| \leq C_1,$$

where

$$C_1 = s_{min}.$$

Put option

Let us consider put option, which gives the owner the right to sell a share of stock at time T for the price E . The payoff function is given by $p(s) = \max(E - s, 0)$. Let us first consider the boundary $s = s_{min}$ ($s_{min} \leq E$), for this we use estimates

$$E - s \leq p(s) \leq E,$$

by (5) we can further write,

$$Ee^{-r(T-t)} - se^{-D(T-t)} \leq u(s, t) \leq Ee^{-r(T-t)}$$

$$\begin{aligned} & \max_{t \in [0, T]} |u(s_{min}, t) - p(s_{min})| \\ &= \max_{t \in [0, T]} |u(s_{min}, t) - (E - s_{min})| \\ &= \max_{t \in [0, T]} \max(|Ee^{-r(T-T)} - E + s_{min}|, |Ee^{-r(T-t)} - s_{min}e^{-D(T-t)} - \\ & \quad E + s_{min}|) \\ &= \max\left(\max_{t \in [0, T]} |Ee^{-r(T-T)} - E + s_{min}|, \max_{t \in [0, T]} |Ee^{-r(T-t)} - s_{min}e^{-D(T-t)} - \right. \\ & \quad \left. E + s_{min}|)\right) \end{aligned}$$

Let us first consider the first term.

$$M_3 = \max_{t \in [0, T]} (|Ee^{-r(T-t)} - E + s_{min}|).$$

The function inside the absolute value signs increasing when t increases, therefore its largest and smallest values are attained at $t=T$ and $t=0$. Thus

$$\begin{aligned} M_3 &= \max(|Ee^{-rT}e^{r0} - E + s_{min}|, |Ee^{-rT}e^{rT} - E + s_{min}|) \\ &= \max(|Ee^{-rT} - E + s_{min}|, s_{min}). \end{aligned}$$

For second term, we have

$$M_4 = \max_{t \in [0, T]} (|Ee^{-r(T-t)} - s_{min}e^{-D(T-t)} - E + s_{min}|).$$

Let

$$f(t) = E(e^{-rT}e^{rt}) - s_{min}e^{-DT}e^{Dt} - E + s_{min}.$$

The extremal values of this function can occur at $t = 0, t = T$ and at the critical points $t = t_c$ but since $f(T) = 0$, we have to consider only $t = 0$ and $t = t_c$. Finding the critical points of the function $f(t)$, we get

$$f'(t_c) = E r e^{-r(T-t_c)} - D s_{min} e^{-D(T-t_c)} = 0$$

$$t_c = \frac{1}{r-D} \ln \left(\frac{D s_{min}}{r E} \right) + T,$$

if $r \neq 0, D \neq r$ and no critical points otherwise. After some algebra we get

$$f(t_c) = s_{min} \left(\frac{D}{r} - 1 \right) \left(\frac{D s_{min}}{r E} \right)^{\left(\frac{D}{r-D} \right)} - E + s_{min}.$$

Therefore for $0 < t_c < T$ we have

$$M_4 = \max \left(|E e^{-rT} - s_{min} e^{-DT} - E + s_{min}|, \left| s_{min} \left(\frac{D}{r} - 1 \right) \left(\frac{D s_{min}}{r E} \right)^{\left(\frac{D}{r-D} \right)} - E + s_{min} \right| \right),$$

and for $t_c \notin [0, T]$, we have

$$M_4 = |E e^{-rT} - s_{min} e^{-DT} - E + s_{min}|.$$

Finally, we have obtained the estimate

$$\max_{t \in [0, T]} |u(s_{min}, t) - p(s_{min})| \leq C_1, t \in [0, T],$$

where

$$C_1 = \max(M_3, M_4).$$

Getting the estimates for the upper boundary $s_{max} (s_{max} > E)$. Now we use estimates

$$0 \leq p(s) \leq E,$$

we see by (5) we can further write,

$$0 \leq u(s, t) \leq E e^{-r(T-t)}$$

Note that $p(s_{max}) = 0$, thus

$$\begin{aligned} \max_{t \in [0, T]} |u(s_{max}, t) - p(s_{max})| &= \max_{t \in [0, T]} |u(s_{max}, t)| \\ &= \max_{t \in [0, T]} |E e^{-r(T-t)}| \\ &= E. \end{aligned}$$

Therefore

$$|u(s_{max}, t) - p(s_{max})| \leq C_2,$$

where

$$C_2 = E.$$

Chapter 4

Obtaining estimates inside the region

In this section, we will first start by verifying the exponential function which is known to satisfies Black-Scholes equation and later use the derivative of this exponential function to obtain estimates inside in the region.

4.1 Estimates inside the region

The solution of original Black-Scholes equation equation in an integral form is given by

$$u(s, t) = \frac{e^{-r(T-t)}}{\sqrt{2\pi(T-t)}} \int e^{\frac{-(\ln s + (r-D-\frac{\sigma^2}{2})(T_1-t) - \xi)^2}{2\sigma^2(T_1-t)}} p(e^\xi) d\xi.$$

From this expression we get the idea to consider the exponential function

$$u_1(s, t) = e^{\frac{-(\ln s + (r-D-\frac{\sigma^2}{2})(T_1-t) - \xi)^2}{2\sigma^2(T_1-t)} - r(T-t) - \frac{\ln(T_1-t)}{2}}. \quad (4.1)$$

Let us check if (4.1) satisfies Black-Scholes equation.

For simplicity, let

$$\varphi(s, t) = \frac{-(\ln s + (r-D-\frac{\sigma^2}{2})(T_1-t) - \xi)^2}{2\sigma^2(T_1-t)} - r(T_1-t) - \frac{\ln(T_1-t)}{2}.$$

Therefore

$$u_1 = e^{\varphi(s,t)}.$$

Taking partial derivatives of (4.1) yields

$$\begin{aligned}\frac{\partial u_1}{\partial s} &= -\frac{(\ln s + (r - D - \frac{\sigma^2}{2})(T_1 - t) - \xi)}{s\sigma^2(T_1 - t)}e^{\varphi(s,t)}, \\ \frac{\partial u_1}{\partial t} &= \left(\frac{(\ln s + (r - D - \frac{\sigma^2}{2})(T_1 - t) - \xi)}{\sigma^2(T_1 - t)}(r - D - \frac{\sigma^2}{2}) - \right. \\ &\quad \left. \frac{(\ln s + (r - D - \frac{\sigma^2}{2})(T_1 - t) - \xi)^2}{2\sigma^2(T_1 - t)^2} + r + \frac{1}{2(T_1 - t)}\right)e^{\varphi(s,t)}, \\ \frac{\partial^2 u_1}{\partial s^2} &= \left(\left(\frac{(\ln s + (r - D - \frac{\sigma^2}{2})(T_1 - t) - \xi)}{s\sigma^2(T_1 - t)}\right)^2 - \frac{1}{s^2\sigma^2(T_1 - t)} + \right. \\ &\quad \left. \frac{(\ln s + (r - D - \frac{\sigma^2}{2})(T_1 - t) - \xi)}{s^2\sigma^2(T_1 - t)}\right)e^{\varphi(s,t)}.\end{aligned}$$

Substitute the partial derivatives to the Black-Scholes equation

$$\frac{\partial u_1}{\partial t}(s,t) + \frac{\sigma^2 s^2}{2} \frac{\partial^2 u_1}{\partial s^2}(s,t) + (r - D)s \frac{\partial u_1}{\partial s}(s,t) - ru_1(s,t) = 0$$

without including the exponential function since it appears in all the terms, we divide both sides of the equation with it.

Thus we get

$$\begin{aligned}&\frac{(\ln s + (r - D - \frac{\sigma^2}{2})(T_1 - t) - \xi)}{\sigma^2(T_1 - t)}(r - D - \frac{\sigma^2}{2}) + r + \frac{1}{2(T_1 - t)} - \\ &\frac{(\ln s + (r - D - \frac{\sigma^2}{2})(T_1 - t) - \xi)^2}{2\sigma^2(T_1 - t)^2} + \frac{(\ln s + (r - D - \frac{\sigma^2}{2})(T_1 - t) - \xi)^2}{2\sigma^2(T_1 - t)^2} \\ &+ \frac{(\ln s + (r - D - \frac{\sigma^2}{2})(T_1 - t) - \xi)}{2(T_1 - t)} - \frac{1}{2(T_1 - t)} - \\ &\frac{r(\ln s + (r - D - \frac{\sigma^2}{2})(T_1 - t) - \xi)}{\sigma^2(T_1 - t)} + \frac{D(\ln s + (r - D - \frac{\sigma^2}{2})(T_1 - t) - \xi)}{\sigma^2(T_1 - t)} - r.\end{aligned}$$

Simplifying, we get

$$\begin{aligned}&(\ln s + (r - D - \frac{\sigma^2}{2})(T_1 - t) - \xi)\left(\frac{r}{\sigma^2} - \frac{D}{\sigma^2} - \frac{1}{2} + \frac{1}{2} - \frac{r}{\sigma^2} + \right. \\ &\quad \left. \frac{D}{\sigma^2}\right) \\ &= (\ln s + (r - D - \frac{\sigma^2}{2})(T_1 - t) - \xi)(0), \\ &= 0.\end{aligned}$$

Therefore (4.1) satisfies Black-Scholes equation.

Claim: For any fixed value of $s \neq e^{\varphi(s,t)}$ the minimum of the exponential function for $t \in [0, T]$ can only occur at $t = 0, t = T$.

Consider

$$\frac{\partial u_1}{\partial t} = \left(\frac{(\ln s + (r - D - \frac{\sigma^2}{2})(T_1 - t) - \xi)}{\sigma^2(T_1 - t)} (r - D - \frac{\sigma^2}{2}) - \frac{(\ln s + (r - D - \frac{\sigma^2}{2})(T_1 - t) - \xi)^2}{2\sigma^2(T_1 - t)^2} + r + \frac{1}{2(T_1 - t)} \right) e^{\varphi(s,t)}.$$

We show that there is only one critical point for $t < T_1$ and that $\frac{\partial u}{\partial t} > 0$ for $t < t_c$ and $\frac{\partial u}{\partial t} < 0$ for $t > t_c$. Therefore t_c is a maximum point.

In order to find critical points we can ignore the exponential term.

Denote

$$x_t = (T_1 - t)(r - D - \frac{\sigma^2}{2}).$$

$$y_s = \ln s - \xi.$$

Thus the factor before the exponential term in $\frac{\partial u_1}{\partial t}$ term is

$$\begin{aligned} & \frac{(y_s + x_t)x_t}{\sigma^2(T_1 - t)} - \frac{(y_s - x_t)^2}{2\sigma^2(T_1 - t)^2} + r + \frac{1}{2(T_1 - t)}, \\ &= \frac{x_t^2 - y_s^2 + 2\sigma^2\tau^2r + \sigma^2\tau}{2\sigma^2(T_1 - t)^2}. \end{aligned}$$

From this expression we see that $\frac{\partial u_1}{\partial t}$ is less than zero when t approaches T_1 and it becomes larger than zero when t approaches $-\infty$.

Let

$$\tau = T_1 - t$$

then we want to solve

$$\frac{\tau^2(r - D - \frac{\sigma^2}{2}) - y_s^2 + 2\sigma^2\tau^2r + \sigma^2\tau}{2\sigma^2\tau^2} = 0.$$

Equating the numerator to zero and solving for τ , we get

$$\begin{aligned} \tau^2(r - D - \frac{\sigma^2}{2}) - y_s^2 + 2\sigma^2\tau^2r + \sigma^2\tau &= 0 \\ \tau^2(2\sigma^2r + (r - D - \frac{\sigma^2}{2})) + \sigma^2\tau - y_s^2 &= 0 \end{aligned}$$

Using quadratic equation we get the value of τ as

$$\tau = \frac{-\sigma^2 \pm \sqrt{\sigma^4 + 4(2\sigma^2 r + (r - D - \frac{\sigma^2}{2})^2)y_s^2}}{2(2\sigma^2 r + (r - D - \frac{\sigma^2}{2})^2)}.$$

Since the expression under the square root is larger than σ^4 , one of the τ values is positive and one is negative. Thus there is only solution $t_c < T_1$ and it is maximum point for u_1 .

Now we can construct the functions \bar{w} and \underline{w} for estimating the value of the truncation error. Let s_{min}, s_{max} be fixed.

Pick $T_1 > T$ (say $T_1 = 2T$) and $(\xi_{max} \geq \ln s_{max})$.

Let

$$w_2(s, t) = e^{\frac{-(\ln s + (r - D - \frac{\sigma^2}{2})(T_1 - t) - \xi)^2}{2\sigma^2(T_1 - t)} - r(T - t) - \frac{\ln(T_1 - t)}{2}}.$$

Denote

$$c_2 = \min_{t \in [0, T]} w_2(s, t) = \min(w_2(s_{max}, 0), w_1(s_{max}, T))$$

Pick $\xi_{min}, \xi_{min} < \ln s_{max}$.

Let

$$w_1(s, t) = e^{\frac{-(\ln s + (r - D - \frac{\sigma^2}{2})(T_1 - t) - \xi_{min})^2}{2\sigma^2(T_1 - t)} - r(T_1 - t) - \frac{\ln(T_1 - t)}{2}}.$$

Denote

$$c_1 = \min_{t \in [0, T]} w_1(s_{min}, t) = \min(w_1(s_{min}, 0), w_1(s_{min}, T)).$$

For upper estimate we have

$$\bar{w} = \frac{C_1}{c_1} w_1(s, t) + \frac{C_2}{c_2} w_2(s, t),$$

where

$$s = s_{min}, \bar{w}(s, t) \geq C_1 \text{ and at } s = s_{max}, \bar{w}(s, t) \geq C_2$$

\bar{w} satisfies Black-Scholes equation because of the lineary property of Black-Scholes equation. According to (3) now the truncation error is less than $\bar{w}(s, t) \forall s, t$.

For lower estimate have

$$\underline{w} = -\bar{w}$$

hence

$$\underline{w} = -\left(\frac{C_1}{c_1}w_1(s, t) + \frac{C_2}{c_2}w_2(s, t)\right),$$

where

$$s = s_{min}, \bar{w}(s, t) \leq -C_1 \text{ and at } s = s_{max}, \bar{w}(s, t) \leq -C_2$$

\underline{w} satisfies Black-Scholes equation because of the linearity property of Black-Scholes equation. Hence by (4) the truncation error is bounded below by $\underline{w}(s, t)$.

Chapter 5

Sample applications

Consider the case of pricing put and call options in the case of parameters $r = 0.03$, $D = 0.02$, $T = 0.5$, $E = 100$, $S_0 = 95$, $\sigma = 0.5$.

Let us first consider the call option. Suppose we want to estimate the truncation error if we choose $s_{min} = S_0/\rho$, $s_{max} = \rho S_0$ when $\rho = 3$. If we choose $T_1 = 2T$, $\xi_{min} = s_{min}/1.5$, $\xi_{max} = s_{max}/1.5$ and use our procedures from the previous chapters we first get estimates $C_1 = 31.7$, $C_2 = 100$ for the error at the truncation boundary and then compute the estimate for the error for $s = S_0$, $t = 0$ by

$$estimate = \left(\frac{C_1}{c_1} w_1(s, t) + \frac{C_2}{c_2} w_2(s, t) \right),$$

which gives us 1.699, see appendix A. This is one possible estimate and for different T_1 , ξ_{min} , ξ_{max} we get different estimates. In order to find the best estimate obtainable by our procedures, we used minimization function in the subpackage of the SciPy of Python and we obtained that the best estimate is 0.620762 corresponding to the choices $T_1 = 0.53$, $\xi_{min} = s_{min}/3$, $\xi_{max} = 3s_{max}$. The code is in appendix B. Repeating this for values $\rho = 2, 3, 4, 5$ we get the estimates in the table.

ρ	2	3	4	5
estimates	13.886	0.621	0.031	0.002
actual errors	0.0026	0.003	< 0.0005	< 0.0005

In addition to our estimates we have also computed the actual truncation errors by using the code from the Computational Finance course, see appendix E. We see that the actual errors are much smaller than our estimated errors but still our estimates are useful because we can choose the values of the

location of the boundaries s_{min} and s_{max} by the estimates so that we can be sure that the truncation errors are small enough.

Consider also the case of the put option. By repeating the procedures outline for call option we get the estimates in the table. See appendix C, D and E.

ρ	2	3	4	5
estimates	14.092	0.634	0.032	0.002
actual errors	0.017	0.016	< 0.0005	< 0.0005

As we see, again our estimates for the truncation error get smaller quickly when the parameter ρ increases, so again the estimates allow one to choose the value of ρ .

Finally, as we saw, our error estimates were much larger than the actual errors. This is mainly caused by quite crude estimates at the boundaries. If one can estimate the errors at the boundaries better, our technique would enable to get better estimates inside the region.

Conclusion

In this thesis we first defined important terms in option pricing and introduced Black-Scholes model together with Black-Scholes equation in a finite region. We then defined the truncated problem in a finite region and an important principle in our work called maximum principle, we went further by illustrating and proving lemmas which are useful in the next sections.

Our aim was to study the error caused by the change from an infinite region to finite region, we did that by obtaining estimates for upper and lower boundaries for both call and put options at the truncated boundaries and we used these boundaries to obtain estimates inside the domain.

Finally, we presented concrete examples for call and put option and numerical verification.

Lõpmatu piirkonna lõplikuga asendamisest tingitud vea hindamine Black-Scholesi võrrandi lahendamisel

Babalwa Mehlomakulu
Kokkuvõte

Paljud finantsmatemaatika rakendustega seotud ülesanded on taandatavad osatuletistega diferentsiaalvõrrandite lahendamisele. Üheks selliseks ülesandeks on finantsoptioonide hindade arvutamise ülesanne, mis Black-Scholesi turumudeli kehtimise korral on lahendatav Black-Scholesi osatuletistega diferentsiaalvõrrandi lahendamise teel. Kahjuks enamasti ei ole sellised võrrandid analüütiliselt lahendatavad, seetõttu tuleb kasutada numbrilise lahendamise meetodeid. Siin on aga probleemiks see, et enamik populaarsetest osatuletistega diferentsiaalvõrrandite lahendamise meetoditest on kasutatavad lõplikes piirkondades püstitatud ülesannete korral, Black-Scholesi võrrand tuleb aga lahendada tõkestamata piirkonnas. Seetõttu asendatakse enamasti esialgne ülesanne sama võrrandi mingis suures lõplikus piirkonnas lahendamise ülesandega ja rakendatakse numbrilisi meetodeid juba lõplikus piirkonnas. Selline piirkonna muutmine toob aga endaga kaasa täiendava vea, mille suurust ja käitumist sõltuvalt asendamisest tehtavatest valikutest on küllalt vähe uuritud.

Käesoleva bakalaureusetöö eesmärgiks on tuletada hinnangud lõpmatu piirkonna lõplikuga asendamisest tingitud veale ühemõõtmelise Black-Scholesi võrrandi lahendamisel. Et vältida tehnilisi keerukusi, piirdatakse konstantsete kordajatega võrranditega. Töö on jaotatud viieks peatükiks. Esimeses neist tutvustatakse finantsoptioonidega seotud mõisteid. Teises tõestatakse nn maksimumprintsipi tagurpidi ajaga parabolsete võrrandite jaoks ning visandatakse skeem, kuidas selle printsipi abil saab uuritavat viga hinnata. Kolmandas peatükis tuletatakse piirkonna rajal kehtivad veahinnangud ostu- ja müügioptsioonide jaoks ning neljandas peatükis näidatakse, kuidas saab piirkonna raja jaoks tõestatud veahinnangute baasil konstrueerida funktsioonid, mis võimaldavad saada punktiviisilisi hinnanguid piirkonna sees. Viendas peatükis on kasutatud tõestatud tulemusi selleks, et leida konkreetset veahingud juhul, kui ostu- ja müügioptsiooni korral lõpmatu piirkonna asemel lahendada võrrand erineva suurusega lõplikes piirkondades.

Töös tõestatud veahinnangud on küll suhteliselt ebatäpsed, kuid võimaldavad täie kindlusega asendada lõpmatu piirkond sellise lõpliku piirkonnaga, et lahendajale olulises punktis on saadava lahendi väärtuse erinevus täpse lahendi väärtusest täie kindlusega väiksem etteantavast lubatavast veast. See

annab kindluse, et hiljem numbriliste meetoditega leitud ligikaudne lahend on samuti piisavalt lähedane esialgse ülesande lahendile. Töös saadud tulemusi oleks võimalik oluliselt täpsustada, kui õnnestuks paremini hinnata piirkonna rajal tekkivaid vigu, kuid see jääb juba järgnevate uuringute ülesandeks.

Bibliography

- [1] Black F, Scholes M, "The pricing of the options and corporate liabilities,"
Journal of Political Economy 81 (1973), 637-659.
- [2] Wilmontt P, Howison S, Dewyne. J, The Mathematics of Financial
Derivatives. A Student Introduction. Cambridge, 1995.
- [3] John C, Hull, Options, Futures, And The Other Derivative Securities.
Second Edition, 1993.

Appendix A

```
from scipy import *
#For call option
#data
r=0.03
D=0.02
sigma=0.5
T=0.5
S0=95
E=100
rho=3.0
rho1=1.5 #the location of xmin and xmax

smin=S0/rho
smax=rho*S0

M1=maximum(abs(smax*exp(-D*T)-smax+E),E)

    #critical point
t_c=1/float(D-r)*log(r*E/float(D*smax))+T

if 0<=t_c and t_c<=T:
    M2=maximum(abs(smax*exp(-D*T)-E*exp(-r*T)-smax+E),
        abs(E*((r/D-1)*(r*E/float(D*smax))**(r/float(D-r)))-smax+E))
else:
    M2=abs(smax*exp(-D*T)-E*exp(-r*T)-smax+E)

C2=maximum(M1,M2)
C1=smin
print C2
print C1
```



```

T1=2*T
def phi(T1,xi,S,t):
    return (-((log(S)+(r-D-sigma**2/2.0)*(T1-t)-xi)**2)/float(
        2*sigma**2*(T1-t))-r*(T1-t)-log(T1-t)/2.0)
def w(T1,xi,S,t):
    return exp(phi(T1,xi,S,t))

ximax=log(smax*rho1)
ximin=log(smin/rho1)
c2=minimum(w(T1,ximax,smax,0),w(T1,ximax,smax,T))
c1=minimum(w(T1,ximin,smin,0),w(T1,ximin,smin,T))
#estimate at s=S0, t=0
estimate=(C1/float(c1))*w(T1,ximin,S0,0)+(C2/float(c2))*w(T1,ximax,S0,0)
print estimate

```

Appendix B

```
from scipy import *
from scipy import optimize
#For call option
#data
r=0.03
D=0.02
sigma=0.5
T=0.5
S0=95
E=100
rho=3.0
rho1=1.1 #the location of xmin and xmax

smin=S0/rho
smax=rho*S0
def estimate(params):
    T1=1.01*T+params[0]**2
    rho1=params[1]
    rho2=params[2]
    M1=maximum(abs(smax*exp(-D*T)-smax+E),E)

    #critical point
    t_c=1/float(D-r)*log(r*E/float(D*smax))+T

    if 0<=t_c and t_c<=T:
        M2=maximum(abs(smax*exp(-D*T)-E*exp(-r*T)-smax+E),
                    abs(E*((r/D-1)*(r*E/float(D*smax))**(r/float(D-r)))-smax+E))

    else:
        M2=abs(smax*exp(-D*T)-E*exp(-r*T)-smax+E)
```

```

C2=maximum(M1,M2)
C1=smin
#print C2
#print C1

T1=2*T
def phi(T1,xi,S,t):
    return (-((log(S)+(r-D-sigma**2/2.0)*(T1-t)-xi)**2)/float(
        2*sigma**2*(T1-t))-r*(T1-t)-log(T1-t)/2.0)
def w(T1,xi,S,t):
    return exp(phi(T1,xi,S,t))

ximax=log(smax*rho2)
ximin=log(smin/rho1)
c2=minimum(w(T1,ximax,smax,0),w(T1,ximax,smax,T))
c1=minimum(w(T1,ximin,smin,0),w(T1,ximin,smin,T))
#estimate at s=S0, t=0
estimate=(C1/float(c1))*w(T1,ximin,S0,0)+(C2/float(c2))*w(T1,ximax,S0,0)
return(estimate)
params=optimize.fmin(estimate,[0.1,1.1,1.1])
T1=1.01*T+params[0]**2
print T1
\chapter*{Appendix C}
\begin{verbatim}
from scipy import *
#For put option
#data
r=0.03
D=0.02
sigma=0.5
T=0.5
S0=95
E=100
rho=3.0
rho1=1.5 #the location of xmin and xmax

smin=S0/rho
smax=rho*S0

M3=maximum(abs(E*exp(-r*T)-E-smin),smin)

```

```

        #critical point
t_c=1/float(r-D)*log(D*smin/r*E)+T
if 0<=t_c and t_c<=T:
    M4=maximum(abs(E*exp(-r*T)-smin*exp(-D*T)-E+smin),
        abs(smin*((D/r-1)*(D*smin/float(r*E))**(D/float(D-r)))-E+smin))

else:
    M4=abs(E*exp(-r*T)-smin*exp(-D*T)-E+smin)

C1=maximum(M3,M4)
C2=E
print C2
print C1

T1=2*T
def phi(T1,xi,S,t):
    return (-((log(S)+(r-D-sigma**2/2.0)*(T1-t)-xi)**2)/float(
        2*sigma**2*(T1-t))-r*(T1-t)-log(T1-t)/2.0)
def w(T1,xi,S,t):
    return exp(phi(T1,xi,S,t))

ximax=log(smax*rho1)
ximin=log(smin/rho1)
c2=min(w(T1,ximax,smax,T),w(T1,ximax,smax,0))
c1=min(w(T1,ximin,smin,T),w(T1,ximin,smin,0))
#estimate at s=S0, t=0
estimate=((C1/float(c1))*w(T1,ximin,S0,0)+(C2/float(c2))*w(T1,ximax,S0,0))
print estimate

```

Appendix D

```
from scipy import *
from scipy import optimize
#For put option
#data
r=0.03
D=0.02
sigma=0.5
T=0.5
S0=95
E=100
rho=3.0
rho1=1.1 #the location of xmin and xmax

smin=S0/rho
smax=rho*S0
def estimate(params):
    T1=1.01*T+params[0]**2
    rho1=params[1]
    rho2=params[2]
    M3=maximum(abs(E*exp(-r*T)-E-smin),smin)

    #critical point
    t_c=1/float(r-D)*log(D*smin/r*E)+T
    if 0<=t_c and t_c<=T:
        M4=maximum(abs(E*exp(-r*T)-smin*exp(-D*T)-E+smin),
                    abs(smin*((D/r-1)*(D*smin/float(r*E))**(D/float(D-r)))-E+smin))
    else:
        M4=abs(E*exp(-r*T)-smin*exp(-D*T)-E+smin)

    C1=maximum(M3,M4)
```

```

C2=E
#print C2
#print C1

T1=2*T
def phi(T1,xi,S,t):
    return (-((log(S)+(r-D-sigma**2/2.0)*(T1-t)-xi)**2)/float(
        2*sigma**2*(T1-t))-r*(T1-t)-log(T1-t)/2.0)
def w(T1,xi,S,t):
    return exp(phi(T1,xi,S,t))

ximax=log(smax*rho2)
ximin=log(smin/rho1)
c2=minimum(w(T1,ximax,smax,0),w(T1,ximax,smax,T))
c1=minimum(w(T1,ximin,smin,0),w(T1,ximin,smin,T))
#estimate at s=S0, t=0
estimate=(C1/float(c1))*w(T1,ximin,S0,0)+(C2/float(c2))*w(T1,ximax,S0,0)
return(estimate)
params=optimize.fmin(estimate,[0.1,1.1,1.1])

```

Appendix E

```
from scipy import *
from scipy import linalg
from scipy import stats
Phi=stats.norm.cdf
#The formulas for Black-Scholes call and put options
def Call(S,E,T,r,sigma,D=0):
    d1=(log(S/float_(E))+(r-D+sigma**2/2.0)*T)/(sigma*sqrt(T))
    d2=d1-sigma*sqrt(T)
    return(S*exp(-D*T)*Phi(d1)-E*exp(-r*T)*Phi(d2))
def Put(S,E,T,r,sigma,D=0):
    d1=(log(S/float(E))+(r-D+sigma**2/2.0)*T)/(sigma*sqrt(T))
    d2=d1-sigma*sqrt(T)
    return(-S*exp(-D*T)*Phi(-d1)+E*exp(-r*T)*Phi(-d2))
r=0.03
D=0.02
sigma=0.5
T=0.5
S0=95
E=100
rho=3.0

exact_call=Call(S0,E,T,r,sigma,D)
exact_put=Put(S0,E,T,r,sigma,D)

def implicit_transformed(m,n,p,rho,r,D,sigma,T,S0,phi1,phi2):
    """solver for the transformed Black-Scholes, constant volatility case.
    the parameter S0 and rho are used to define the boundaries xmin, xmax
    the result is the option price corresponding to to S(0)=S0."""
    xmin=log(S0/float(rho))
```

```

xmax=log(rho*S0)
dx=(xmax-xmin)/n
x=linspace(xmin,xmax,n+1)
dt=T/m #dt for our numerical method
U=zeros(shape=(n+1,m+1))
M=zeros(shape=(n+1,n+1))
M[0,0]=1
M[n,n]=1
F=zeros(n+1)
#the final condition
U[:,m]=p(exp(x))
i=arange(1,n)
for k in arange(m-1,-1,-1):#from m to 1. k corresponds to the time level
    t=k*dt #for which we have already computed the
    alpha=sigma**2/2 #values of U
    beta=r-D-alpha
    a=-dt/(dx**2)*(alpha-beta/2.0*dx)
    b=1+2*dt/dx**2*alpha+r*dt
    c=-dt/(dx**2)*(alpha+beta/2.0*dx)
    #form the system matrix
    M[i,i-1]=a
    M[i,i]=b
    M[i,i+1]=c
    #define the right hand side
    #boundary conditions
    F[0]=phi1(k*dt,exp(xmin))
    F[n]=phi2(k*dt,exp(xmax))
    F[i]=U[i,k+1]
    U[:,k]=linalg.solve(M,F)
return U[n/2,0]# return the value corresponding to S(0)=S0

def p_put(s):
    return(maximum(E-s,0))
def phi1_put(t,smin):
    return(p_put(smin))
def phi2_put(t,smax):
    return(0)
def price_put(n,m,rho):
    return(cn(n,m,rho,r,D,S0,T,sigma,phi1_put,phi2_put,p_put))
def price_put2(n,m,rho):
    return(implicit_transformed(m,n,p_put,rho,r,D,sigma,T,S0,

```



```

        phi1=phi1_put,phi2=phi2_put))
def p_call(s):
    return(maximum(s-E,0))
def phi1_call(t,smin):
    return(0)
def phi2_call(t,smax):
    return(p_call(smax))
def price_call(n,m,rho):
    return(cn(n,m,rho,r,D,S0,T,sigma,phi1_call,phi2_call,p_call))
def price_call2(n,m,rho):
    return(implicit_transformed(m,n,p_call,rho,r,D,sigma,T,S0,
        phi1=phi1_call,phi2=phi2_call))

def runge(epsilon,rho,m0,n0,value):
    #so that the diskretization error is less than epsilon/2
    m=m0
    n=n0
    error=epsilon+1 #to force to do the while cycle at least once
    result2=value(n,m,rho) #result2 denotes the most current value
    while(abs(error)>epsilon):#if the error oflast computation was
        m=m*4          #not small enough
        n=n*2
        result1=result2 #result1 denotes the previous value.
        result2=value(n,m,rho) #compute the new current value
        error=(result1-result2)/(3)
        print "runge",rho,m,n,result2,error
    return result2

epsilon=0.001
m0=10
n0=10
option_price=runge(epsilon,rho,m0,n0,price_call2)
print "call", "rho=", rho, "actual_error=",abs(exact_call-option_price)

option_price=runge(epsilon,rho,m0,n0,price_put2)
print "put", "rho=", rho, "actual_error=",abs(exact_put-option_price)

```

Non-exclusive licence to reproduce thesis and make thesis public

I _____Babalwa Mehlomakulu_____

(*author's name*)

(date of birth: _____1989-03-09_____),

1. herewith grant the University of Tartu a free permit (non-exclusive licence) to:
 - 1.1. reproduce, for the purpose of preservation and making available to the public, including for addition to the DSpace digital archives until expiry of the term of validity of the copyright, and
 - 1.2. make available to the public via the web environment of the University of Tartu, including via the DSpace digital archives until expiry of the term of validity of the copyright,

ESTIMATING THE TRUNCATION ERROR IN THE CASE OF SOLVING ONE
DIMENSIONAL BLACK-SCHOLES EQUATION. _____,

(title of thesis)

supervised by _____Raul Kangro_____

(supervisor's name)

2. I am aware of the fact that the author retains these rights.
3. I certify that granting the non-exclusive licence does not infringe the intellectual property rights or rights arising from the Personal Data Protection Act.

Tartu/Tallinn/Narva/Pärnu/Viljandi, **20.05.2013**