WHOLESALE BANKING

Fast solver for the three-factor Heston—Hull–White problem

F.H.C. Naber

Amsterdam 2006-2007

Thesis Committee:

Prof. dr. ir. C. Vuik (Delft University of Technology).
Dr. ir. C.W. Oosterlee (Delft University of Technology, CWI, Amsterdam).
Dr. ir. G.J.M. Pieters (ING, Amsterdam).
Dr. H.M. Schuttelaars (Delft University of Technology).

Acknowledgements

This thesis was submitted in the partial fulfillment for the requirements for the Master's degree in Applied Mathematics at Delft University of Technology. I owe gratitude to a lot of people, without whom this thesis would not have been what it is now. First of all I would like to thank my supervisor at ING, Gert-Jan Pieters, for all his support, proofreading all my drafts and his enthusiasm during the last nine months. Furthermore I would like to thank my supervisor at the TU Delft, Kees Oosterlee, for all of his valuable suggestions for this project. I am also indebted to Ben Sommeijer for his contribution to this project. Without the RKC method, this project would not have become a success.

I am also thankful for the friendly colleagues, Alain Verberkmoes, Anna Shepeleva and Vladimir Kulikov, who made my internship at ING a very pleasant one. Further I would like to thank my parents, family and friends for their help regarding the project and / or their never ending moral support.

Summary

The goal of this thesis is to implement a multi-dimensional finite difference solver, which can solve one-, two- and three-factor financial models in a fast and accurate way.

In the financial world most problems are initial value problems. To solve such problems with the finite difference method the computational domain has to be truncated and appropriate boundary conditions have then to be chosen. However, in the valuation of some financial instruments these boundary conditions are not known a priori. In literature the ”linearity” condition (which means that it is assumed that the second derivative is zero at the boundary of a truncated domain) is often used at boundaries where no conditions are known. This, however, may be an incorrect assumption.

Since the payoff basically implies an asymptotic behavior at the boundaries, the choice of the boundary condition has to be chosen as to avoid its influence on the solution on the whole domain. There are two ways to achieve this:

1. The use of implicit methods with the ”PDE” boundary condition. In this approach the whole PDE is discretized on the boundary grid node, using one-sided differences. In this way, all before mentioned models become initial boundary value problems, which can be solved with implicit time-integration methods [8].

2. The use of explicit time integration methods. In each time-integration step the boundary grid nodes do not contribute to the solution in all further time-integration steps. For this reason, one does not need to prescribe boundary conditions. Although it is the most natural strategy, it exhibits serious limitations concerning stability.

It is seen that applying the ”PDE” boundary condition does not resolve the problem of the a priori unknown boundary conditions in a satisfactory way. Therefore we will now focus on explicit methods for which boundary conditions can be omitted.

There are three approaches proposed, which are all based on an explicit time integration scheme. The first method is EulTree, which is based on the explicit Euler forward method. While this method can solve the one- and two-factor Black–Scholes and Hull–White model on an exponentially stretched grid, it can not be applied to all (multi-dimensional) models and all grids, so this method is not an appropriate one. The second method proposed is RKCTree, which is based on the Runge–Kutta–Chebyshev method. This method has the same disadvantages as EulTree and therefore the third method is introduced: RKCTreeImpr, which is also based on the RKC-method. This last method is well suited to solve one-, two- and three-factor models in a fast and accurate way. This model could, however, also be improved to perform even better. This will be the subject of further study.

Contents

Chapter 1

Introduction

1.1 The Black–Scholes model and drawbacks

It was in 1973 that Myron Scholes and Fischer Black came up with their Black–Scholes model to price options. Until then a few over-the-counter options and some exchange-warrants were traded, but from that moment on options exchanges were set up in Chicago, New York and Philadelphia; Later on in London, Paris and Tokyo. Nowadays there are exchanges in many countries. Options are traded all over the world.

The key assumption in the Black–Scholes model is that the return of the stock has a log-normal distribution. However, it can be seen in the market that the returns typically do not have a log-normal distribution. As Fischer Black remarks in [1], the assumptions are simple and unrealistic and when Black and Scholes tried to make money with the formula by simply buying options that were under priced and selling options that were overpriced, all they gained was a loss.

Another example where the Black–Scholes model is shown to price options insufficiently is after the crash in 1987. On October 19, 1987, better known as Black Monday, major indices all over the world dropped dramatically. Several reasons for the crash are given. Due to the strong economic growth, inflation was becoming a serious concern. Therefore short term interest rates were raised to temper the inflation. This had, however, also a serious effect on stocks. From that moment on major institutions protected themselves against further stock dips by buying future contracts, which can compensate for the losses in the stock market. The future market was taking in billions of dollars causing the market to crash from instability. It was also the time when computer trading began to flourish. People started to trust the new fast computational instruments blindly. Computers had taken over many tasks of the traders and were programmed to automatically sell or buy stocks when certain trends prevailed. The crash, started from instability, caused computers to sell stocks as fast as they could, rendering a snow ball effect in the downward spiral and causing the stock market to crash even further. The instability and the computer trading programs are believed to be two of the main reasons for the crash [2].

From that moment on participants did no longer believe in a log-normal return on the stock market, but rather, since they feared a repetition of the crash, put more value on instruments expected to do well in the event of large declines. Into the 1990's put options deep out-of-the-money were relatively more expensive than at-the-money calls and puts and out-of-the-money calls. This implies a higher volatility for these kinds of options, also known as the volatility smile or skew, which did not exist before the crash of 1987.

People tried to extend the Black–Scholes model such that it would capture the smile, while still having realistic underlying dynamics. The Black–Scholes was adjusted in various ways, e.g. by local volatility, displaced diffusion models, pure diffusion models, jump diffusion models. Others tried to model the volatility by assuming that it varies in a random way (stochastic volatility), such as Heston [3], Hull and White [19], Cox, Ingersoll and Ross [20]. The stochastic volatility models reflect the apparent randomness of the volatility and they can capture and explain the smile/skew, the mean reverting nature of volatility and other structures which are observed in the market. From these models (and other, potential more realistic ones) the Heston model (which is a version of the square root process described by Cox, Ingersoll and Ross) is the most popular. The Heston model will be discussed in Section 1.2.

One now could ask the question: If the volatility in the Black–Scholes model is so crucial, why would the other parameter in their model, the interest rate, be less important? As some financial instruments have exposure to (forward) volatility skewness, some others may have exposure to the stochastic nature of the interest rate. For that reason, one sometimes cannot rely on a deterministic instantaneous interest rate model and one has to model it stochastically.

In the literature several suggestions to model the stochastic interest rate are given. Among these models are, the Vasiček, Cox–Ingersoll–Ross, Dothan, Black–Derman–Toy, Ho–Lee and Hull–White model [7] of which the Hull–White model is the most popular one, because it can fit the observed market data completely. The Hull–White model will be discussed in Section 1.3. A combination of both stochastic volatility and stochastic interest rate is introduced in Section 1.4.

1.2 Extension to stochastic volatility: The Heston model

In the Black–Scholes model the value of the underlying financial product S_t (also called the spot value) is modeled by a stochastic process. The spot evolves according to the stochastic differential equation (SDE)

$$dS_t = (r - q)S_t dt + \sigma S_t dW, \tag{1.1}$$

with a Wiener process W, r the constant interest rate, q the constant dividend yield and $\sigma > 0$ describing the volatility.

In stochastic volatility models, σ in (1.1) is replaced by the square root of a variance which satisfies a second stochastic differential equation. In this thesis the Heston stochastic volatility model will be used. The Heston model is a mean reverting Ornstein–Uhlenbeck process (as observed in the market), where the dynamics for the underlying equity S_t and the Heston variance (square root of the volatility) v_t are given by:

$$\begin{aligned} dS_t &= (r - q)S_t dt + \sqrt{v_t} S_t dW_1, & (1.2)\\ dv_t &= -\lambda(v_t - \overline{v})dt + \eta\sqrt{v_t} dW_2, & (1.3)\\ \rho_{12} &= Cov(dW_1, dW_2)/dt, & (1.4) \end{aligned}$$

where $Cov(\cdot,\cdot)$ is the covariance, W_1 and W_2 are Wiener processes, λ is the speed of reversion of the instantaneous variance v_t to its long term mean $\overline{v}$ and η is the volatility of the volatility. Further ρ_{12} is the correlation between random stock price returns and changes in v_t.

The correlation ρ_{12} between the underlying equity and the volatility can also be interpreted as the distribution of the equity's log-return. Different ρ_{12} give different distributions. If $\rho_{12} > 0$ then the volatility increases as the equities price/return increases, causing the right tail to spread and the left tail to squeeze. If $\rho_{12} < 0$ it is the other way around. ρ_{12} effects the skewness of the density function, which is illustrated in Figure 1.1 [4].

Other than the skewness, ρ_{12}, also effects the shape of the implied volatility surface and it can imply a variety of volatility surfaces, thus overcoming the shortcoming of constant volatility in the Black–Scholes model.

The other parameters in the Heston model $\lambda, \overline{v}$ and η also influence the distribution of the asset price. λ can be seen as the degree of "volatility clustering". In Figures 1.2, 1.3 and 1.4 the sensitivity of the parameters on the distribution of the underlying value is shown. The λ, $\overline{v}$ and η parameters influence the variance and the kurtosis.

The main reason why the Heston model is so popular is due to the existence of a quasi closed form solution for European options [3]. In Figure 1.5 the option prices for a European call option in the Heston model with various parameters are plotted. "BS" indicates the standard Black-Scholes solution with $S_0 = 100$, $K = 100$, $v = \sqrt{0.183}$, $r = \log(1.0375)$, $t = 0$, $T = 2$ and all parameters in the Heston model set to zero. "Heston with positive correlation" refers to the same problem but now with $\lambda = 1.29$, $\overline{v} = 0.223^2, \eta = 0.431, \rho = 0.514$. "Heston with negative correlation" refers to the same problem with $\rho = -0.514$. From Figure 1.5 it can be concluded that the higher the correlation the more expensive the in-the-money options become, while the lower the correlation the more expensive the out-of-the-money option become. This is consistent with Figure 1.1 which states that the probability of

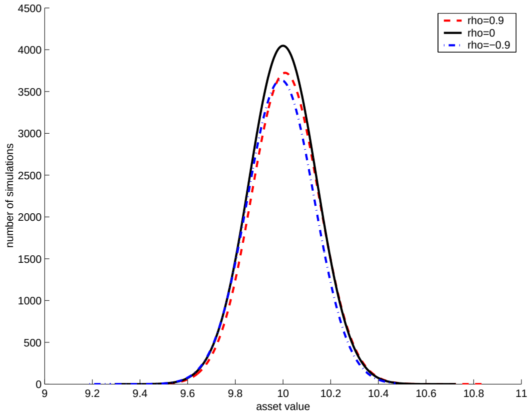

Figure 1.1: Distribution of asset price under Heston model. $S_0 = 10, r = 0, q = 0, \lambda = 0.01, \overline{v} = 2, \eta = 0.1, v_0 = 0.01$.

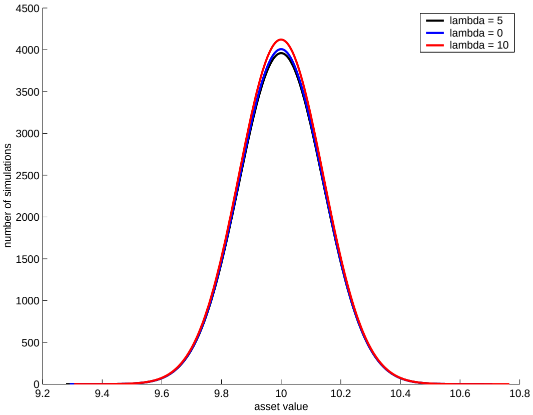

Figure 1.2: Distribution of asset price under Heston model. $S_0 = 10, r = 0, q = 0, \rho = 0, \overline{v} = 2, \eta = 0.1$.

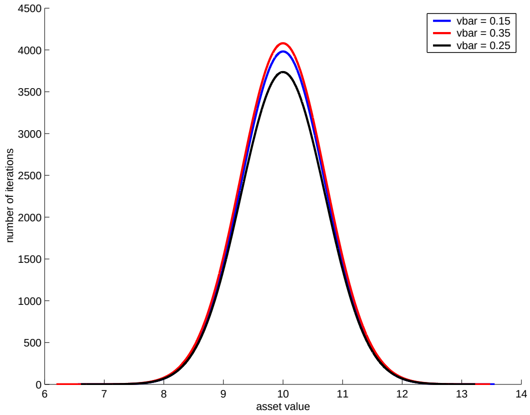

Figure 1.3: Distribution of asset price under Heston model. $S_0 = 10, r = 0, q = 0, \lambda = 0.01, \rho = 0, \eta = 0.1$.

a high asset price increases if the correlation becomes higher. As a consequence the option price becomes higher.

In abstract terms, a pricing equation (or pricing partial differential equation), is the ”law” of evolution of the value of the option. It allows for the valuation of options with a general payoff at maturity: The pricing equation is solved as a ”terminal value problem” in which the terminal condition is given by the payoff function. The pricing equation is derived from the stochastic differential equation (SDE).

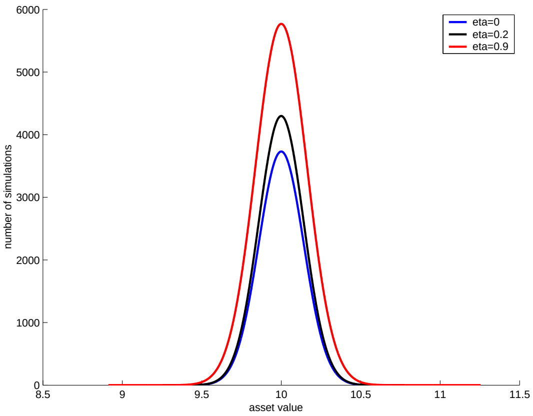

Figure 1.4: Distribution of asset price under Heston model. $S_0 = 10, r = 0, q = 0, \lambda = 0.01, \rho = 0, \overline{v} = 2$.

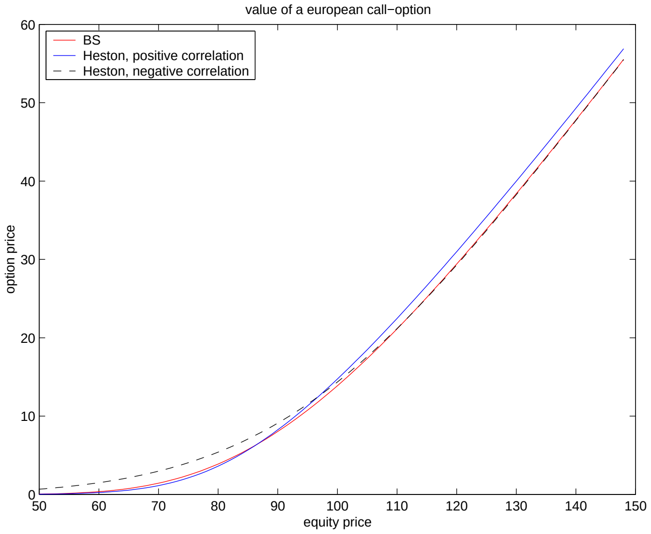

Figure 1.5: Call option prices in the Heston model.

There are two approaches to find the pricing equation for the two-factor model with stochastic volatility and an equity underlying. The first one is by setting up a portfolio Π containing the option being priced denoted by $V(S, v, t)$, $-\Delta$ of the underlying equity and $-\Delta_1$ of another equity whose value depends on volatility. After this portfolio is constructed one hedges the portfolio to make it risk free and since it is risk-free the return $\mathbf{d}\Pi$ of the portfolio should equal the return under the risk free rate, i.e. $\mathbf{d}\Pi$ = $r\Pi\mathbf{d}t$. Filling in all terms gives the pricing equation for the two-factor model. In [3] this approach is followed and the pricing equation for the Heston model is obtained.

The second approach is applying the Feynman–Kač theorem [6]. Feynman and Kač derived a relationship between stochastic differential equations and partial differential equations.

Theorem 1.1 (Feynman-Kač) *Suppose the underlying processes $y_1(t), y_2(t), .., y_n(t)$ follow the stochastic differential equation*

$$dy_i = \mu_i(y_1, y_2, .., y_n, t)dt + \sigma_i(y_1, y_2, .., y_n, t)dW_i. \tag{1.5}$$

Then the function

$$g(y_1, y_2, .., y_n, t) = E_{y_1,y_2,..,y_n,t}[f(y_1(T), .., y_n(T))] \tag{1.6}$$

is given by the solution of the partial differential equation

$$\frac{\partial g}{\partial t} + \sum_{i=1}^{n} \mu_i \frac{\partial g}{\partial y_i} + \frac{1}{2} \sum_{i,j=1}^{n} \rho_{ij}\sigma_i\sigma_j \frac{\partial^2 g}{\partial y_i \partial y_j} = 0, \tag{1.7}$$

subject to

$$g(y_1, y_2, .., y_n, T) = f(y_1, y_2, .., y_n), \tag{1.8}$$

where $\rho_{ij} = Cov(dW_i, dW_j)/dt$.

Let $y_1 = S_t$, $y_2 = v_t$ and let the price of a claim on S_t paying $f(S_T, v_T)$ at maturity be given by

$$V(S_t, v_t, t) = E_{S_t,v_t,t}\left(f(S_T, v_T)e^{-\int_t^T r(s)ds}\right). \tag{1.9}$$

Since r is constant, the term $e^{-\int_t^T r(s)ds}$ can be taken out of the expectation. Here after the following function can be defined

$$V(S_t, v_t, t) = e^{-r(T-t)}U(S_t, v_t, t), \tag{1.10}$$

where $U(S_t, v_t, t) = E_{S,v,t}(F(T))$. Now Theorem 1.1 can be applied to the function $U(S_t, v_t, t)$. Transforming the $U(S_t, v_t, t)$ back to $V(S_t, v_t, t)$ gives the pricing equation for the two-factor model

$$(\mathrm{H}) \begin{cases} \dfrac{\partial V}{\partial t} + (r-q)S\dfrac{\partial V}{\partial S} - \lambda(v-\overline{v})\dfrac{\partial V}{\partial v} - rV + \dfrac{1}{2}vS^2\dfrac{\partial^2 V}{\partial S^2} \\ \qquad +\rho_{12}Sv\eta\dfrac{\partial^2 V}{\partial S\partial v} + \dfrac{1}{2}\eta^2 v\dfrac{\partial^2 V}{\partial v^2} = 0, \quad S \in (0,\infty), \quad v \in (0,1), \quad t \in (0,T). \\ V(S,v,T) = f(S,v), \quad S \in (0,\infty), \quad v \in (0,1). \end{cases}$$

1.3 Extension to stochastic interest: The Hull–White model

The short term interest rate r_t will be modelled using the Hull–White model, for which the underlying dynamics are given by

$$dr_t = (\theta(t) - ar_t)dt + \sigma_r dW, \tag{1.11}$$

where θ is a function of time determining the average direction in which r_t moves ($\theta_{max} \approx 0.07$). The θ is chosen such that movements in r_t are consistent with todays zero coupon yield curve. The mean reversion rate a is usually taken constant ($a \approx 0.05$) and σ_r is the annual standard deviation of the short rate ($\sigma_r \approx 0.01$). The short term interest rate volatility σ_r is determined via calibration to caplets.

For the two-factor Hull–White model the dynamics for the underlying value S_t and the interest rate r_t are given by:

$$\begin{aligned} dS_t &= (r_t - q)S_t dt + \sigma S_t dW_1, \\ dr_t &= (\theta(t) - ar_t)dt + \sigma_r dW_2, \\ \rho_{12} &= Cov(dW_1, dW_2)/dt. \end{aligned} \tag{1.12}$$

Applying the Feynman–Kač Theorem to (1.12) gives the pricing equation. However, the derivation is not as straightforward as in the stochastic volatility case, since now the interest rate r_t is not constant and the term $e^{-\int_t^T r(s)ds}$ can not be pulled out of the expectation. This can be solved by defining an auxiliary process of the form $dz = -r(t)dt$. The function $V(S_t, v_t, r_t, t)$ is then equal to $e^{-z}E_{S_t,v_t,r_t,z,t}[f(T)e^{z(T)}]$.

Defining $V(S_t, v_t, r_t, t) = e^{-z}U(S_t, v_t, r_t, t)$ and applying Theorem 1.1 to $U(S_t, v_t, r_t, t)$ then gives the desired result. The whole derivation can be read in [6] or [5] and the pricing equation reads

$$\text{(H–W)}\begin{cases} \dfrac{\partial V}{\partial t} + (r-q)S\dfrac{\partial V}{\partial S} + (\theta - ar)\dfrac{\partial V}{\partial r} - rV + \dfrac{1}{2}\sigma^2 S^2 \dfrac{\partial^2 V}{\partial S^2} \\ \qquad +\rho_{12} S\sigma\sigma_r \dfrac{\partial^2 V}{\partial S \partial r} + \dfrac{1}{2}\sigma_r^2 \dfrac{\partial^2 V}{\partial r^2} = 0, \quad S \in (0,\infty), \quad r \in (-\infty,\infty), \quad t \in (0,T). \\ V(S,r,T) = f(S,r), \quad S \in (0,\infty), \quad r \in (-\infty,\infty). \end{cases}$$

1.4 Hybrid model

In this section we will derive the pricing equation for the three-factor model with stochastic interest, stochastic volatility and an equity underlying.
The dynamics for the underlying equity S_t, the interest rate r_t and the variance v_t are given by:

$$\begin{aligned} dS_t &= (r_t - q)S_t dt + \sqrt{v_t} S_t dW_1, && (1.13)\\ dr_t &= (\theta(t) - ar_t)dt + \sigma_r dW_2, && (1.14)\\ dv_t &= -\lambda(v_t - \overline{v})dt + \eta\sqrt{v_t} dW_3. && (1.15)\\ \rho_{ij} &= Cov(dW_i, dW_j)/dt. && (1.16) \end{aligned}$$

The price of a claim on S_t paying $F(S_T, r_T, v_T)$ at maturity is given by

$$V(S_t, v_t, r_t, t) = E_{S_t, r_t, v_t, t}\left(f(S_T, r_T, v_T) e^{-\int_t^T r(s)ds} \right). \tag{1.17}$$

The extension to the hybrid model is now relatively easy. Applying Theorem 1.1 and taking the auxiliary process for the interest into account we obtain the following pricing equation for the three-factor Heston–Hull–White model:

$$\text{(H—H–W)}\begin{cases} \dfrac{\partial V}{\partial t} + (r-q)S\dfrac{\partial V}{\partial S} + (\theta(t) - ar)\dfrac{\partial V}{\partial r} - \lambda(v_t - \underline{v})\dfrac{\partial V}{\partial v} - rV + \dfrac{1}{2}vS^2\dfrac{\partial^2 V}{\partial S^2} \\ \qquad +\rho_{12}S\sqrt{v}\sigma_r\dfrac{\partial^2 V}{\partial S\partial r} + \rho_{13}Sv\eta\dfrac{\partial^2 V}{\partial S\partial v} + \dfrac{1}{2}\sigma_r^2\dfrac{\partial^2 V}{\partial r^2} + \rho_{23}\sigma_r\eta\sqrt{v}\dfrac{\partial^2 V}{\partial r\partial v} \\ \qquad +\dfrac{1}{2}\eta^2 v\dfrac{\partial^2 V}{\partial v^2} = 0, \quad S \in (0,\infty), \quad r \in (-\infty,\infty), \quad v \in (0,1), \quad t \in (0,T). \\ V(S,v,r,T) = f(S,v,r), \quad S \in (0,\infty), \quad r \in (-\infty,\infty), \quad v \in (0,1). \end{cases}$$

1.5 Outline of this thesis

The goal of this thesis is to implement a multi-dimensional finite difference solver, which can solve initial value problems such as (H) (H–W) and (H—H–W) in a fast and accurate way. A problem which occurs when using a finite difference solver is that the computational domain must be truncated and as a consequence boundary values are needed. These boundary values are not always known and therefore we prefer the solver to be independent of the boundary values. In Chapter 2 we will introduce methods which solve the initial value problems with a finite difference approach without boundary conditions. After these methods are discussed they will be applied to the one-factor Black–Scholes and Hull–White models in Chapters 3 and 4 to analyse the performance on one-factor models. It turns out that one of the proposed methods is especially well suited for solving initial value problems and with this method the two-factor models Black–Scholes, (H) and (H–W) will be solved in Chapters 5, 6 and 7. An extension to three-factor models is made in Chapter 8.

Chapter 2

The finite difference method for option valuation

In this chapter we discuss numerical solution strategies for (H), (H–W) and (H—H–W). We focus on a finite difference implementation.

Generally, in numerical simulations the computational domain is truncated and appropriate boundary conditions have to be chosen. However, in the valuation of some financial instruments these boundary conditions are not known a priori. In literature the "linearity" condition (which means that it is assumed that the second derivative is zero at the boundary of a truncated domain) is often used at boundaries where no conditions are known. This, however, may be a wrong assumption. If we take for example a variance swap its payoff does not behave linearly at the boundaries.

Since the payoff basically implies an asymptotic behavior at the boundaries, the choice of the boundary condition has to be chosen as to avoid its influence on the solution on the whole domain. There are two ways to achieve this:

1. The use of implicit methods with the "PDE" boundary condition. In this approach the PDE is discretized on the boundary grid nodes, using one-sided differences. In this way, all before mentioned models become initial boundary value problems, which can be solved by implicit time-integration methods [8].
2. The use of explicit time integration methods. In each time-integration step the boundary grid nodes do not contribute to the solution in all further time-integration steps. For this reason, one does not need to prescribe boundary conditions. Although it is the most natural strategy, it exhibits serious limitations concerning stability.

In Section 2.2 we discuss strategy 1 and strategy 2 is explained in Section 2.3. First of all some basic spatial discretization will be introduced in the next section.

2.1 Spatial discretization

Consider the general one-dimensional differential operator L defined by

$$L(V) := \sigma(S,t)\frac{\partial^2 V}{\partial S^2} + \mu(S,t)\frac{\partial V}{\partial S} - r(t)V, \tag{2.1}$$

where $\sigma(S,t)$ is the volatility term, $\mu(S,t)$ the drift term, and $r(t)$ the discounting term. Further, let $\{S_i\}_{i=1}^{N+1}$ be the set of grid nodes representing some (non-uniform) partition of the yet unspecified truncated domain ($S_{min} := S_1, S_{N+1} =: S_{max}$). Define $V_i := V(S_i)$, $\mathbf{V} = (V_1, .., V_{N+1})^T$, and $\Delta S_i := S_{i+1} - S_i$. Then the first- and second-order spatial derivatives will be approximated by second order accurate three-point stencils. The first derivative in the interior grid nodes S_i is approximated as a linear combination of the neighboring values, i.e.

$$\frac{\partial V}{\partial S}(S_i) \quad \approx \quad \alpha_{-1}^i V_{i-1} + \alpha_0^i V_i + \alpha_1^i V_{i+1}, \qquad i = 2, .., N. \tag{2.2}$$

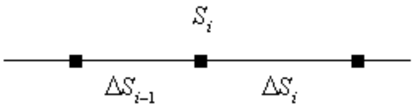

Figure 2.1: Discretization of the interior points.

The terms $V_{i-1} = V(S_i - \Delta S_{i-1})$ and $V_{i+1} = V(S_i + \Delta S_i)$ can be expanded in Taylor series

$$V(S_i - \Delta S_{i-1}) = V_i - \Delta S_{i-1}V_i' + (\Delta S_{i-1})^2\frac{1}{2}V_i'' - (\Delta S_{i-1})^3\frac{1}{6}V_i''' + O(\Delta S_{i-1}^4), \tag{2.3}$$

$$V(S_i + \Delta S_i) = V_i + \Delta S_i V_i' + (\Delta S_i)^2\frac{1}{2}V_i'' + (\Delta S_i)^3\frac{1}{6}V_i''' + O(\Delta S_i^4). \tag{2.4}$$

After substitution of (2.3) and (2.4) in (2.2) the $\alpha_{-1}^i, \alpha_0^i, \alpha_1^i$ have to be chosen such that the first derivative is approximated by second order accuracy. Working out all terms it can be easily shown that second order accuracy is achieved if $\alpha_{-1}^i, \alpha_0^i$ and α_1^i satisfy the linear system

$$\begin{bmatrix} 1 & 1 & 1 \\ -\Delta S_{i-1} & 0 & \Delta S_i \\ \frac{1}{2}\Delta S_{i-1}^2 & 0 & \frac{1}{2}\Delta S_i^2 \end{bmatrix} \begin{bmatrix} \alpha_{-1}^i \\ \alpha_0^i \\ \alpha_1^i \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}. \tag{2.5}$$

The solution of this system is given by

$$\begin{aligned} \alpha_{-1}^i &= \frac{\Delta S_i}{(\Delta S_i + \Delta S_{i-1})\Delta S_{i-1}}, \\ \alpha_0^i &= \frac{-(\Delta S_{i-1} - \Delta S_i)}{\Delta S_i \Delta S_{i-1}}, \\ \alpha_1^i &= \frac{\Delta S_{i-1}}{(\Delta S_i + \Delta S_{i-1})\Delta S_i}. \end{aligned}$$

For convection dominated problems, the first derivative cannot be approximated by the central scheme, because it may lead to unstable solutions. In this case the first-order upstream discretization is used. The "flow" direction depends on the coefficient in front of the first derivative, $\mu_i := \mu(S_i, t), i = 1, .., N+1$, and the formula for the upstream discretization is given by

$$\mu_i \frac{\partial V}{\partial S}(S_i) \approx \begin{cases} \mu_i \frac{V_{i+1} - V_i}{\Delta S_i} & \mu_i > 0, \\ \\ \mu_i \frac{V_i - V_{i-1}}{\Delta S_{i-1}} & \mu_i < 0. \end{cases} \tag{2.6}$$

This can again be written in the form (2.2) with

$$\begin{aligned} \alpha_{-1}^i &= -\frac{1}{2}(\frac{\mu_i}{\Delta S_i} - |\frac{\mu_i}{\Delta S_{i-1}}|), \\ \alpha_1^i &= \frac{1}{2}(\frac{\mu_i}{\Delta S_i} + |\frac{\mu_i}{\Delta S_{i-1}}|), \\ \alpha_0^i &= -\alpha_{-1}^i - \alpha_1^i. \end{aligned}$$

Similarly, the second derivative will be approximated by

$$\frac{\partial^2 V}{\partial S^2}(S_i) \approx \beta_{-1}^i V_{i-1} + \beta_0^i V_i + \beta_{-1}^i V_{i+1}, \qquad i = 2, .., N,$$

where the coefficients are now given by

$$\begin{aligned} \beta_{-1}^i &= \frac{2}{(\Delta S_i + \Delta S_{i-1})\Delta S_{i-1}}, \\ \beta_0^i &= \frac{-2}{(\Delta S_i + \Delta S_{i-1})\Delta S_{i-1}\Delta S_i}, \\ \beta_1^i &= \frac{2}{(\Delta S_i + \Delta S_{i-1})\Delta S_i}. \end{aligned}$$

Let A be the discretization matrix of the spatial operator L. Thus A contains the discretization of the interior and the boundary points and is of the form

$$A = \begin{bmatrix} \gamma_1 & \gamma_2 & \gamma_3 & \gamma_4 & \gamma_5 & 0 & \dots & 0 \\ a_2 & b_2 & c_2 & 0 & 0 & 0 & & \vdots \\ 0 & \ddots & \ddots & \ddots & \ddots & & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & & \ddots & \ddots & \ddots & \ddots & 0 \\ \vdots & & 0 & 0 & 0 & a_N & b_N & c_N \\ 0 & \dots & 0 & \epsilon_{N-3} & \epsilon_{N-2} & \epsilon_{N-1} & \epsilon_N & \epsilon_{N+1} \end{bmatrix}. \tag{2.7}$$

The elements a_i, b_i and $c_i,\ i = 2, .., N$, depend on the interior discretization, while $\gamma_1, .., \gamma_5$ and $\epsilon_{N-3}, .., \epsilon_{N+1}$ depend on the boundary conditions used.

In the following two sections we consider the semi-discretization problem

$$\begin{cases} \dfrac{d\mathbf{V}}{dt} + A\mathbf{V} = 0, \quad t \in (0, T), \\ \mathbf{V}(t = T) = \mathbf{V}_0. \end{cases} \tag{2.8}$$

This problem is solved backwards in time, since for our problems the value is given at $t = T$ from which we want to solve back to $t = 0$.

2.2 Initial boundary value problems and implicit methods

The first strategy to solve (2.8) is to use the implicit method with the "PDE" boundary condition. The "PDE" boundary condition will influence the first and last row of the semi-discretization matrix A (2.7) depending on the discretization used. The boundary discretizations are obtained in a similar way as the interior point discretization, but now by using only one-sided differences. Suppose the approximation of the first derivative at the boundary is given by

$$\frac{\partial V}{\partial S}(S_1) \quad \approx \quad \alpha_{-1}^1 V_1 + \alpha_0^1 V_2 + \alpha_1^2 V_3. \tag{2.9}$$

Then second order accuracy is reached if the following linear system is satisfied

$$\begin{bmatrix} 1 & 1 & 1 \\ 0 & \Delta S_1 & \Delta S_1 + \Delta S_2 \\ 0 & \frac{1}{2}\Delta S_1^2 & \frac{1}{2}(\Delta S_1 + \Delta S_2)^2 \end{bmatrix} \begin{bmatrix} \alpha_{-1}^1 \\ \alpha_0^1 \\ \alpha_1^1 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}. \tag{2.10}$$

The solution of this system is given by

$$\begin{aligned} \alpha_{-1}^1 &= \frac{-(2\Delta S_1 + \Delta S_2)}{(\Delta S_2 + \Delta S_1)\Delta S_1}, \\ \alpha_0^1 &= \frac{-(\Delta S_1 - \Delta S_2)}{\Delta S_2 \Delta S_1}, \\ \alpha_1^1 &= \frac{-\Delta S_1}{(\Delta S_2 + \Delta S_1)\Delta S_2}. \end{aligned}$$

First- and second-order accurate first and second derivatives are obtained in a similar way. After the boundaries are discretized the semi-discretization problem (2.8) is obtained and this can be solved with an implicit method, see Section 3.1.

It will, however, turn out that, as a consequence of the discretization at the boundaries using one-sided differences, the spectrum of the semi-discretization matrix A will contain positive eigenvalues, see [8].

This is something that should be avoided, since stability is then very hard to achieve. Another drawback of this approach is that when an extension is made to higher-dimensional models, the computational time will increase significantly because a simple tridiagonal solver cannot be used.

The second remark regarding the validity of the "PDE" boundary condition concerns the nature of the PDE. Suppose that $\mu(S,t) > 0$, where $\mu(S,t)$ is the drift term as defined in (2.1). Then the convection term at the right boundary is positive and since the problem is solved backwards in time, this means that this boundary is an inflow boundary. Hence, the discretization at the inflow boundary is opposite to the flow direction.

The last point is the choice of one-sided difference stencils. Rewriting the first derivative on the left boundary, using a uniform grid, gives

$$\frac{\partial V}{\partial S}(S_1) \approx \frac{V_2 - V_1}{\Delta S} = \frac{V_2 - V_0}{2\Delta S} + \frac{\Delta S}{2}\frac{V_2 - 2V_1 + V_0}{(\Delta S)^2},$$

where V_0 is a virtual point. This shows that some more diffusion is added to the problem at the left boundary and at the right boundary diffusion is subtracted. This negatively influences the accuracy of the solution.

Closer analysis of the first order approximation of the second derivative yields

$$\frac{\partial^2 V}{\partial S^2}(S_1) \approx \frac{V_3 - 2V_2 + V_1}{(\Delta S)^2} = \frac{V_2 - 2V_1 + V_0}{(\Delta S)^2} + (\Delta S)\frac{V_3 - 3V_2 + 3V_1 - V_0}{(\Delta S)^3},$$

where the last term is a numerical approximation to the third derivative. So here dispersion is added to the problem.

We conclude that applying the "PDE" boundary condition does not resolve the problem of the a priori unknown boundary conditions in a satisfactory way. Therefore we will now focus on explicit methods for which boundary conditions can be omitted.

2.3 Initial value problems and trees

In this section explicit time-integration methods are discussed. We will show that this approach does not require boundary conditions and is therefore suitable for solving initial value problems. Drawbacks of this approach are a stringent stability condition, leading to extremely small time steps. Therefore we propose a new family of explicit methods (Runge–Kutta–Chebyshev schemes [9]) which have weaker stability restrictions regarding the time step. The latter approach, albeit with some modifications, can be applied to all models (H), (H–W) and (H—H–W).

2.3.1 Explicit methods (Euler forward) and its limitations

For a three-point discretization stencil, explicit time-integration of (2.8) with constant coefficients using Euler's method can be described by the following recursive formula

$$(1 - r\Delta t)V_i^{n+1} = [p_{i,i-1}V_{i-1}^n + p_{i,i}V_i^n + p_{i,i+1}V_{i+1}^n], \quad i = 1,..,N+1, \quad n = 0,..,\lfloor T/\Delta t \rfloor, \tag{2.11}$$

or in the matrix vector notation

$$\begin{aligned} (1 - r\Delta t)\mathbf{V}^{n+1} &= (I + \Delta t A)\mathbf{V}^n =: B\mathbf{V}^n, \\ \mathbf{V}^0 &= \mathbf{V}_0. \end{aligned} \tag{2.12}$$

where

$$B = \begin{bmatrix} \ddots & \ddots & & & 0 \\ \ddots & \ddots & \ddots & & \\ & p_{i,i-1} & p_{i,i} & p_{i,i+1} & \\ & & \ddots & \ddots & \ddots \\ 0 & & & \ddots & \ddots \end{bmatrix}. \tag{2.13}$$

The matrix elements $p_{i,i-1}, p_{i,i}, p_{i,i+1}$, depend on the spatial discretization and will be discussed in Chapters 3 and 4.

If this problem is solved on a complete grid, boundary conditions are needed. However, this is what we want to avoid and instead of solving the problem on a complete grid *we will use a tree structured grid as shown in Figure 2.2.* Starting from the green nodes solving for the solution at the red node it is shown that boundary conditions are not necessary, since they do not influence the solution at the red point. Thus all information from the past is known and no information from the future is needed to calculate the new value. By stripping off these apparently redundant boundary grid nodes, the problem is solved backwards in time toward the spot price, or at least toward an interval containing the spot price. Hence this way of solving the problem requires $N \geq 2Q$, where $N+1$ denotes the number of grid nodes and $Q := \lfloor T/\Delta t \rfloor$ the number of time steps. Thus explicit methods may be used on a tree structured grid, which resolves the problem of the a priori unknown boundary conditions and saves computational time.

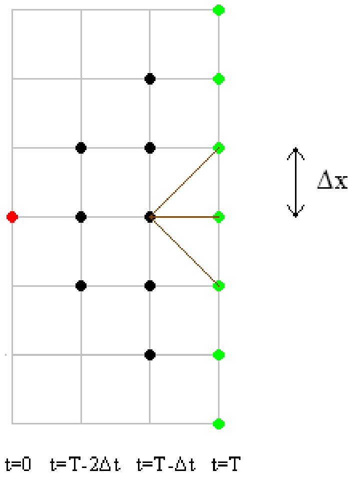

Figure 2.2: Tree grid approach.

Stability

In finance it is well-known that trinomial trees and the explicit Euler forward method are related, see for example [15]. The trinomial tree is based upon the idea that at each time level there are probabilities to jump to the nodes in the next time level. Probabilities have to be positive then.

Consider again difference equation (2.12). Suppose A can be diagonalized, i.e. $S^{-1}AS = diag(\lambda_1, .., \lambda_{N+1})$. Then the amplification matrix $B = B(\Delta t A)$ has the following properties:

$B(\Delta t A)$ is diagonalizable,

$S^{-1}AS = M$ with $M = diag(\tilde{Q}(\Delta t\lambda_1), .., \tilde{Q}(\Delta t\lambda_{N+1}))$,

$\tilde{Q}(\Delta t\lambda)$ is the amplification factor of the numerical method.

The numerical method is stable if the inequalities

$$|\tilde{Q}(\Delta t\lambda_i)| \leq 1 \qquad (2.14)$$

hold for all $1 \leq i \leq N+1$. For the Euler method, the amplification factor is given by $\tilde{Q}(\Delta t\lambda) = 1 + \Delta t\lambda$. We prove

Theorem 2.1 (Sufficient conditions for stability) *Scheme (2.11) is stable in the above-mentioned sense if the following holds:*

$p_{i,i-1}, p_{i,i}, p_{i,i+1} \geq 0,$

$p_{i,i-1} + p_{i,i} + p_{i,i+1} = 1.$

Proof. We use the Gershgorin circle theorem [16] to locate the eigenvalues of matrix B defined in (2.13). Let G be an arbitrary matrix with elements g_{ij}, and let C_i denote the circle in the complex plane with center g_{ii} and radius

$$R_i = \sum_{\substack{j=1 \\ j \neq i}}^{N+1} |g_{ij}|, \tag{2.15}$$

i.e. $C_i = \{z \in \mathbb{C} : |z - g_{ii}| \leq R_i\}$. The Gershgorin circle theorem then states that the eigenvalues of G are contained within the union of circles C_i, i.e. $\bigcup_{i=1}^{N+1} C_i$. Applying Gershgorin to matrix B yields

$$R_i = |p_{i,i+1}| + |p_{i,i-1}| = p_{i,i+1} + p_{i,i-1} \tag{2.16}$$

since $p_{i,i+1}$ and $p_{i,i-1}$ are both positive. The first and last row of B (the boundary conditions) give only smaller radii and are hence omitted in the proof. In particular we have

$$C_i = \{z \in \mathbb{C} : |z - p_{i,i}| \leq p_{i,i+1} + p_{i,i-1}\}. \tag{2.17}$$

Using $p_{i,i+1} + p_{i,i-1} + p_{i,i} = 1$, it is easily shown that

$$C_i \subseteq \{z \in \mathbb{C} : |z| \leq 1\}, \quad 1 \leq i \leq N+1, \tag{2.18}$$

which is just the condition for stability. See Figure 2.3 for a geometrical picture in the unit circle. ■

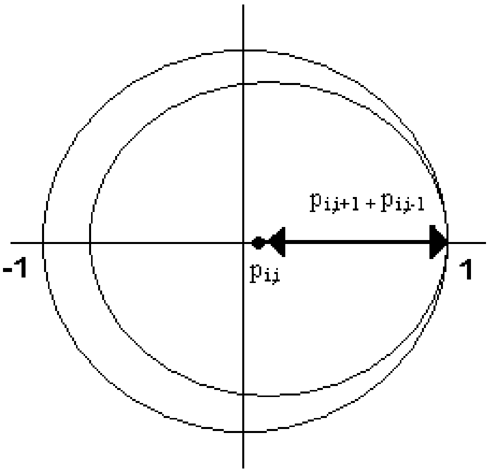

Figure 2.3: Gershgorin circles.

For stiff problems, the positivity condition for $p_{i,i}$ generally leads to very small time steps. This will be illustrated in Chapters 3 and 4. Therefore we will focus on another explicit method, which weakens the restrictions as derived above.

2.3.2 Weakening restrictions: The Runge–Kutta–Chebyshev method

In the previous section the Euler explicit method was discussed. It is known that this method has severe restrictions and since we would like to weaken these restrictions, an alternative explicit time integration method will be proposed.

One of the properties of this time discretization method has to be that the stability region includes a large part of the real negative axis. This allows us to deal with stiff problems. Therefore the Runge–Kutta–Chebyshev methods are proposed [9], since this time discretization method has the property that the stability region can cover as much of the real negative axis as needed. The method is however not very well suited if the spectrum of the discretization matrix contains imaginary eigenvalues.

Since the RKC method is an explicit method it is to be expected that a time step restriction is prescribed to keep the method stable. However with this method the number of grid nodes $N+1$ as well as the number of time steps Q can be chosen independently. To obtain stability for each integration step, from t_n to t_{n+1}, the number of stages s has to be chosen sufficiently large. A larger number for s does, however, not contribute to a higher accuracy.

Let the semi-discretization of some pricing equation be given by

$$\begin{cases} \mathbf{y}'(t) = \mathbf{F}(t, \mathbf{y}(t)), \quad t > 0, \\ \mathbf{y}(0) = \mathbf{y}_0. \end{cases} \tag{2.19}$$

Let $\mathbf{y}_n$ denote the numerical approximation to the exact solution of the semi-discrete system (2.19) at $t = t_n$, let τ be the stepsize from t_n to t_{n+1} and furthermore let $\mathbf{F}_k = \mathbf{F}(t_n + c_k\tau, \mathbf{W}_k)$. Then the second order explicit RKC method has the form

$$\begin{aligned} \mathbf{W}_0 &= \mathbf{y}_n, \\ \mathbf{W}_1 &= \mathbf{W}_0 + \tilde{\mu}_1 \tau \mathbf{F}_0, \\ \mathbf{W}_j &= (1 - \mu_j - \nu_j)\mathbf{W}_0 + \mu_j \mathbf{W}_{j-1} + \nu_j \mathbf{W}_{j-2} + \tilde{\mu}_j \tau \mathbf{F}_{j-1} + \tilde{\gamma}_j \tau \mathbf{F}_0, \quad j = 2, .., s, \\ \mathbf{y}_{n+1} &= \mathbf{W}_s. \end{aligned}$$

All coefficients are available in analytical form.

$$\begin{aligned} &\tilde{\mu}_1 = b_1\omega_1, \\ &\mu_j = \frac{2b_j\omega_0}{b_{j-1}}, \quad \nu_j = \frac{-b_j}{b_{j-2}}, \quad \tilde{\mu}_j = \frac{2b_j\omega_1}{b_{j-1}}, \quad \tilde{\gamma}_j = -a_{j-1}\tilde{\mu}_j, \quad j = 2, .., s, \\ &c_j = b_j\omega_1 T_j'(\omega_0), \end{aligned}$$

where

$$\begin{aligned} &T_j(x) = 2xT_{j-1}(x) - T_{j-2}(x), \quad T_0(x) = 1, \quad T_1(x) = x, \\ &a_j = 1 - b_j T_j(\omega_0), \\ &b_0 = b_2, \quad b_1 = \frac{1}{\omega_0}, \quad b_j = \frac{T_j''(\omega_0)}{(T_j'(\omega_0))^2}, \\ &\omega_0 = 1 + \frac{\epsilon}{s^2}, \quad \omega_1 = \frac{T_s'(\omega_0)}{T_s''(\omega_0)}. \end{aligned}$$

The damping parameter $\epsilon \geq 0$ influences the stability region. Taking ϵ small gives a long and small region of stability in the left half space, while taking ϵ large gives a wider but shorter strip as region of stability in the left half space.

Stability

For a fixed timestep Δt and for some fixed grid, the RKC method is stable if the number of stages s satisfies

$$s > \sqrt{\frac{\Delta t \rho(\mathbf{F}') + 1}{p_1}}, \tag{2.20}$$

where $\rho(\mathbf{F}')$ is the spectral norm of the Jacobian $\mathbf{F}'$ and p_1 depends on ϵ (in most cases ϵ will be taken zero, the corresponding value of p_1 is then $2/3$). It must be emphasized that the stability region regarding the real axis increases quadratically with s.

A good approximation for $\rho(\mathbf{F}')$ is needed. For this purpose we use again the Gershgorin circle theorem. For real spectra, a relatively sharp estimate for the spectral radius can be found by taking the diameter of the largest Gershgorin circle. In this largest circle, complex eigenvalues may exist. If the imaginary part of these complex eigenvalues is too large, then the RKC method may become unstable since these complex eigenvalues may not be contained in its stability region. Therefore we assume in the sequel of this thesis that the imaginary parts (if they exist) of the eigenvalues of the semi-discretization matrix A are small.

RKC method and the tree approach

The RKC method can be fit in the tree approach in two ways. The first way is to set the number of grid nodes N greater or equal to $2Qs$, where the number of stages s is determined using (2.20). After each stage the outer points will be stripped off, as they do not contribute the final solution in the spot and to reduce the computational complexity. This approach will be called RKCTree. However, the problem with this approach is that it is hard to match the number of grid nodes with the number of time steps and stages. The convergence is expected to be poor, because the grid has to be chosen in a very special way. Therefore a new approach is suggested in which stability can be guaranteed in all cases. Instead of stripping off the outer points after each stage, the outer points are stripped off after each time step and the boundary points are obtained by extrapolating the internal points. Therefore we can take $N \geq 2Q$, as for the Euler method. This approach will be referred to as RKCTreeImpr. In Figure 2.4 a sketch of this method is depicted. The green nodes are the initial values of the problem. The black nodes are the internal nodes and the blue nodes are the nodes that are extrapolated from the internal nodes. There are a number of ways to extrapolate these points such as linear, quadratic and cubic extrapolation. They will be explained below. Suppose that $S_1, .., S_{n+1}$ distinct grid nodes are given with the corresponding function values $V(S_1), .., V(S_{n+1})$ then there exists a unique polynomial $P(S)$ of at most n with

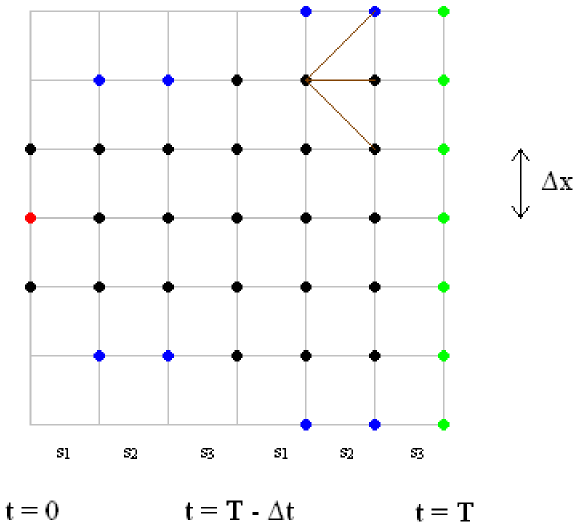

Figure 2.4: Improved tree grid approach.

$$V(S_k) = P(S_k), \quad k = 1, .., n+1. \tag{2.21}$$

This polynomial is given by

$$P(S) = \sum_{i=1}^{n+1} V(S_k) L_{n,k}(S), \tag{2.22}$$

where

$$L_{n,k}(S) = \prod_{\substack{i=1 \\ i \neq k}}^{n+1} \frac{S - S_i}{S_k - S_i}. \tag{2.23}$$

The general approach is to construct the Lagrange polynomial using an appropriate number n of internal points (linear $n = 2$, quadratic $n = 3$, cubic $n = 4$). Then the extrapolated grid node is put into the Lagrange polynomial to obtain the extrapolated value [12].

As a consequence of the extrapolation, the problem can not be solved back to a single point, because the boundary points must be extrapolated making use of only the interior points. Therefore the number of grid intervals N will be taken equal to

$$\begin{aligned} linear : N &\geq 2Q+2, \\ quadratic : N &\geq 2Q+4, \\ cubic : N &\geq 2Q+6. \end{aligned} \tag{2.24}$$

Apart from the fact that enough grid nodes should be taken such that the extrapolation makes sense it also has to be noted that RKCTreeImpr cannot be solved back to one point, since it is observed that processing backwards toward one single point will cause the solution to diverge too much from the full grid solution. This is caused by the fact that the extrapolation is linear, but the solution is not. As a consequence we have to solve to the smallest interval for which the solutions at the end nodes has minimal influence on the solution at the internal nodes [18].

2.4 Extension to higher-dimensional models

In the previous sections the discretization and the time-integration methods were discussed to solve a one-factor model. However, the goal of this thesis is to solve higher-dimensional models.

To solve higher-dimensional models the same approach can be applied. There are, however, some additional features which will be discussed below. Take for example the following general two-dimensional spatial operator L defined by

$$\begin{aligned} L(V) := \sigma_1(S_1,S_2,t)\frac{\partial^2 V}{\partial S_1^2} + \zeta_1(S_1,S_2,t)\frac{\partial^2 V}{\partial S_1 S_2} + \sigma_2(S_1,S_2,t)\frac{\partial^2 V}{\partial S_2^2} + \\ \mu_1(S_1,S_2,t)\frac{\partial V}{\partial S_1} + \mu_2(S_1,S_2,t)\frac{\partial V}{\partial S_2} - r(t)V = 0. \end{aligned} \tag{2.25}$$

As can be seen in (2.25), the higher-dimensional problem has the same terms in every direction as in the one-dimensional case. The main difference is, however, that crossterm(s) occur. So, while the drift and volatility terms in every direction can be discretized using the adjusted central scheme or upwind scheme, a discretization for the crossterm(s)has to be given to complete the discretization.

The discretization of the crossterm(s) will be done by taking the derivative in one direction and take the derivative of the result in the other direction using the adjusted central scheme. This leads to the second order accurate nine-point stencil

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}, \tag{2.26}$$

where

$$\begin{aligned}
a_{11} &= \frac{\Delta S_{2,k+1}}{\Delta S_{2,k+2}(\Delta S_{2,k+1}+\Delta S_{2,k+2})}\frac{-\Delta S_{1,j+2}}{\Delta S_{1,j+1}(\Delta S_{1,j+1}+\Delta S_{1,j+2})}\\
a_{12} &= \frac{\Delta S_{2,k+1}}{\Delta S_{2,k+2}(\Delta S_{2,k+1}+\Delta S_{2,k+2})}\frac{-\Delta S_{1,j+1}-\Delta S_{1,j+2}}{\Delta S_{1,j+1}\Delta S_{1,j+2}}\\
a_{13} &= \frac{\Delta S_{2,k+1}}{\Delta S_{2,k+2}(\Delta S_{2,k+1}+\Delta S_{2,k+2})}\frac{\Delta S_{1,j+1}}{\Delta S_{1,j+2}(\Delta S_{1,j+1}+\Delta S_{1,j+2})}\\
a_{21} &= \frac{\Delta S_{2,k+1}-\Delta S_{2,k+2}}{\Delta S_{2,k+1}\Delta S_{2,k+2}}\frac{-\Delta S_{1,k+2}}{\Delta S_{1,k+1}(\Delta S_{1,k+1}+\Delta S_{1,k+2})}\\
a_{22} &= \frac{\Delta S_{2,k+1}-\Delta S_{2,k+2}}{\Delta S_{2,k+1}\Delta S_{2,k+2}}\frac{-\Delta S_{1,k+1}-\Delta S_{1,k+2}}{\Delta S_{1,k+1}\Delta S_{1,k+2}}\\
a_{23} &= \frac{\Delta S_{2,k+1}-\Delta S_{2,k+2}}{\Delta S_{2,k+1}\Delta S_{2,k+2}}\frac{\Delta S_{1,k+1}}{\Delta S_{1,k+2}(\Delta S_{1,k+1}+\Delta S_{1,k+2})}\\
a_{31} &= \frac{\Delta S_{2,j+2}}{\Delta S_{2,j+1}(\Delta S_{1,j+1}+\Delta S_{1,j+2})}\frac{-\Delta S_{1,k+2}}{\Delta S_{1,k+1}(\Delta S_{1,k+1}+\Delta S_{1,k+2})}\\
a_{32} &= \frac{\Delta S_{2,j+2}}{\Delta S_{2,j+1}(\Delta S_{1,j+1}+\Delta S_{1,j+2})}\frac{-\Delta S_{1,k+1}-\Delta S_{1,k+2}}{\Delta S_{1,k+1}\Delta S_{1,k+2}}\\
a_{33} &= \frac{\Delta S_{2,j+2}}{\Delta S_{2,j+1}(\Delta S_{1,j+1}+\Delta S_{1,j+2})}\frac{\Delta S_{1,k+1}}{\Delta S_{1,k+2}(\Delta S_{1,k+1}+\Delta S_{1,k+2})}.
\end{aligned}$$

When a uniform grid is taken in both directions the elements of the stencil reduce to

$$\begin{aligned}
a_{11} &= \frac{1}{4\Delta S_1\Delta S_2}\\
a_{13} &= \frac{-1}{4\Delta S_1\Delta S_2}\\
a_{31} &= \frac{-1}{4\Delta S_1\Delta S_2}\\
a_{33} &= \frac{1}{4\Delta S_1\Delta S_2}.
\end{aligned}$$

Discretization of a higher-dimensional model will influence the discretization matrix (2.27). Extra off diagonals are introduced in the matrix, which will lead to the following matrix structure in the two-dimensional case

$$A = \begin{bmatrix}
\ddots & \ddots & & \ddots & \ddots & \ddots & \\
\ddots & \ddots & \ddots & & \ddots & \ddots & \ddots\\
 & \ddots & \ddots & \ddots & & \ddots & \ddots\\
\ddots & & \ddots & \ddots & \ddots & & \ddots\\
\ddots & \ddots & & \ddots & \ddots & \ddots & \\
\ddots & \ddots & \ddots & & \ddots & \ddots & \ddots\\
 & \ddots & \ddots & \ddots & & \ddots & \ddots
\end{bmatrix}. \tag{2.27}$$

The resulting semi-discretization can be solved by means of explicit schemes as for the one-factor models, by taking setting the number of grid nodes in each direction equal or greater than $2Q$, where Q is the number of time steps.

2.5 Computational costs

It is hard to compare the different methods by the amount of time it takes to calculate the solution, since these computational times depend on the implementation and time-measurement tools. Therefore, instead of looking at the computational times, we will look at the number of iterations needed to compute the solution. The different approaches will be compared by the number of iterations to compute the solution.

Suppose that we have a d-dimensional problem with $N_i+1, \quad i=1,..,d$ grid nodes in each direction. If a 3^d-point stencil is used then the number of iterations is given by Table 2.1. In this table $v_i := \prod_{j=1}^{d}(N_j+1-2(i-1))-\prod_{j=1}^{d}(N_j-1-2(i-1))$ and $w := \prod_{j=1}^{d}(N_j+1)-\prod_{j=1}^{d}(N_j-1)$.

Number of iterations		
	Matrix vector computations × and ÷	+ and −
Full Grid	$\sum_{i=1}^{Qs} 3^d \prod_{j=1}^{d}(N_j-1)$	$\sum_{i=1}^{Qs} 3^d \prod_{j=1}^{d}(N_j-2)$
EulTree	$\sum_{i=1}^{Q} 3^d \prod_{j=1}^{d}(N_j+1-2i)$	$\sum_{i=1}^{Q} 3^d \prod_{j=1}^{d}(N_j-2i)$
RKCTree	$\sum_{i=1}^{Qs} 3^d \prod_{j=1}^{d}(N_j+1-2i)$	$\sum_{i=1}^{Qs} 3^d \prod_{j=1}^{d}(N_j-2i)$
RKCImpr (linear)	$s\sum_{i=1}^{Q} 3^d \prod_{j=1}^{d}(N_j+1-2i)$	$s\sum_{i=1}^{Q} 3^d \prod_{j=1}^{d}(N_j-2i)$
RKCImpr (quadratic)	$s\sum_{i=1}^{Q} 3^d \prod_{j=1}^{d}(N_j+1-2i)$	$s\sum_{i=1}^{Q} 3^d \prod_{j=1}^{d}(N_j-2i)$
RKCImpr (cubic)	$s\sum_{i=1}^{Q} 3^d \prod_{j=1}^{d}(N_j+1-2i)$	$s\sum_{i=1}^{Q} 3^d \prod_{j=1}^{d}(N_j-2i)$
	Extrapolation × and ÷	+ and −
Full Grid	$(2\times(2+1))w$	$(2\times 4+1)w$
EulTree		
RKCTree		
RKCImpr (linear)	$(2\times(2+1))v_i$	$2\times(2\times 4+1)v_i$
RKCImpr (quadratic)	$(3\times(4+1))v_i$	$3\times(2\times 6+2)v_i$
RKCImpr (cubic)	$(4\times(6+1))v_i$	$4\times(2\times 8+3)v_i$

Table 2.1: Number of iterations for the various methods.

Part I

One-factor models

Chapter 3

Black–Scholes model

In the previous chapter two types of time integration methods were proposed. The implicit method with the "PDE" boundary condition and the explicit method solving on a tree structured grid. Both approaches will be applied to the Black–Scholes equation, which is given by

$$\text{(BS)} \begin{cases} \dfrac{\partial V}{\partial t} + (r-q)S\dfrac{\partial V}{\partial S} + \dfrac{1}{2}\sigma^2 S^2 \dfrac{\partial^2 V}{\partial S^2} - rV = 0, \quad S \in (0,\infty), \quad t \in (0,T) \\ V(S,T) = f(S), \quad S \in (0,\infty). \end{cases}$$

where σ is the (constant) volatility, r the (constant) interest rate, q the dividend yield and S is the underlying value.

For a European call, i.e. $V(S,T) = \max(S-K,0)$, a closed formula exists. It is given by

$$V(S,t) = Se^{-q(T-t)}N(d_1) - Ke^{-r(T-t)}N(d_2), \tag{3.1}$$

$$d_1 = \frac{\log(S/K) + (r - q + \frac{1}{2}\sigma^2)}{\sigma(T-t)}, \tag{3.2}$$

$$d_2 = d_1 - \sigma\sqrt{T-t}. \tag{3.3}$$

where $N(\cdot)$ is the cumulative normal distribution.

The closed form expression will be used to validate numerical results.

3.1 Implicit methods

Solving (BS) with an implicit method using the "PDE" boundary condition is done by discretizing in space after which the semi-discretization problem is integrated in time.

Discretizing in space is done by using the central scheme for the convection and the three point stencil for the diffusion. The discretization of the boundary points depends on the accuracy that is required at the boundaries. For time integration the ω-scheme [13] is used. This scheme can be compactly rewritten as

$$\mathbf{V}^{n+1} = (I + \Delta\omega A)^{-1}((I - \Delta(1-\omega)A)\mathbf{V}^n), \quad \omega = 0.51.$$

As an example a European call/put option with parameters

$$r = 0.05, \quad \sigma = 0.5, \quad q = 0, \quad K = 100, \quad T = 5, \tag{3.4}$$

will be considered. For the discretization of the boundary condition we use the following approximations for the derivatives which occur in the "PDE" boundary condition:

$$\begin{aligned} (pde1) &: \frac{\partial u}{\partial x} : O(h^2) \quad , \quad \frac{\partial^2 u}{\partial x^2} : O(h), \\ (pde2) &: \frac{\partial u}{\partial x} : O(h) \quad , \quad \frac{\partial^2 u}{\partial x^2} : O(h^2), \\ (pde3) &: \frac{\partial u}{\partial x} : O(h) \quad , \quad \frac{\partial^2 u}{\partial x^2} : O(h). \end{aligned}$$

The behavior of the European call and put are shown in Figures 3.1, 3.3, 3.5, and Figures 3.2, 3.4, 3.6 respectively, for uniform grids with increasing S_{max}. It can be seen that when S_{max} increases the numerical solution (solid blue line) converges toward the exact solution (dashed black line). Hence, the boundary conditions are of minimal influence to the internal solution. However, if S_{max} is not taken large enough, the numerical solution deviates significantly from the exact solution. Since the discretization is non-legitimate, which means that there can occur both positive and negative eigenvalues, the implicit method with the "PDE" boundary conditions does not solve our problem adequately. Therefore, experiments with explicit methods will be discussed in the next section.

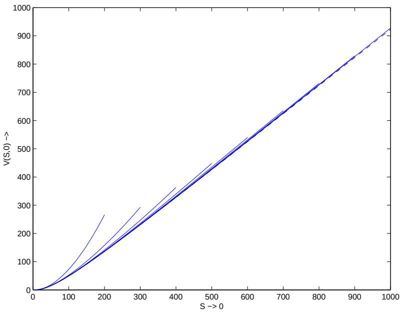

Figure 3.1: Call option. Solid blue line: Numerical solution with pde3. Dashed black line: Exact solution.

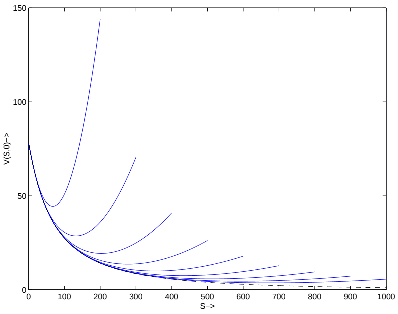

Figure 3.2: Put option. Solid blue line: Numerical solution with pde3. Dashed black line: Exact solution.

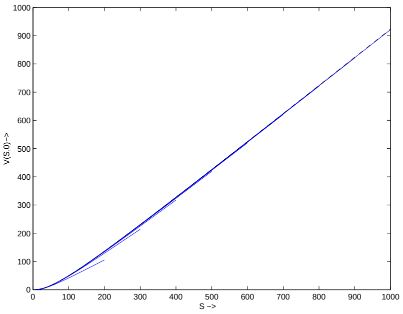

Figure 3.3: Call Option. Solid blue line: Numerical solution with pde2. Dashed black line: Exact solution.

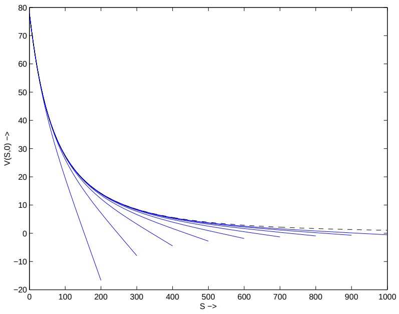

Figure 3.4: Put option. Solid blue line: Numerical solution with pde2. Dashed black line: Exact solution.

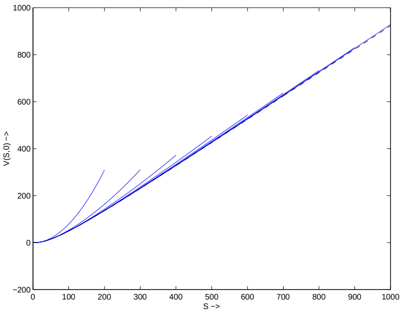

Figure 3.5: Call option. Solid blue line: Numerical solution with pde1. Dashed black line: Exact solution.

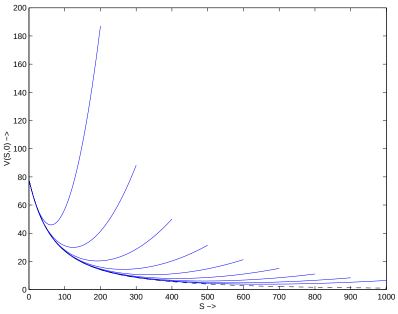

Figure 3.6: Put option. Solid blue line: Numerical solution with pde1. Dashed black line: Exact solution.

3.2 Explicit methods

In the previous chapter three explicit approaches EulTree, RKCTree and RKCTreeImpr were suggested to solve an initial value problem. In this chapter the three approaches will be applied to solve the Black–Scholes equation (BS).

3.2.1 Stability restrictions for the Euler method

The (BS) initial value problem can be spatially discretized using the second order stencil for the first and second derivative as discussed in Chapter 2. For the time integration the explicit Euler method is used. The discounting term '$-rV$' will be treated implicitly. Combining this gives the following recursive equation for the time integration

$$(1-r\Delta t)V_i^{n+1} = [p_{i,i-1}V_{i-1}^n + p_{i,i}V_i^n + p_{i,i+1}V_{i+1}^n], \quad n=1,..,Q, \quad i=1,..,N+1, \tag{3.5}$$

with

$$\begin{aligned} p_{i,i-1} &= \Delta t(\frac{(r-q)S_i\Delta S_i}{(\Delta S_{i-1}+\Delta S_i)\Delta S_{i-1}} - \frac{\sigma^2 S_i^2}{\Delta S_{i-1}(\Delta S_{i-1}+\Delta S_i)}), \\ p_{i,i} &= 1+\Delta t\frac{(r-q)S_i(\Delta S_{i-1}-\Delta S_i)}{\Delta S_i \Delta S_{i-1}} + \Delta t\frac{\sigma^2 S_i^2}{\Delta S_{i-1}\Delta S_i}, \\ p_{i,i+1} &= -\Delta t(\frac{(r-q)S_i\Delta S_{i-1}}{(\Delta S_{i-1}+\Delta S_i)\Delta S_i} + \frac{\sigma^2 S_i^2}{\Delta S_i(\Delta S_{i-1}+\Delta S_i)}). \end{aligned}$$

Trying to solve the Black–Scholes equation straightforward with EulTree or RKCTree gives rise to several problems. Since the problem is solved explicitly, the eigenvalues of the discretization matrix must lie in a stability region depending on the method used.

Applying Theorem 2.1 on a uniform grid, we obtain

$$0 \geq \Delta t \geq -\frac{\Delta S^2}{\sigma^2 S_i^2}, \qquad \text{and} \qquad \frac{|r-q|\Delta S}{\frac{1}{2}\sigma^2 S_i^2} \leqslant 2. \tag{3.6}$$

For large S the equation is dominated by the diffusion term and the full discretization matrix becomes very stiff. As can be easily seen from (3.6), this will lead to extremely small time steps Δt, or, equivalently, to a large number of time levels Q.

The choice of the grid influences the spectrum of the semi-discretization matrix A. For some specific grid definitions, the eigenvalues are such that they imply relatively weaker restrictions for the time step. We will discuss this property with an example. It is widely known that the Black–Scholes equation can be transformed with the transformation

$$S = e^x. \tag{3.7}$$

A transformation of the equation itself is less elegant, since this also requires a transformation of the initial value (payoff). Therefore, rather than transforming the equation itself, the underlying grid is transformed. Then we have the following stability result.

Theorem 3.1 *Let Q be the number of time levels, and let $N+1=2Q+1$ be the number of grid nodes. Define $x_0=\log(S_0)$. The uniform grid with center x_0, denoted by $\{x_i\}_{i=1}^{N+1}$, is given by*

$$x_i = x_0 + \Delta x(i-1-\frac{N}{2}), \quad i=1,..,N+1, \tag{3.8}$$

and

$$\begin{aligned} S_i &= e^{x_0+\Delta x(i-1-N/2)}, \\ \Delta S_i &= S_{i+1}-S_i = e^{x_0+\Delta x(i-N/2)}(1-e^{-\Delta x}). \end{aligned} \tag{3.9}$$

Furthermore, let $(r-q) - \frac{\sigma^2}{(e^{\Delta x}-1)^2 e^{-\Delta x}} < 0$. *Then scheme (3.5) is stable if*

$$\frac{1}{(r-q) - \dfrac{\sigma^2}{(e^{\Delta x}-1)^2 e^{-\Delta x}}} \leq \Delta t \leq 0, \tag{3.10}$$

with the restriction that

$$\log(\frac{r-q}{\sigma^2 + (r-q)}) \leq \Delta x \leq \log(\frac{\sigma^2 + r - q}{r-q}). \tag{3.11}$$

Proof. Substitution of (3.9) into (3.5) and using Theorem 2.1 gives for p_i

$$1 - \Delta t(r-q) + \Delta t \frac{\sigma^2}{(e^{\Delta x}-1)^2 e^{-\Delta x}} \geq 0,$$

which immediately gives (3.10). The upper and lower off-diagonal elements are treated similar. Applying the transformation and requiring that the off-diagonal elements must be positive immediately gives (3.11). ■

Thus for a fixed Δx satisfying (3.11), the number of time levels Q follows from (3.10) with $\Delta t = -\frac{T}{Q}$, i.e.

$$Q \geq -T \left[(r-q) - \frac{\sigma^2}{(e^{\Delta x}-1)^2 e^{-\Delta x}} \right]. \tag{3.12}$$

One could also obtain the grid the other way around by choosing a Q and then calculate the appropriate Δx, see appendix A.

Applying the grid transformation to the problem ensures that the spectrum is good enough to apply the EulerTree and RKCTree. The grid transformation is, however, very specific and since the spatial steps near the spot price can become large, accuracy might be lost. Therefore the RKCTreeImpr was introduced. In the next section the three methods will be compared.

3.2.2 Numerical results for a European call

We start by examining a simple European call option with the parameters

$$r = 0.1, \quad q = 0, \quad K = 100, \quad S_0 = 100, \quad \sigma = 0.25, \quad T = 1. \tag{3.13}$$

This example is an initial value problem (BS) with payoff $V(S,T) = \max(S-K, 0)$. By applying the grid transformation (3.7) the spectrum of the semi-discretization matrix is such that EulTree can be applied. The same transformation will be applied for all the other methods and results are shown in Table 3.1. All three methods converge toward the exact solution. In Figure 3.7 and 3.9, Euler Tree and RKCTreeImpr are compared to resp. FullEuler and FullRKC. In Figure 3.8 the value of the option with a scalings factor is plotted and it is shown that the solution is solved back toward one point. It is seen that the solutions are stable, which is exactly what we expect from stability analysis. It is also shown that the solutions exactly match. The important remark has to be made here that RKCTreeImpr is not solved back to one point, but to an interval, since it was mentioned that unstable solutions could be obtained if the remaining interval is not large enough. By setting $N = 2Q + 10$ the results will be as accurate as by solving on the full grid. We will see later on what happens if the remaining grid (this is the grid that is still left at $t = 0$) is not taken large enough.

However, although the solution is stable and the converges toward the exact solution, the rate of convergence is not satisfying. This is due to the exponential grid. The spatial intervals around the point of interest are too large, and this is exactly where the payoff is not differentiable. To deal with non-smooth payoffs, we would like to have a grid refinement located at the non-smooth part of the payoff. We will first focus on a uniform grid to see if the methods still work. Then we will try to apply the methods to refined grids. For the grid refinement we follow [14, Eqn. (4.39) with $\mu = 1$].

If a uniform grid is used then it can be seen that the spectrum of the semi-discretization matrix will be such that EulTree and RKCTree cannot be applied anymore. RKCTreeImpr could still

N	Q	FullEuler	EulTree	s	FullRKC	RKCTree
50	25	14.9105	14.9105	2	14.851	14.851
100	50	14.9431	14.9431	2	14.9141	14.9141
200	100	14.9595	14.9595	2	14.9451	14.9451
400	200	14.9676	14.9676	2	14.9605	14.9605
800	400	14.9717	14.9717	2	14.9681	14.9681
1600	800	14.9737	14.9737	2	14.972	14.972
3200	1600	14.9748	14.9748	2	14.9739	14.9739
N	Q	s	RKCTreeImpr			
			linear	quadratic	cubic	
50	25	2	14.851	14.851	14.851	
100	50	2	14.9141	14.9141	14.9141	
200	100	2	14.9451	14.9451	14.9451	
400	200	2	14.9605	14.9605	14.9605	
800	400	2	14.9681	14.9681	14.9681	
1600	800	2	14.972	14.972	14.972	
3200	1600	2	14.9739	14.9739	14.9739	

Table 3.1: Comparison of 3 approaches on an exponential grid. The exact solution to the problem is 14.9758.

be applied, but at the cost of taking a very small time step. This increases the computational time significantly and therefore this method will be omitted. Therefore we will only focus on RKCTreeImpr, and in particular the linear case since quadratic and cubic extrapolation are shown to give the same answers. The results for RKCTreeImpr on a uniform grid are shown in Table 3.2. From this table it can be concluded that it is very important as to what to choose as the remaining interval. If the remaining interval is taken too small, then inaccurate answers are obtained. In Figure 3.10 and 3.11 it is shown more precisely how the size of the grid influences the solution. It can be seen that if the remaining interval is not taken large enough, the linear extrapolation is not a good approximation and leads to inaccurate solutions. It was shown that only RKCTreeImpr can be appplied to solve (BS) on a uniform grid. However, it might be more convenient if a grid refinement could be applied to capture the non-smooth behavior of the payoff. Refining the grid leads to very stiff problems and therefore EulTree and RKCTree can also not be applied to solve this problem.

N	Q	s	FullRKC	RKCTreeImpr	s	extra points	
				A			*B*
50	25	4	14.8816	14.7599	4	8	14.8742
100	50	6	14.9357	14.8604	6	16	14.9344
200	100	7	14.9510	14.8927	8	32	14.9472
400	200	10	14.9595	14.9818	11	64	14.9622
800	400	14	14.9697	14.8366	16	128	14.9693
1600	800	19	14.9724	14.8037	22	280	14.9725
3200	1600	27	14.9742	14.85	31	600	14.9742

Table 3.2: RKCTreeImpr on uniform grids. *A*: Solving back to one point. *B*: Solving back to large grid. The exact solution to the problem is 14.9758.

In Table 3.3 the results for the refined grid are shown. It is seen that solving back to one point gives once again inaccurate answers. To solve this problem a number of points is added such that the remaining interval is large enough. This large number of extra points increases the computational time and is therefore undesirable. Figures 3.12 and 3.13 illustrate what the influence of the linear extrapolation at the endpoints is. Solving (BS) on the refined grid can only be done if a lot of extra points is added, which will slow down the method.

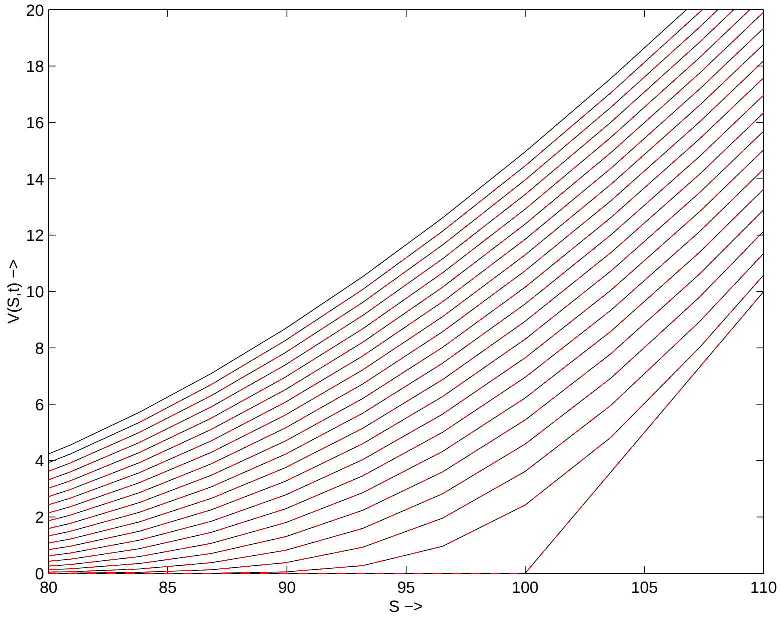

Figure 3.7: European call option with $Q = 100$. Black line: FullEuler. Red line: EulerTree. Solution is shown for every 5 timesteps.

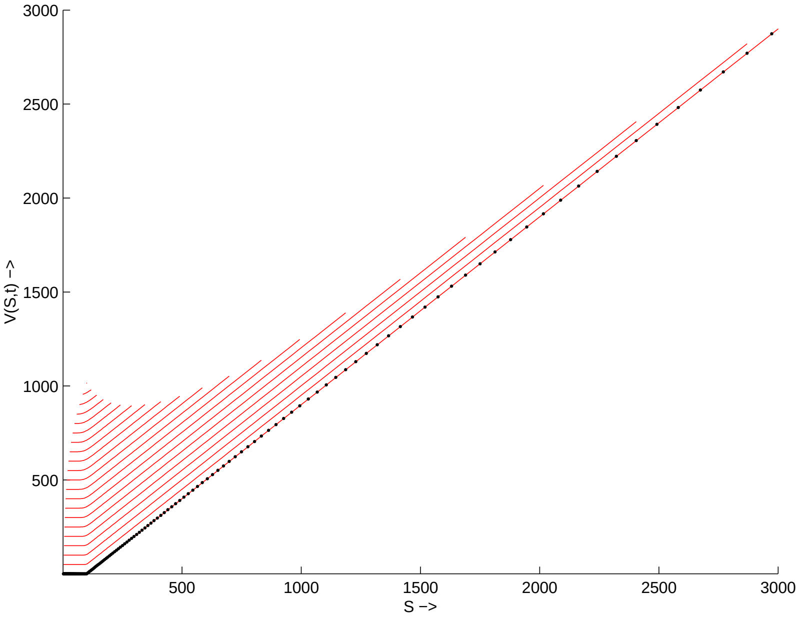

Figure 3.8: European call option with EulerTree and $Q = 100$. Solution is shown for every 5 timesteps.

3.3 Conclusion

The Black–Scholes equation can be solved as an initial value problem with EulTree, RKCTree and RKCTreeImpr on an exponential grid. However, this choice of grid is too restricted in view of accuracy of the method and therefore other grids were tried: uniform and mesh-refined grids. It is shown that only RKCTreeImpr can handle these grids. When applying RKCTreeImpr, it is important that the remaining interval (this is the spatial interval which is left at $t = 0$) is not too small. So instead of solving back to a single point, the problem is solved back to an interval. If the remaining interval is too small inaccurate answers are obtained.

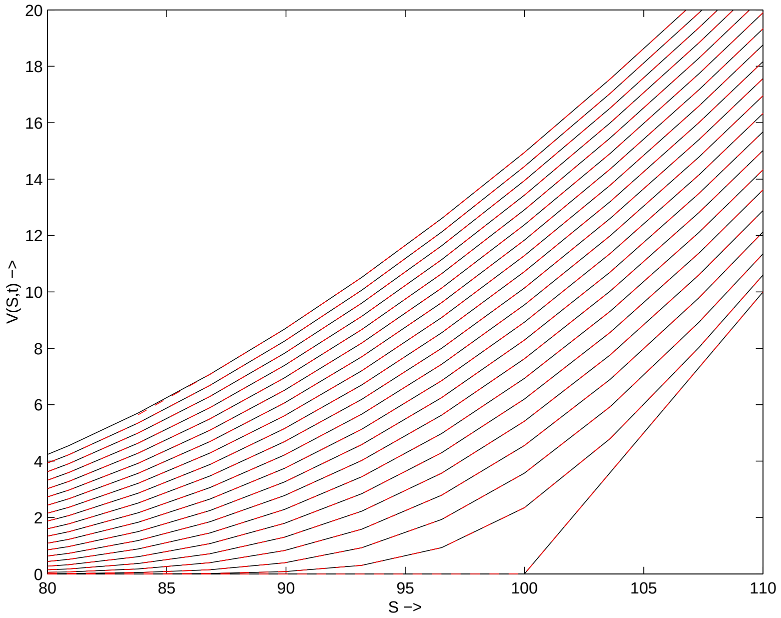

Figure 3.9: European call option with $Q = 100$. Black line: FullRKC. Red line: RKCTreeImpr. Solution is shown for every 5 timesteps.

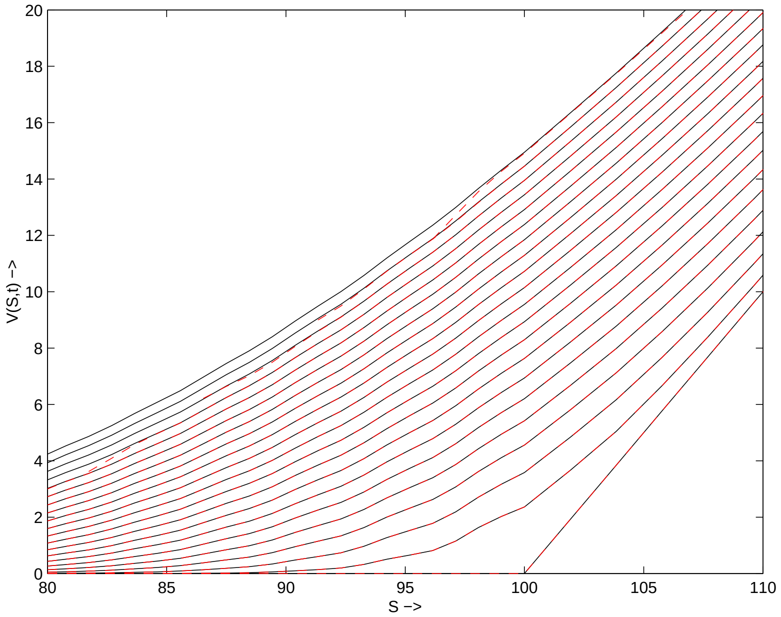

Figure 3.10: European call option on a uniform grid with $Q = 100$ and $N = 200$. Black line: FullRKC. Red line: RKCTreeImpr.

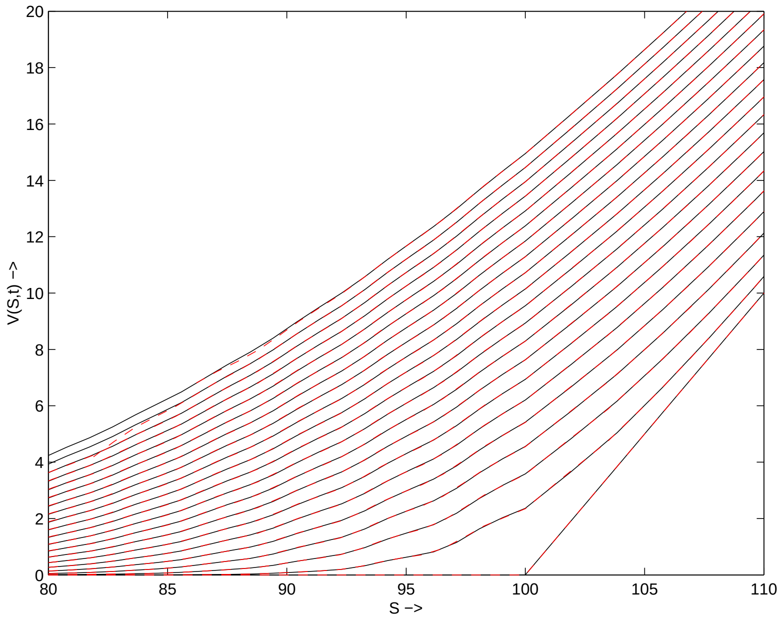

Figure 3.11: European call option on a uniform grid with $Q = 100$ and $N = 200 + 32$. Black line: FullRKC. Red line: RKCTreeImpr.

N	Q	s	FullRKC	RKCTreeImpr	s	extra points for starting grid	
				A			B
50	25	40	14.8703	14.4551	114	90	14.8497
100	50	58	14.9125	7.8514	162	180	14.9112
200	100	82	14.9476	10.7895	236	376	14.9402
400	200	116	14.9640	13.2453	333	750	14.9548
800	400	164	14.9693	12.6115	470	1500	14.9615
1600	800	232	14.9727	12.7680	665	3000	14.9649
3200	1600	327	14.9742	12.8546	940	6000	14.9667

Table 3.3: RKCTreeImpr on refined grids. A: Solving back to one point. B: Solving back to a larger grid. The exact solution to the problem is 14.9758.

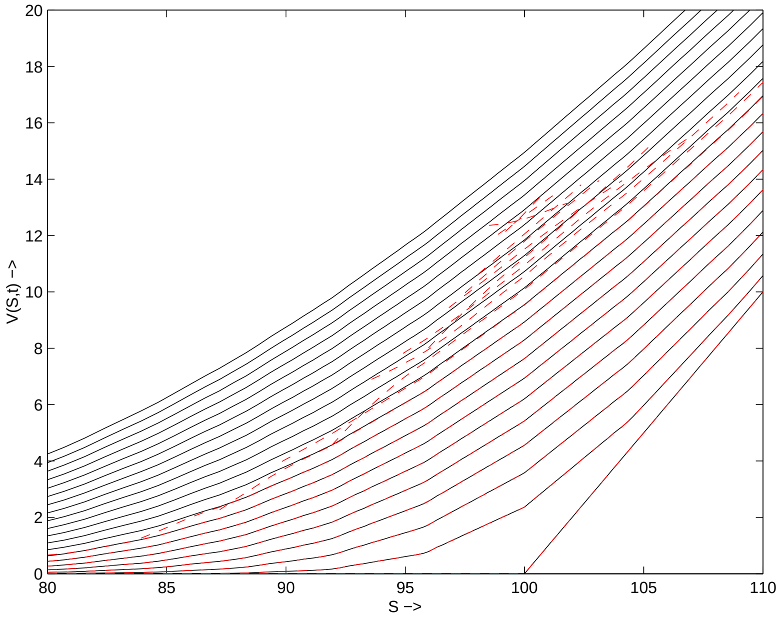

Figure 3.12: European call option on a refined grid with $Q = 100$ and $N = 200$. Black line: FullRKC. Red line: RKCTreeImpr. Solution is shown for every 5 timesteps.

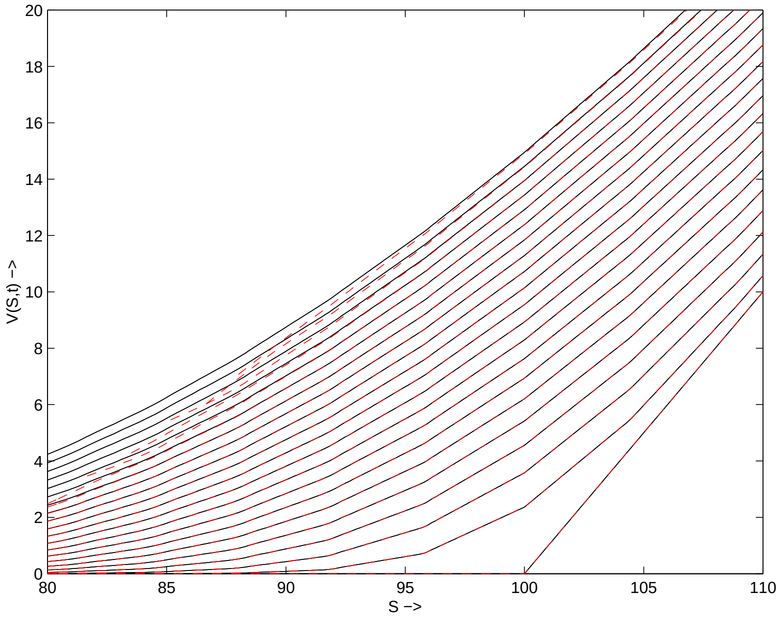

Figure 3.13: European call option on a refined grid with $Q = 100$ and $N = 200 + 376$. Black line: FullRKC. Red line: RKCTreeImpr. Solution is shown for every 5 timesteps.

Chapter 4

Hull–White model

The interest rate under the Hull–White model follows the stochastic differential equation (1.11). Suppose a security $V(r,t)$ pays $F(r)$ at T, then $V(r,t)$ solves the following one-factor partial differential equation

$$\begin{cases} \dfrac{\partial V}{\partial t} + (\theta(t) - ar)\dfrac{\partial V}{\partial r} + \dfrac{1}{2}\sigma_r^2 \dfrac{\partial^2 V}{\partial r^2} - rV = 0, \quad r \in (-\infty, \infty), \quad t \in (0,T), \\ V(r,T) = f(r), \quad r \in (-\infty, \infty). \end{cases} \tag{4.1}$$

The value $P(t,T)$ of a zero-coupon bond maturing at time T depends on the interest rate r and satisfies the above-mentioned partial differential equation with payoff $F(r) = 1$. An exact formula can be derived (see Appendix B), which is given by

$$\begin{aligned} P(t,T) &= e^{A(t)+rB(t)}, && (4.2)\\ B(t) &= \frac{-\int_t^T e^{\int_u^T a(s)ds} du}{e^{\int_t^T a(s)ds}}, && (4.3)\\ A(t) &= \int_t^T [\theta(s)B(s) + \frac{1}{2}\sigma_r^2(s)B^2(s)]ds. && (4.4) \end{aligned}$$

The solution of the zero coupon bond will be used to transform the one-factor Hull–White equation.

As another example we will take a caplet. A caplet is a particular type of a European option whose underlying is the curve of interest rates. It can also be seen as a call option on the short rate. This will be explained some more.

Given that at the moment we are at t, then the forward rate in the interval $[T_s, T_e]$ is given by

$$F(t, T_s, T_e) := \frac{1}{\tau}(\frac{P(t,T_s)}{P(t,T_e)} - 1).$$

where τ denotes the day count fraction. A call option on the forward rate with maturity $t = T_f$ and paying at time $t = T_e$ is given by

$$\tau \max(F(T_f, T_s, T_e) - K, 0), \quad at \quad t = T_e, \tag{4.5}$$

Shifting this back to $t = T_f$ yields

$$\tau \max(F(T_f, T_s, T_e) - K, 0)P(T_f, T_e), \quad at \quad t = T_f. \tag{4.6}$$

So the payoff of the caplet $\text{Caplet}(t, T_f, T_s, T_e, \tau, K) \equiv \mathcal{C}_{T_f,T_s,T_e,\tau,K}(t,r)$ is given by

$$\text{Caplet}(T_f, T_f, T_s, T_e, \tau, K) = \max(P(T_f, T_s) - (1+\tau K)P(T_f, T_e), 0) = \mathcal{C}_{T_f,T_s,T_e,\tau,K}(T_f, r). \tag{4.7}$$

4.1 Explicit discretizations for the Hull–White model

The Hull–White initial value problem will be spatially discretized using an upstream discretization, because the convection term can either be positive or negative, changing signs at $\theta(t) - ar = 0$, for the first derivative. We use the three-point stencil for the second derivative. For the time integration the explicit Euler method is used. We will use a uniform grid, so $\Delta r_n = \Delta r_{n-1} = \Delta r$. Combining this gives the following recursive equation for the time integration

$$V_i^{n+1} = [p_{i-1}V_{i-1}^n + p_iV_i^n + p_{i+1}V_{i+1}^n], \quad n = 1,..,Q, \quad i = 1,..,N+1, \tag{4.8}$$

with

$$\begin{aligned} p_{i,i-1} &= -\Delta t(-\frac{1}{2}(\frac{\beta_i}{\Delta r} - |\frac{\beta_i}{\Delta r}|) + \frac{f_i}{\Delta r^2}), \\ p_{i,i} &= 1 - \Delta t(-|\frac{\beta_i}{\Delta r}| - \frac{2f_i}{\Delta r^2} - r_i), \\ p_{i,i+1} &= -\Delta t(\frac{1}{2}(\frac{\beta_i}{\Delta r} + |\frac{\beta_i}{\Delta r}|) + \frac{f_i}{\Delta r^2}). \end{aligned}$$

and

$$\begin{aligned} \beta_i &= \theta^n - ar_i, \quad \theta^n := \theta(t^n),\ i = 2,..,N, \\ f_i &= \frac{1}{2}\sigma_r^2,\ i = 2,..,N. \end{aligned}$$

The discounting term '$-rV$' is taken here explicitly (see $p_{i,i}$). We disregard the term structure for θ, i.e. it is taken constant.

The $p_{i,i-1}$, $p_{i,i}$, $p_{i,i+1}$ cannot be treated as probabilities since they do not add up to 1. Therefore stability of (4.8) cannot be proved via Theorem 2.1. Numerical experiments in Subsection 4.1.1, however, show that the method is stable if a particular grid is chosen. This particular grid is obtained by taking r_{min} not too negative, since a too small r_{min} destroys the diagonal dominance of the matrix. To control the stability of the used schemes, we transform in Subsection 4.1.2 the original equation to get rid of the '$-rV$' term.

4.1.1 Numerical results for the untransformed Hull–White equation

In this section the results for a zero coupon bond (Table 4.1) with parameters

$$T = 5, \quad r_0 = 0.03, \quad a = 0.05, \quad \theta(t) = \theta = 0.025, \quad \sigma_r = 0.01, \tag{4.9}$$

and a caplet (Table 4.2) with parameters

$$r_0 = 0.03, \quad \theta(t) = \theta = 0.025, \quad a = 0.05, \quad \sigma_r = 0.01, \quad K = 0.06, \quad T_f = 5, \quad T_S = 5, \quad T_e = 6 \tag{4.10}$$

will be given for EulTree and RKCTreeImpr. Both can be applied to solve the one-factor Hull–White problem on a uniform grid defined by

$$r_i = r_0 + \Delta r(i - 1 - N/2), \qquad i = 1, \cdots, N+1, \tag{4.11}$$

with $\Delta r = \sqrt{\sigma^2 \Delta t}$. If RKCTreeImpr is applied, extra points have to be added to ensure that the remaining interval is large enough.

The tables show that the numerical solutions converge toward the exact solution. In Figures 4.1 and 4.2 the zero-coupon bond is shown solved with EulTree and RKCTreeImpr. In Figures 4.3 and 4.4 the caplet is shown solved on a full grid with Euler explicit and with EulTree.

The tree methods once again match exactly with the full grid solver. So both methods can solve the Hull–White initial value problem effectively.

N	Q	FullEuler	EulTree	s	FullRKC
50	25	0.6604	0.6604	2	0.6600
100	50	0.6595	0.6595	2	0.6594
200	100	0.6590	0.6590	2	0.6589
400	200	0.6586	0.6586	2	0.6585
800	400	0.6583	0.6583	2	0.6583
1600	800	0.6582	0.6582	2	0.6581
3200	1600	0.6580	0.6580	2	0.6580
N	Q	s	RKCTreeImpr linear		
50	25	2	0.6568		
100	50	2	0.6580		
200	100	2	0.6582		
400	200	2	0.6581		
800	400	2	0.6580		
1600	800	2	0.6579		
3200	1600	2	0.6579		

Table 4.1: Pricing of a zero coupon bond with EulTree and RKCTreeImpr on a uniform grid. The exact solution to the problem is 0.6577.

N	Q	FullEuler	EulTree	s	FullRKC
50	25	0.0532	0.0532	2	0.0515
100	50	0.0527	0.0527	2	0.0518
200	100	0.0525	0.0525	2	0.0520
400	200	0.0524	0.0524	2	0.0523
800	400	0.0525	0.0525	2	0.0524
1600	800	0.0525	0.0525	2	0.0525
3200	1600	0.0525	0.0525	2	0.0525
N	Q	s	RKCTreeImpr linear		
50	25	2	0.0515		
100	50	2	0.0518		
200	100	2	0.0520		
400	200	2	0.0523		
800	400	2	0.0524		
1600	800	2	0.0525		
3200	1600	2	0.0525		

Table 4.2: Pricing of a caplet with EulTree and RKCTreeImpr on a uniform grid. The exact solution to the problem is 0.0525.

4.1.2 Transformed Hull–White equation

For the transformation of (4.1) we write $V(r,t) = V_{sol}(r,t)\widetilde{V}(r,t)$, where $V_{sol}(r,t)$ is the solution of the zero-coupon bond under the Hull–White model and $\widetilde{V}$ is the new variable which we want to solve. Substitution in (4.1) yields

$$\begin{aligned} & V_{sol}\frac{\partial \widetilde{V}}{\partial t} + \widetilde{V}\frac{\partial V_{sol}}{\partial t} + (\theta(t) - ar)(V_{sol}\frac{\partial \widetilde{V}}{\partial r} + \widetilde{V}\frac{\partial V_{sol}}{\partial r}) \\ & + \frac{1}{2}\sigma_r^2(V_{sol}\frac{\partial^2 \widetilde{V}}{\partial r^2} + 2\frac{\partial V_{sol}}{\partial r}\frac{\partial \widetilde{V}}{\partial r} + \widetilde{V}\frac{\partial^2 V_{sol}}{\partial r^2}) - rV_{sol}\widetilde{V} = 0. \end{aligned}$$

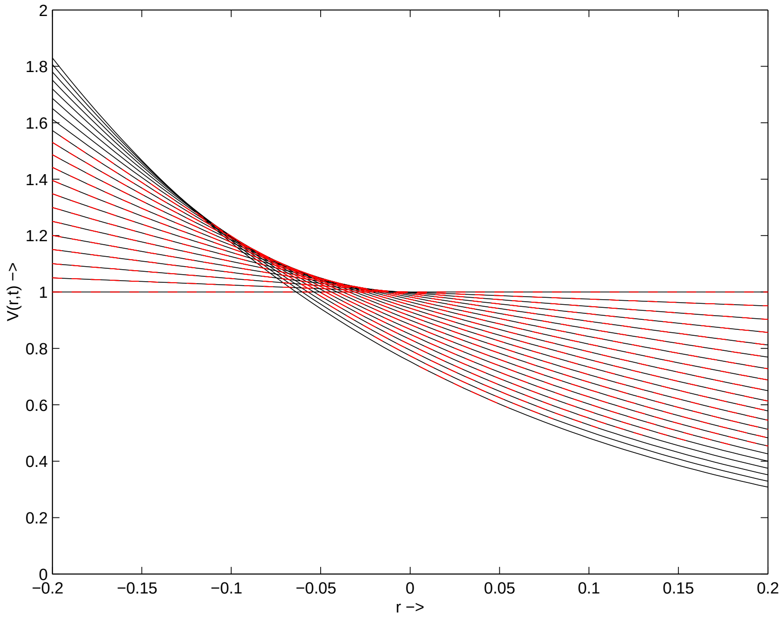

Figure 4.1: Pricing of a zero coupon bond on a uniform grid. Black line: FullEuler. Red line: EulerTree.

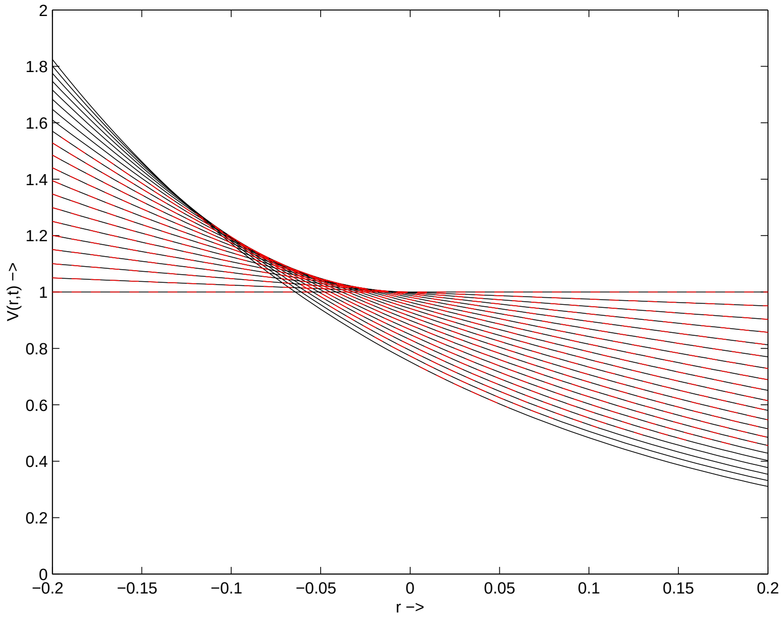

Figure 4.2: Pricing of a zero coupon bond on a uniform grid. Black line: FullRKC. Red line: RKC-TreeImpr.

Rearranging terms gives

$$\begin{aligned} & V_{sol}\frac{\partial \widetilde{V}}{\partial t} + ((\theta(t) - ar)V_{sol} + \sigma_r^2 \frac{\partial V_{sol}}{\partial r})\frac{\partial \widetilde{V}}{\partial r} + \frac{1}{2}\sigma_r^2 V_{sol}\frac{\partial^2 \widetilde{V}}{\partial r^2} \\ & + \widetilde{V}(\frac{\partial V_{sol}}{\partial t} + (\theta(t) - ar)\frac{\partial V_{sol}}{\partial r} + \frac{\partial^2 V_{sol}}{\partial r^2} - rV_{sol}) = 0. \end{aligned} \tag{4.12}$$

And since V_{sol} is the zero-coupon bond satisfying

$$\frac{\partial V_{sol}}{\partial t} + (\theta(t) - ar)\frac{\partial V_{sol}}{\partial r} + \frac{\partial^2 V_{sol}}{\partial r^2} - rV_{sol} = 0, \tag{4.13}$$

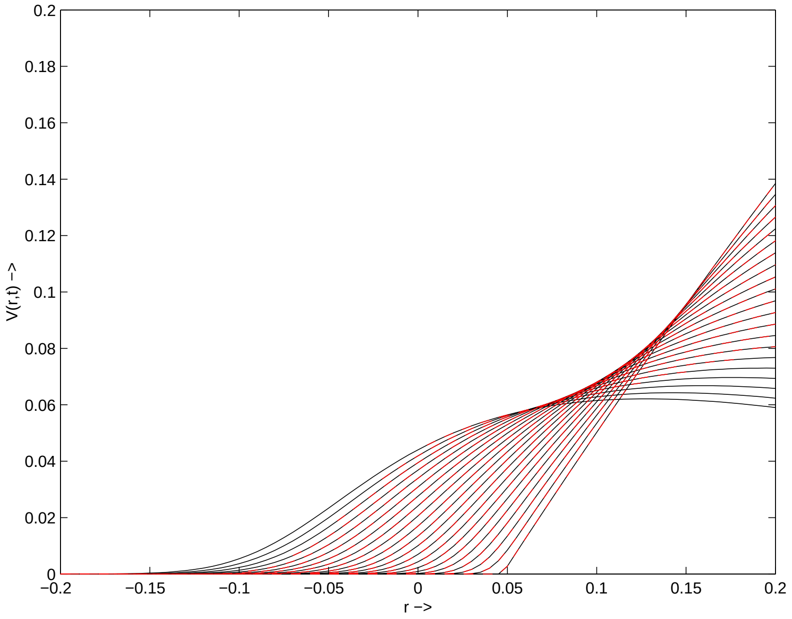

Figure 4.3: Pricing of a caplet on a uniform grid. Black line: FullEuler. Red line: EulerTree.

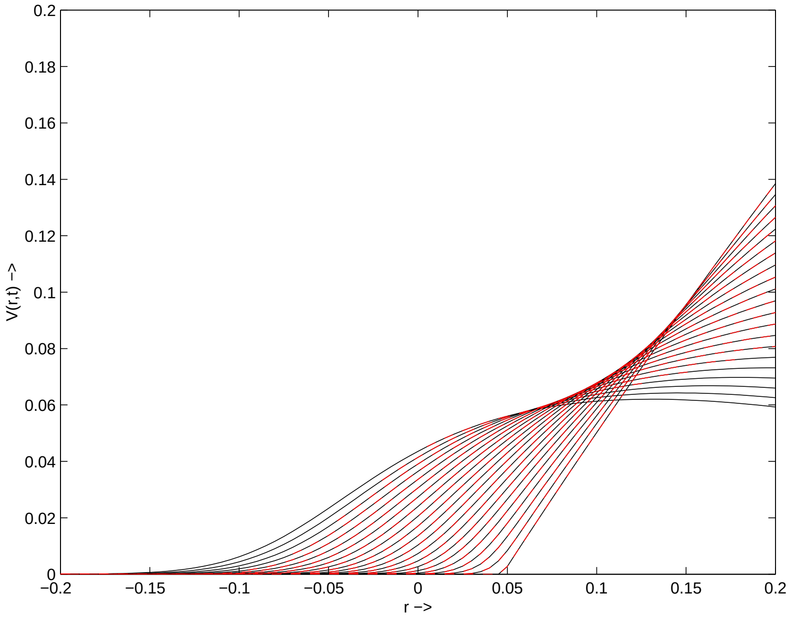

Figure 4.4: Pricing of a caplet on a uniform grid. Black line: FullRKC. Red line: RKCTreeImpr.

equation (4.12) becomes

$$V_{sol}\frac{\partial \widetilde{V}}{\partial t} + ((\theta(t) - ar)V_{sol} + \sigma_r^2 \frac{\partial V_{sol}}{\partial r})\frac{\partial \widetilde{V}}{\partial r} + \frac{1}{2}\sigma_r^2 V_{sol}\frac{\partial^2 \widetilde{V}}{\partial r^2} = 0. \tag{4.14}$$

The solution for the zero coupon bond is given by

$$V_{sol}(r,t) = e^{A(t)+rB(t)},$$

and substituting this in (4.13) leads to the convection-diffusion equation

$$\frac{\partial \widetilde{V}}{\partial t} + (\theta(t) - ar + \sigma_r^2 B(t))\frac{\partial \widetilde{V}}{\partial r} + \frac{1}{2}\sigma_r^2 \frac{\partial^2 \widetilde{V}}{\partial r^2} = 0. \tag{4.15}$$

This transformation is applied to get rid of the '$-rV$' term, in order to obtain a better spectrum for the semi-discretization matrix. We would like to derive sufficient conditions for stability by using Theorem 2.1. The theorem states that if the diagonal and off-diagonals elements are positive then the method is stable. Due to the upwind discretization the off-diagonals are always positive. However, it is hard to proof positiveness for the diagonal elements, but in practice it is seen that stability is easily obtained.

4.1.3 Numerical results for caplets

In Table 4.3 the exact and numerical solution of a caplet under the untransformed and transformed Hull-White equation are compared for EulTree. It can be seen that the results of the transformed and untransformed equation do not differ much.

T_f	T_S	T_e	exact	untransformed	transformed
5	5	6	0.0440	0.0441	0.0444
5	5	10	0.2958	0.2968	0.2939
10	10	11	0.0459	0.0462	0.0461
10	10	15	0.2016	0.2030	0.2008
15	15	16	0.0207	0.0207	0.0207
15	15	20	0.0770	0.0765	0.0760
20	20	21	0.0062	0.0059	0.0060
20	20	25	0.0210	0.0198	0.0201
30	30	31	0.0003	0.0002	0.0002

Table 4.3: EulTree on a uniform grid under the untransformed equation. Number of time steps is 200. The point of interest is $r_0 = 0$.

4.2 Conclusion

It was shown in this chapter that both EulTree and RKCTreeImpr can solve the one-factor Hull-White model on a uniform grid as effectively as FullEuler and FullRKC. However, a special grid, in which r_{min} is not taken too negative, had to be taken, since a too negative r_{min} destroys the diagonal dominance of the matrix, which leads to an unstable method. To control the spectrum of the semi-discretization matrix a transformation was proposed, which scaled out the '$-rV$' term. It was, however, shown that the results for the untransformed and transformed equation were almost similar. Therefore the transformation does not need to be applied, as long as the grid is chosen in a special way.

Part II

Two-factor models

Chapter 5

Black–Scholes model

In this chapter the results for the two-factor Black–Scholes model will be discussed. We will give an answer to the question if it is possible to apply EulTree,RKCTree and RKCTreeImpr to solve this initial value problem. The most important question when higher-dimensional models are treated is what the influence of the correlation term ρ will be.

The two-dimensional Black–Scholes equation is given by

$$\begin{cases} \dfrac{\partial V}{\partial t} + \dfrac{1}{2}\sigma_1^2 S_1^2 \dfrac{\partial^2 V}{\partial S_1^2} + \sigma_1\sigma_2 S_1 S_2 \rho_{12} \dfrac{\partial^2 V}{\partial S_1 S_2} + \dfrac{1}{2}\sigma_2^2 S_2^2 \dfrac{\partial^2 V}{\partial S_2^2} \\ \qquad +(r_1 - q_1) S_1 \dfrac{\partial V}{\partial S_1} + (r_2 - q_2) S_2 \dfrac{\partial V}{\partial S_2} - rV = 0, \qquad S_1,\ S_2 \in (0,\infty), \quad t \in (0,T), \\ V(S_1,S_2,T) = f(S_1,S_2), \quad S_1,\ S_2 \in (0,\infty). \end{cases} \tag{5.1}$$

For an exchange option (which will be taken as an example throughout this chapter) with payoff formula $f(S_1,S_2) = \max(\alpha_1 S_1 - \alpha_2 S_2, 0)$, a closed formula exists. It is given by (see Appendix C)

$$V(S_1,S_2,t) = \alpha_1 S_1 e^{-q_1(T-t)} N(d_1') - \alpha_2 S_2 e^{-q_2(T-t)} N(d_2'), \tag{5.2}$$

where

$$\begin{aligned} d_1' &= \frac{\log(\frac{\alpha_1 S_1}{\alpha_2 S_2}) + (q_2 - q_1 + \frac{1}{2}\sigma'^2)(T-t)}{\sigma'\sqrt{T-t}}, \\ d_2' &= \frac{\log(\frac{\alpha_1 S_1}{\alpha_2 S_2}) + (q_2 - q_1 - \frac{1}{2}\sigma'^2)(T-t)}{\sigma'\sqrt{T-t}}, \\ \sigma' &= \sqrt{\sigma_1^2 - 2\rho_{12}\sigma_1\sigma_2 + \sigma_2^2}. \end{aligned}$$

The closed form formula (5.2) will be again used as a benchmark for the numerical results.

5.1 Numerical results for an exchange option

All first and second derivatives are discretized with respectively the central scheme and the three point stencil. The cross derivatives are discretized with the nine-point stencil.

In Table 5.1 the three methods EulTree,RKCTree and RKCTreeImpr using an exponential grid are shown. The parameters for this problem are

$$\begin{aligned} q_1 &= 1, \quad \sigma_1 = 0.25, \quad D_1 = 0.07, \quad S_{01} = 105, \\ q_2 &= 2, \quad \sigma_2 = 0.2, \quad D_2 = 0.05, \quad S_{02} = 50, \\ \rho_{12} &= 0.5, \quad r = 0.1, \quad T = 5. \end{aligned} \tag{5.3}$$

Table 5.1 shows that the numerical solutions converge toward the exact solution. In Figures 5.1 and 5.2 EulerTree and RKCTreeImpr are depicted for different time levels. For both EulerTree and RKCTreeImpr, the solutions stay stable over the life time of the option.

These calculations can once again also be done for a uniform and refined grid. For these grids only the RKCTreeImpr can be applied to solve this problem. If it is taken into account that the remaining interval stays large enough then these grids also produce accurate answers, see Table 5.2. It follows evidently from the table that the convergence rate is influenced by the number of extra points that is taken into account. Or to formulate it otherwise, the remaining interval should be taken large enough to obtain good approximations. So it is very important for the uniform and refined grid that enough extra points are added and since it is sometimes not clear how many extra points should be taken into account and since the exponentially stretched grid is less influenced, our focus will mainly lie on the exponentially stretched grid.

N	Q	EulTree	s	FullRKC	RKCTree
50	25	13.3687	3	13.3641	13.3641
100	50	13.4798	3	13.4241	13.4241
200	100	13.4757	3	13.4539	13.4539
400	200	13.4816	3	13.4696	13.4696
800	400	13.4827	3	13.4767	13.4767
N	Q	s	RKCTreeImpr		
			linear	quadratic	cubic
50	25	3	13.2617	13.3657	13.3640
100	50	3	13.3745	13.4244	13.4241
200	100	3	13.4296	13.4540	13.4539
400	200	3	13.4575	13.4696	13.4696
800	400	3	13.4707	13.4767	13.4767

Table 5.1: Comparison of the three approaches. The exact solution to the problem is 13.4841.

N	Q	s	RKCTreeImpr
			linear
140	25	17	13.4967
240	100	20	13.4028
300	100	25	13.4859

Table 5.2: RKCTreeImpr on a uniform grid.

5.2 Conclusion

In this chapter the extension to higher-dimensional models is made. It is shown that on an exponential grid, all three approaches EulTree,RKCTree and RKCTreeImpr can be applied. However, if a uniform or refined grid is used then only RKCTreeImpr can be applied. This is similar to the one-factor model and therefore exactly what we would expect.

However, the most important parameter in the higher-dimensional models is the correlation ρ. Eigenvalue analysis shows that positive eigenvalues occur if ρ is taken too large in absolute value ($\rho = -1, 1$). These positive eigenvalues are not very large and as a consequence the method will only become unstable if the maturity T is very large (correlation 1, T = 20). This kind of option are very unusual. However, EulTree,RKCTree and RKCTreeImpr seem to work good if ρ is not taken too large and maturities not too big.

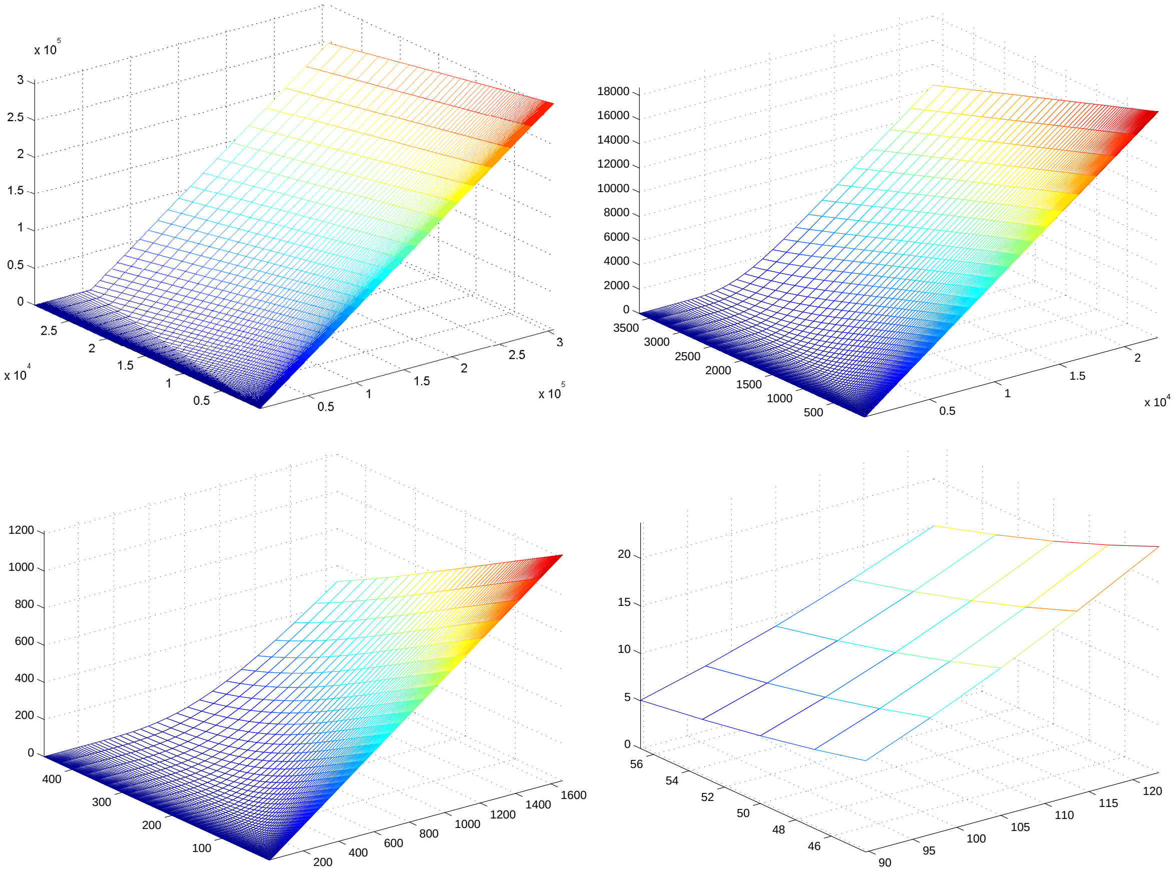

Figure 5.1: Value of a two-factor Black–Scholes call option using EulTree at $t = 5, 3\frac{1}{3}, \frac{5}{3}, 0$.

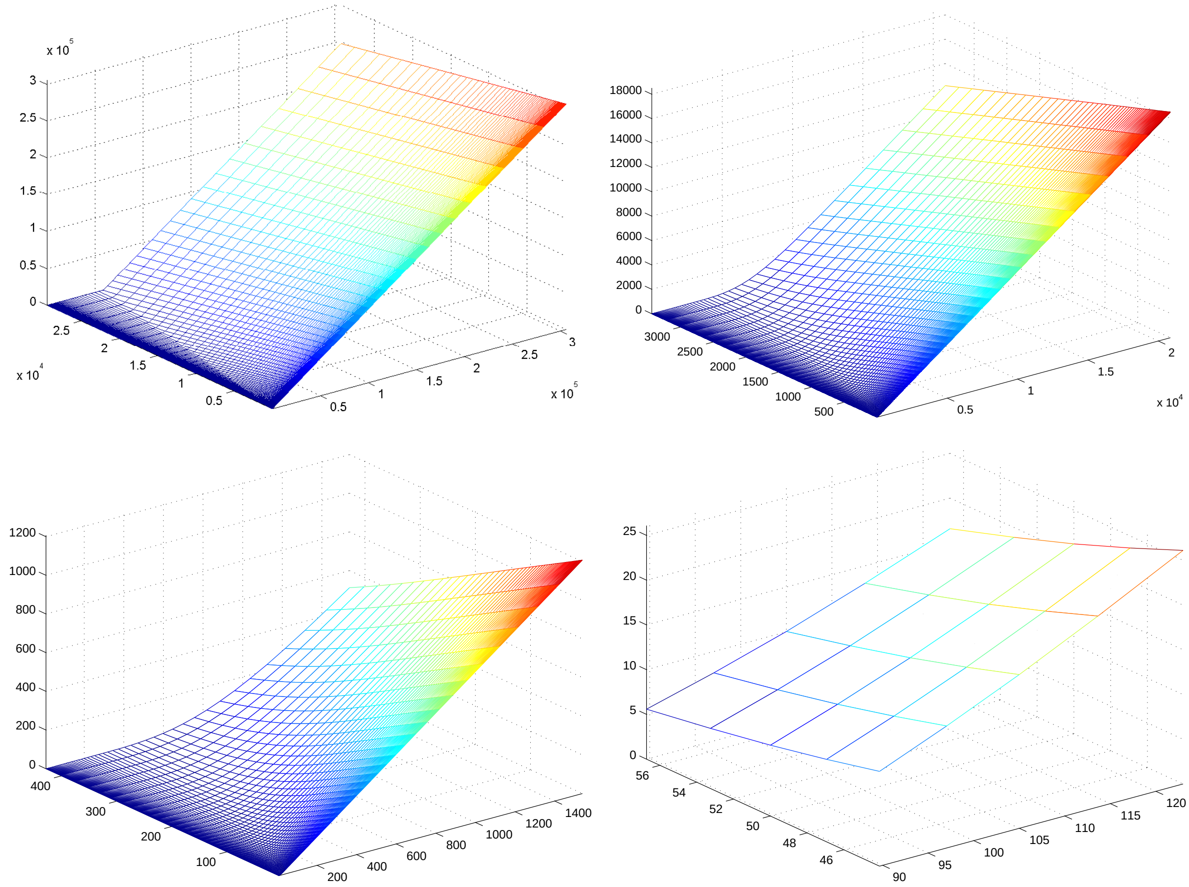

Figure 5.2: Value of a two-factor Black–Scholes call option using RKCTreeImpr with linear extrapolation at $t = 5, 3\frac{1}{3}, \frac{5}{3}, 0$.

Chapter 6

Hull–White model

In the previous chapter the two-factor Black–Scholes equation was treated. It was shown that all three approaches EulTree,RKCTree and RKCTreeImpr could be applied on an exponential grid to solve the problem. It was also discussed what the influence of the correlation term was. In this chapter we will discuss all these things for the two-factor Hull–White model.

The two-dimensional Hull–White equation is given by (H–W) in Chapter 1. For a European call, i.e. $f(S,r) = \max(S-K,0)$ with strike K, the closed formula is given by

$$\begin{aligned} V(S,r,t) = S_0 \exp(-q(T-t)) N(\frac{\log(\frac{S_0}{KP(t,T)}) - q(T-t) + \frac{\frac{1}{2}}{\gamma^2}}{\gamma}) - \\ KP(t,T) N(\frac{\log(\frac{S_0}{KP(t,T)}) - q(T-t) - 0.5\gamma^2}{\gamma}), \end{aligned} \tag{6.1}$$

with

$$\begin{aligned} \beta &= \frac{\sigma_r^2}{a^2}(T-t+\frac{2}{a}\exp(-a(T-t)) - (\frac{1}{2a})\exp(-2a(T-t)) - \frac{3}{2a}), \\ \gamma &= \sqrt{\beta + \sigma^2(T-t) + \frac{2\rho(\sigma_r\sigma)}{a}(T-t-\frac{1}{a}(1-\exp(-a(T-t))))}. \end{aligned}$$

This example will be taken as a benchmark for our numerical results.

6.1 Numerical results for a European call option

In the S-direction we will use an exponential grid. The first derivative in the S-direction is discretized using central differences. In the r-direction a uniform grid is used, where r_{min} (see (4.11) for the grid definition) is chosen in such a way that it prevents the semi-discretization from losing diagonal dominance. The first derivative in the r-direction is discretized using an upwind method. In both directions, the second derivative is discretized using a three-point stencil. The crossterm is discretized via a nine-point stencil as discussed in Chapter 2.

Consider a European call option with the following parameters:

$$T = 5, \quad q = 0, \quad r_0 = 0.03, \quad S_0 = 100, \quad a = 0.05, \quad \theta(t) = \theta = 0.025, \quad \sigma_r = 0.01, \quad \rho = 0.5. \tag{6.2}$$

In Table 6.1 it can be seen that all three approaches converge toward the exact solution. They all converge equally fast and toward the same solution, so based on this table we can not conclude which one is better. In Figures 6.1 and 6.2 it is shown that the solutions are stable.

Note that this problem can also be solved on a uniform or refined grid using RKCTreeImpr, see Table 6.2 for some results for the uniform grid. It follows, as with the two-factor Black–Scholes, that the remaining grid should be taken large enough.

N	Q	EulTree	s	FullRKC	RKCTree
50	25	40.3828	2	40.3866	40.3866
100	50	40.551	2	40.5490	40.5490
200	100	40.6332	2	40.6314	40.6314
400	200	40.6743	2	40.6733	40.6733
800	400	40.6951	2	40.6946	40.6946
N	Q	s	RKCTreeImpr		
			linear	quadratic	cubic
50	25	35	40.3617	40.3846	40.3846
100	50	47	40.5304	40.5485	40.5485
200	100	63	40.6256	40.6313	40.6313
400	200	90	40.6703	40.6733	40.6733
800	400	120	40.6933	40.6946	40.6946

Table 6.1: Four approaches compared. The exact solution to the problem is 40.7172.

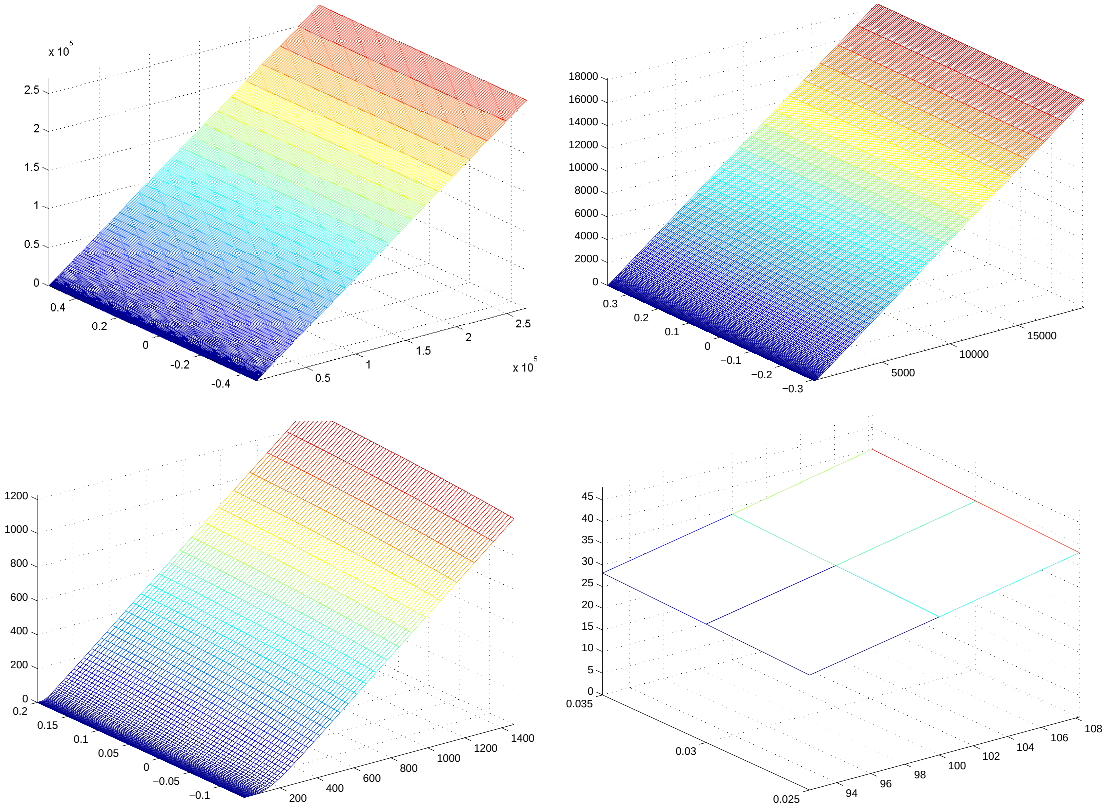

Figure 6.1: Value for a two-factor Hull-White call option using EulTree at $t = 5, 3\frac{1}{3}, \frac{5}{3}, 0$.

N	Q	s	RKCTreeImpr linear
140	25	13	40.5549
240	100	15	40.5057
300	100	19	40.6464

Table 6.2: RKCTreeImpr on a uniform grid.

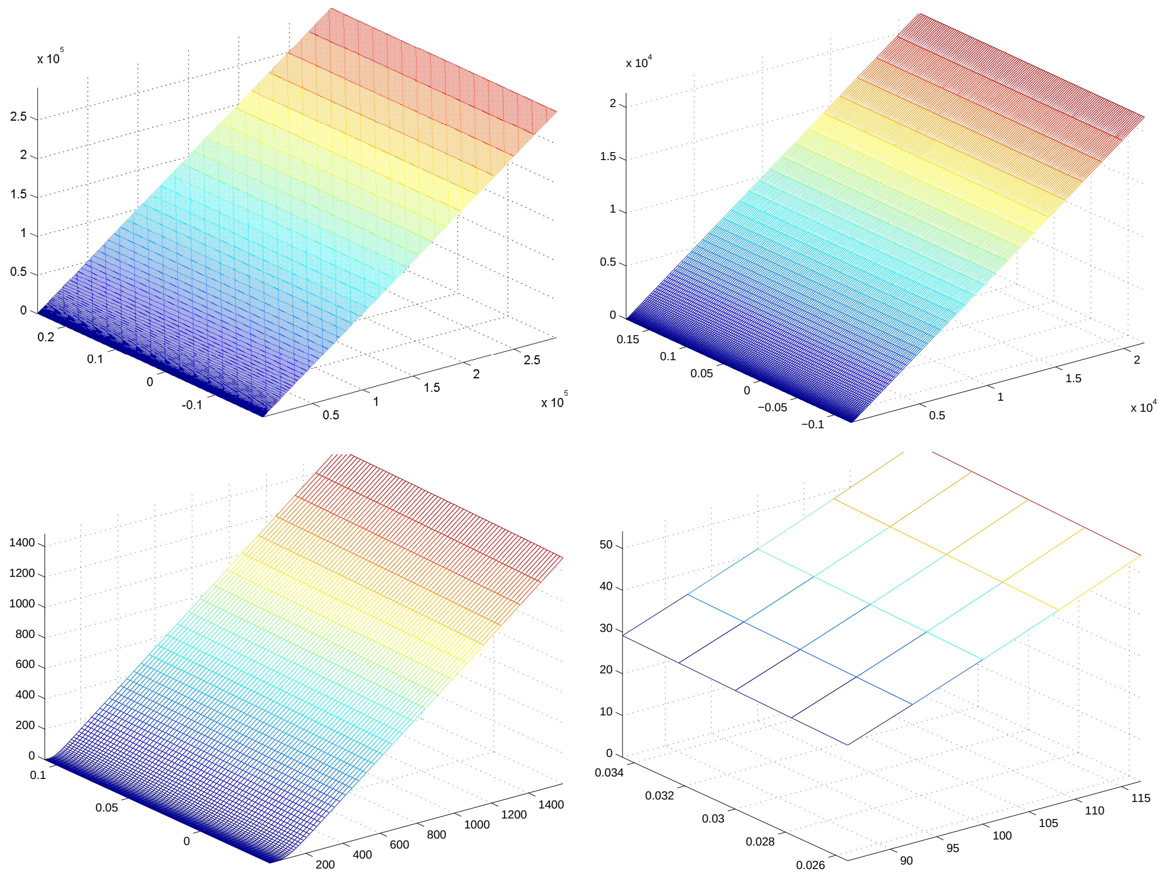

Figure 6.2: Value for a two-factor Hull-White call option using RKCTreeImpr at $t = 5, 3\frac{1}{3}, \frac{5}{3}, 0$.

6.2 Numerical results for a European call option conditional on the interest rate

To show the performance of the RKCTreeImpr we will take a look at a hybrid option. It consists of a call in the S-direction together with a digital in the r-direction. The payoff of this product is given by

$$f(r,S) = \max(S - K_S, 0)\,\mathbf{1}_{r<K_r}, \quad S \in (0,\infty), \quad r \in (-\infty,\infty), \tag{6.3}$$

where K_S and K_r denote the strikes for respectively the call and the digital.

As an example the following parameters are chosen

$$\begin{aligned} T &= 1, \quad q = 0, \quad S_0 = 100, \quad \sigma = 0.25, \quad r_0 = 0.03, \quad a = 0.05, \\ \theta(t) &= \theta = 0.025, \quad K_r = 0.05, \quad \sigma_r = 0.01, \quad \rho = 0.5. \end{aligned} \tag{6.4}$$

and the solution is shown in Figure 6.3. It might be expected that problems arise due to the digital in the r-direction, but the method handles the digital quite well.

6.3 Conclusion

The two-factor Hull–White model can be solved with EulTree,RKCTree and RKCTreeImpr. However, the first two methods can only be applied if a special grid is chosen. Eigenvalue analysis shows that if the correlation term is taken too large, positive eigenvalues occur. As a result inaccurate answers occur with large maturities.

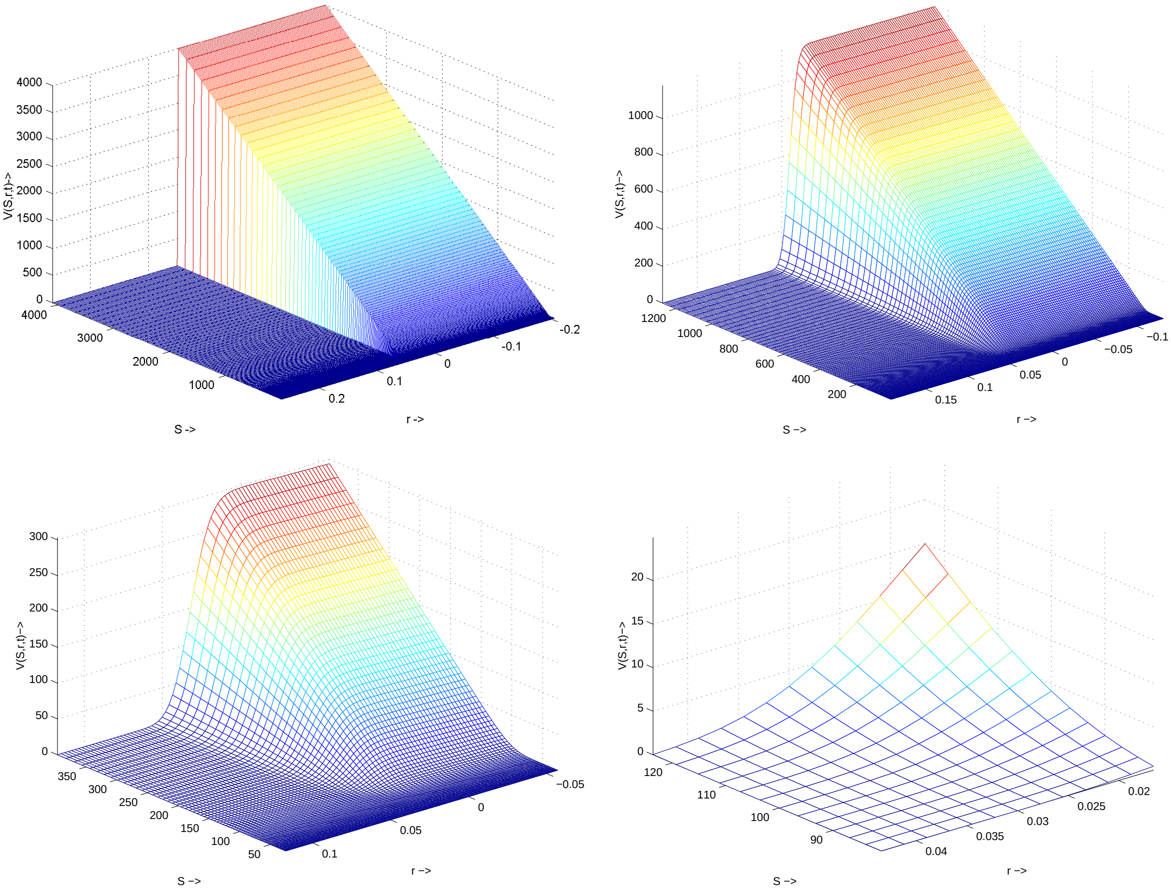

Figure 6.3: Value for a two-factor Hull-White digital-call option using RKCTreeImpr at $t = 1, 2/3, 1/3, 0$. The solution is 2.14759.

Chapter 7

Heston model

In this chapter the results for the two-factor Heston model (H) will be discussed. For a European call option the (quasi) closed formula reads

$$C(x,v,\tau) = Ke^{x}P_1(x,v,\tau) - P_0(x,v,\tau), \tag{7.1}$$

with

$$\begin{aligned}
x &= \log(\frac{F_{t,T}}{K}),\\
F_{t,T} &= e^{-r(T-t)}S_t,\\
P_j(x,v,\tau) &= \frac{1}{2} + \frac{1}{\pi}\int_0^{\infty} Re(\frac{e^{C_j(u,\tau)\overline{v}+D_j(u,\tau)v+iux}}{iu})d\phi,\\
C_j(u,\tau) &= \lambda r_{-}\tau - \frac{2}{\eta^2}\log(\frac{1-ge^{-d\tau}}{1-g}),\\
D_j(u,\tau) &= r_{-}\frac{1-e^{-d\tau}}{1-ge^{-d\tau}},\\
r_{\pm} &= \frac{\beta \pm d}{\eta^2}, d = \sqrt{\beta^2 - 4\alpha\gamma},\\
\alpha &= \frac{-u^2}{2} - \frac{iu}{2} + iju, \beta = \lambda - \rho\eta j - \rho\eta iu, \gamma = \frac{\eta^2}{2}, g = \frac{r_-}{r_+}.
\end{aligned}$$

This option will be used as a benchmark for our numerical experiments. In this chapter we only focus on RKCTreeImpr.

7.1 Numerial results for a European call option

For the Heston model we use the following grid definition in the v-direction:

$$\begin{aligned}
v_i &= v_0 - (\frac{N}{2} + 1 - i)\Delta v_L, \quad i = 1, \cdots, N/2,\\
v_{\frac{N}{2}+1} &= v_0,\\
v_{\frac{N}{2}+1+i} &= v_0 + i\Delta v_R, \quad i = 1, \cdots, N/2.
\end{aligned}$$

The variance can never be negative and therefore $v_{min} := v_0 - \frac{N}{2}\Delta v_L > 0$. Thus, for given N, the grid on the left of v_0 differs from the grid on the right of v_0. For the S-direction we will use an exponential grid and the first derivative in the S-direction will be approximated using central differences. The first derivative in the v-direction is discretized using the upwind scheme. The second derivative in both directions is again approximated using a three-point stencil. The crossterm is discretized via a nine-point stencil as discussed in Chapter 2.

Consider a European call option with parameters

$$T = 1, \quad r = 0.1, \quad S_0 = 100, \quad \rho = 0.1, \quad \eta = 0.9, \quad \lambda = 5, \quad q = 0, \quad v_0 = 0.0625.$$

In Table 7.1 the results for the RKCTreeImpr are shown and it is seen that all, except for the cubic extrapolation, converge toward the exact solution. This failure of the cubic extrapolation is caused by the fact that this extrapolation is not a good one at the boundaries. Furthermore it is shown in Figure 7.1 that the solutions are stable.

N	Q	s	FullRKC	RKCTreeImpr		
				linear	quadratic	cubic
50	25	35	18.9223	18.7664	18.9274	18.9090
100	50	47	19.0701	19.0322	19.0714	19.0694
200	100	65	19.1188	19.1147	19.1194	19.1195
400	200	90	19.1461	19.1502	19.1428	19.4340
800	400	126	19.1600	19.1666	19.1537	13.4798

Table 7.1: Four approaches compared. The exact solution to the problem is 19.1652.

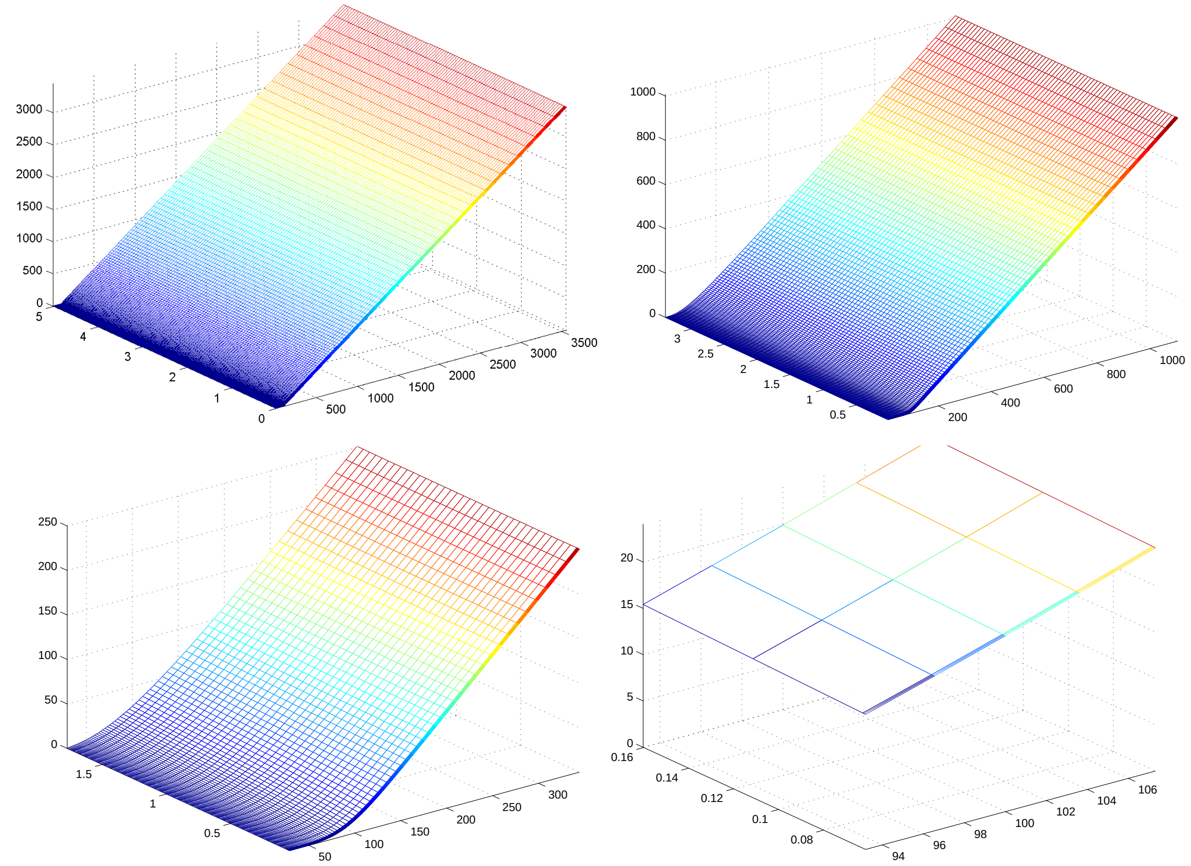

Figure 7.1: Value of a two-factor Heston call option using RKCTreeImpr at $t = 5, 3\frac{1}{3}, \frac{5}{3}, 0$.

7.2 Conclusion

The semi-discretization matrix for the Heston problem is generally very stiff. As a consequence only RKCTreeImpr with linear or quadratic extrapolation can be applied. Better accuracy is reached when a uniform or refined grid would be used in the S-direction. However, these grids will increase the total stiffness of the problem even more. To keep the method stable one must then take a lot of extra stages and this will increase the computational time drastically. In addition, due to these extra stages more extrapolations are needed in each time step and this may result in a less accurate solution.

Part III

Three-factor models

Chapter 8

Black–Scholes and Heston—Hull–White model

In this chapter we will shortly discuss the three-factor Black–Scholes and Heston—Hull–White model. Both models are extensions of the already known and investigated two-factor models. It was shown that only RKCTreeImpr could be applied to all models and therefore we will focus only on this method.

The three-factor Black–Scholes model is given by

$$\begin{cases} \frac{\partial V}{\partial t} + \frac{1}{2}\sigma_1^2 S_1^2 \frac{\partial^2 V}{\partial S_1^2} + + \frac{1}{2}\sigma_2^2 S_2^2 \frac{\partial^2 V}{\partial S_2^2} + \frac{1}{2}\sigma_3^2 S_3^2 \frac{\partial^2 V}{\partial S_3^2} + (r_1 - q_1) S_1 \frac{\partial V}{\partial S_1} + (r_2 - q_2) S_2 \frac{\partial V}{\partial S_2} \\ \qquad + (r_3 - q_3) S_3 \frac{\partial V}{\partial S_3} + \sigma_1 \sigma_2 S_1 S_2 \rho_{12} \frac{\partial^2 V}{\partial S_1 S_2} + \sigma_1 \sigma_3 S_1 S_3 \rho_{13} \frac{\partial^2 V}{\partial S_1 S_3} \\ \qquad + \sigma_2 \sigma_3 S_1 S_2 \rho_{23} \frac{\partial^2 V}{\partial S_2 S_3} - rV = 0, \qquad S_1,\ S_2,\ S_3 \in (0, \infty), \quad t \in (0, T), \\ V(S_1, S_2, S_3, T) = f(S_1, S_2, S_3), \quad S_1,\ S_2,\ S_3 \in (0, \infty). \end{cases}$$

As an example for the three-factor Black–Scholes model we will use a basket option. The payoff for a basket option is given by $f(S_1, S_2, S_3, t) = \max(\sum_{i=1}^{3} \alpha_i S_i, 0)$. Since no closed formula exists for this option, there is no benchmark.

As an example for the three-factor Heston—Hull–White model (H—H–W) we will use a European call option, i.e. $f(S_1, S_2, S_3, t) = \max(S_1 - K, 0)$, and we will also take a look at a call option conditional on the interest: $f(S, r, v) = \max(S - K_S, 0)\, \mathbf{1}_{r<K_r}$.

8.1 Numerical results for a basket option

The following parameters are chosen for the basket option

$$\begin{aligned} \alpha_1 &= 1, \sigma_1 = 0.25, \quad q_1 = 0.07, \quad S_{01} = 105, \\ \alpha_2 &= 2, \sigma_2 = 0.2, \quad q_2 = 0.05, \quad S_{02} = 50, \\ \alpha_3 &= 2, \sigma_3 = 0.2, \quad q_3 = 0.05, \quad S_{03} = 50, \\ \rho_{12} &= 0.5, \quad \rho_{12} = 0.5, \quad \rho_{12} = 0.5, \quad r = 0.1, \quad T = 1, \quad K = 100. \end{aligned}$$

The results for this option are given in Table 8.1 and Figure 8.1. This figure is a slice through $S_3 = 50$ in the S_1- and S_2- direction.

N	Q	s	RKCTreeImpr linear
26	10	35	197.664
46	20	47	197.664
66	30	65	197.664
86	40	65	197.664

Table 8.1: Results for the basket option in a three-factor Black–Scholes model.

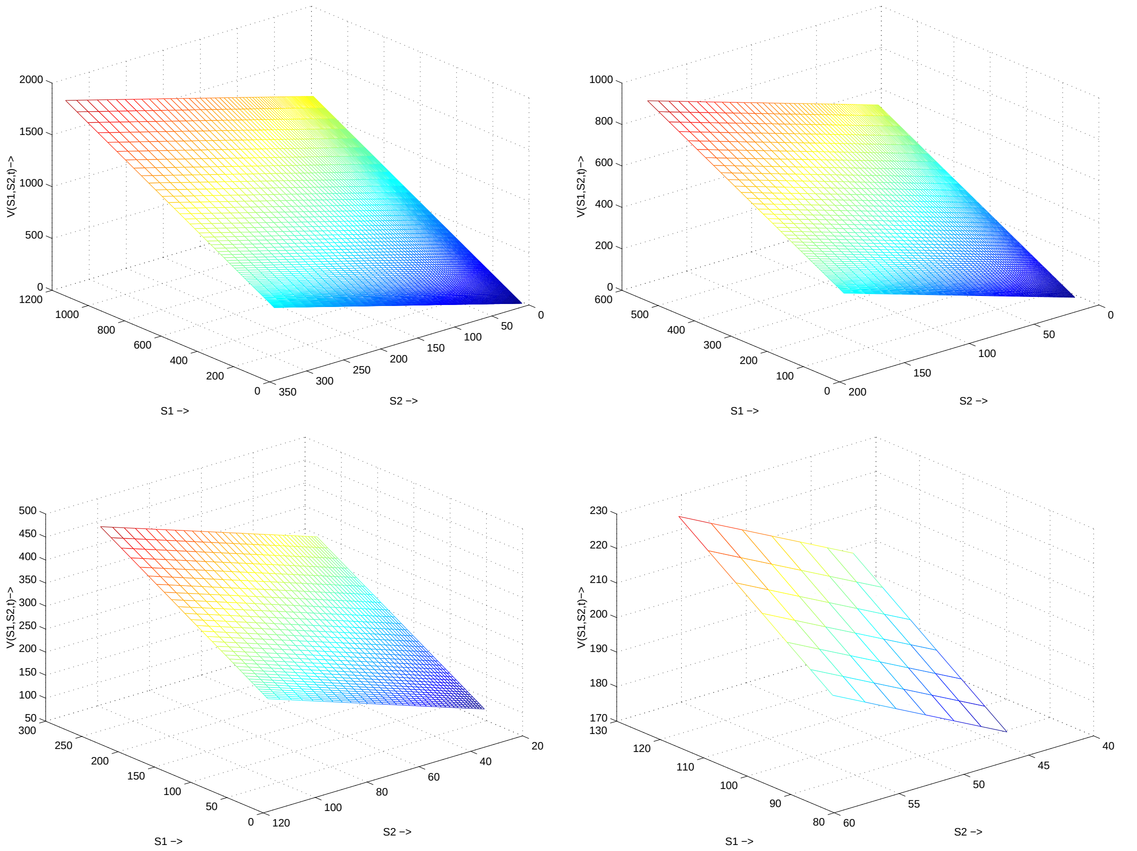

Figure 8.1: Value of a basket option in a three-factor Black–Scholes model using RKCTreeImpr at $t = 1, 23, \frac{1}{3}, 0$.

8.2 Numerical results for a European call and a digital-call

The following parameters are chosen for the European call option

$$\begin{aligned} S_0 &= 100, \quad q = 0, \\ r_0 &= 0.03, \quad a = 0.05, \quad \theta(t) = \theta = 0.025, \quad \sigma_r = 0.01, \\ \eta &= 0.9, \quad \lambda = 5, \quad v_0 = 0.0625, \\ \rho_{12} &= 0.1, \quad \rho_{13} = 0.1, \quad \rho_{23} = 0.1, \quad T = 1, \quad K = 100. \end{aligned} \tag{8.1}$$

and results are given in Table 8.2 and Figure 8.2. For the call option conditional on the interest the same parameters are taken and the strike for the interest is $K_r = 0.05$. Results are shown in Table 8.3 and Figure 8.3. This figure is a slice through v_0 in the S- and r- direction.

It is shown that the numerical results for the call option and the digital call option are the same. This is due to the fact that K_r is above r_0. There is, however, still some work to be done to make the method better and to ensure stability in all cases. It is very important that the correlation terms are controlled to obtain a stable method. Furthermore it is shown that the stable results are obtained.

N	Q	s	RKCTreeImpr linear
30	10	25	15.8962
50	20	30	16.1693
110	50	45	16.3881

Table 8.2: Results for a European call option in a three-factor Heston—Hull–White model.

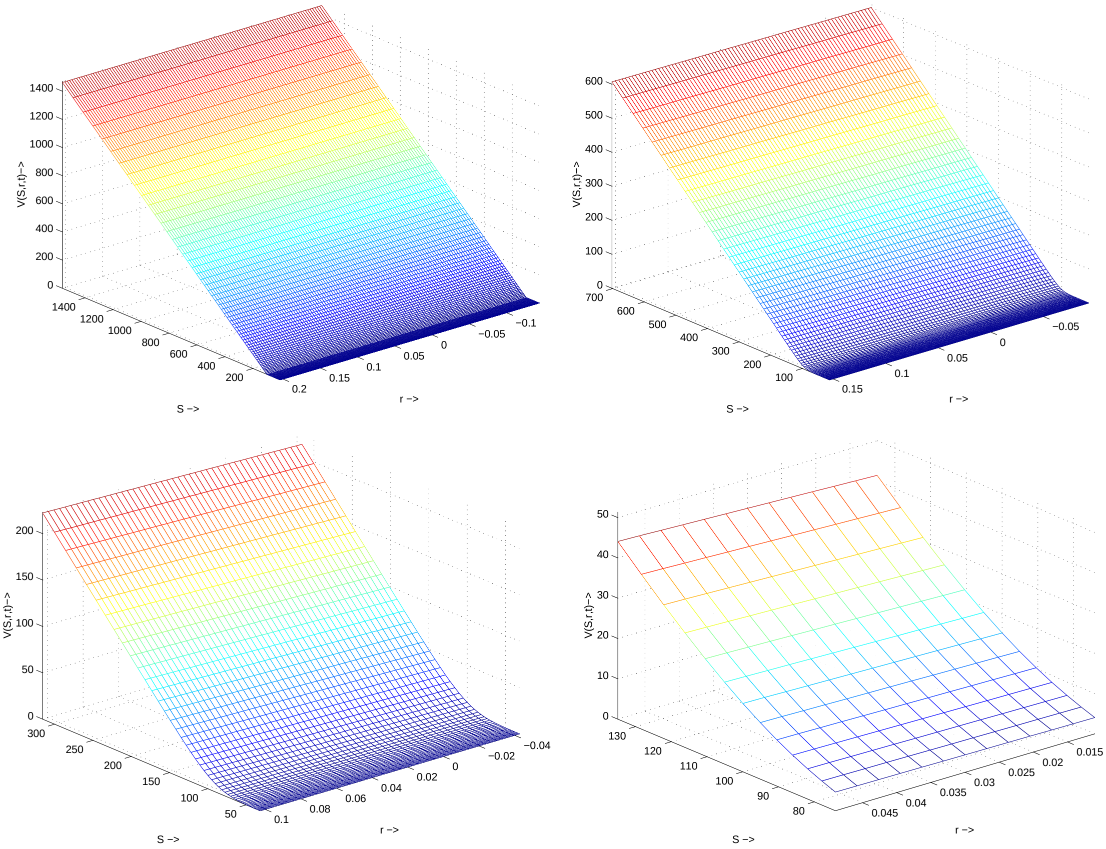

Figure 8.2: Value of a call option in a three-factor Heston—Hull–White model using RKCTreeImpr at $t = 1, \frac{2}{3}, \frac{1}{3}, 0$.

N	Q	s	RKCTreeImpr linear
30	10	25	5.46226
110	50	45	6.23791

Table 8.3: Results for a digital-call option in a three-factor Heston—Hull–White model.

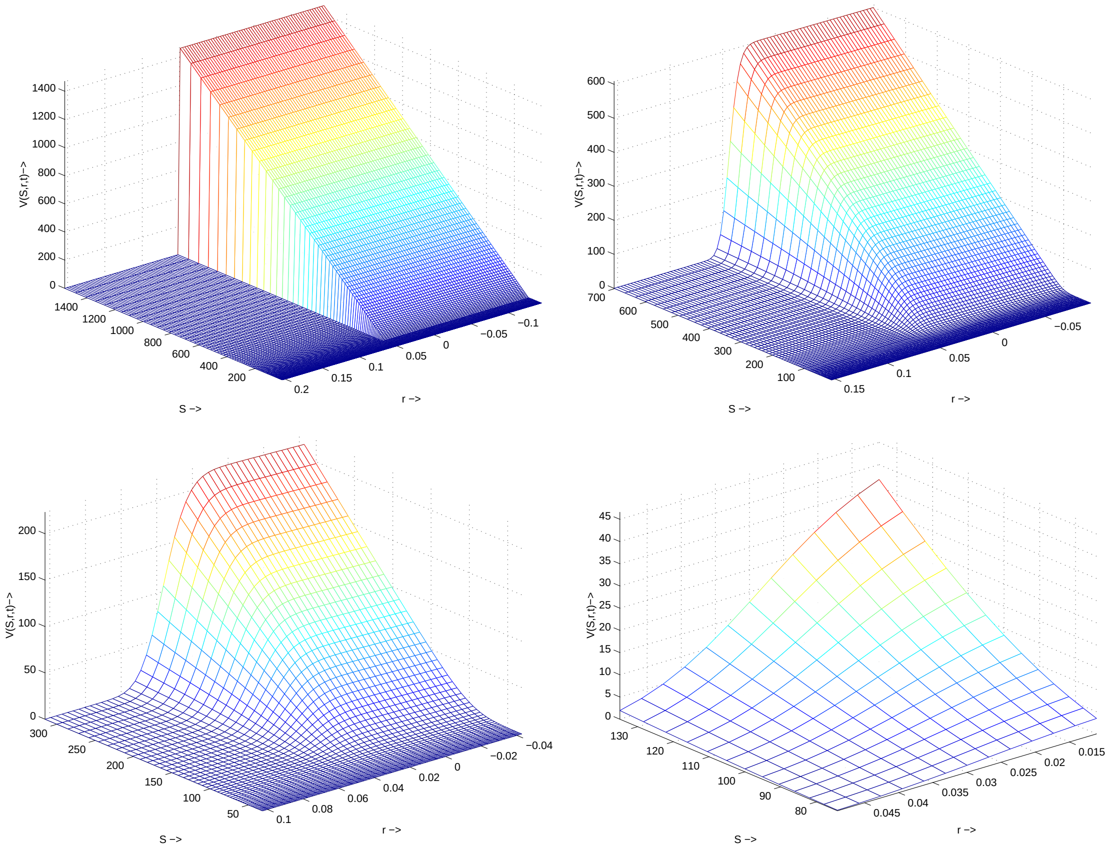

Figure 8.3: Value for a digital call option in a three-factor Heston—Hull–White model using RKC-TreeImpr at $t = 1, \frac{2}{3}, \frac{1}{3}, 0$.

Chapter 9

Conclusion & Recommendations

In this thesis we focused on the implementation of a fast and accurate multi-dimensional finite difference solver for solving initial value problems, as these are the most common problems in finance. The most important issues that needed to be solved were the "unknown boundary conditions" and the stiffness of the discretization matrix of the problem. To solve the first problem two approaches were suggested. The first approach was based on an implicit time integration scheme and the second was based on an explicit time integration scheme. It was seen that the first approach leads to an unstable method. Therefore we focused on the explicit approach only. To solve the initial value problems three explicit methods were proposed: EulTree (based on the explicit Euler forward method), RKCTree and RKCTreeImpr (both based on the RKC method). It was shown that the first two methods could solve the one- and two-dimensional Black–Scholes and Hull–White model. However, this could only be done if a special grid was chosen. So these methods are not very suited, since they could not be applied to general initial value problems and general grids. The third proposed method could, however, solve all (stiff) initial value problems on any desired grid. Furthermore, since boundary points that are of no influence to the final solution are stripped off, the computational costs can be reduced and because this method can be applied to any desired grid accurate solutions can be obtained.

Therefore it can be concluded that we have built an accurate multi-dimensional solver, which can solve initial value problems as they arise in finance.

Although some good results were obtained with RKCTreeImpr, the method can be improved. One of the improvements could be to extrapolate in time instead of the spatial extrapolation, since the solutions might be smoother in time.

A number of different grids were applied, such as an exponentially stretched grid, uniform grid and a refined grid. For convergence it is best to have a grid refinement around the nodes where problems are to be expected, e.g. around the nodes where the solution is discontinuous. However, for computational costs it is better to choose an exponentially stretched grid, since the spectral radius of the discretization matrix is then smaller. It is worth investigating what grid is best to use, or what combination to take to get a better convergence.

The proposed setup allows us to price all kinds of options, such as convertible bonds and CMSs. It furthermore allows us to price strongly path dependent options and it is worth applying the proposed method to solve these problems.

We have investigated the proposed method to solve up to a three-factor model. An extension to four- and five-dimensional models should be made.

Since the method is explicit it is well suited for parallel programming. Computational costs can then drastically decrease.

Traders are not always interested in the price of the option, but in the risk of having an option. It is therefore very important that the Greeks (derivatives of the solution) can be calculated. This can also be done in this model. For example if the $\Delta = \frac{\partial V}{\partial S}$ has to be calculated, the number of grid nodes N is set equal to or greater than $2Q + 2$. The remaining interval is now large enough to approximate the first derivative using numerical methods.

A point of concern regarding this method is/are the correlation term(s). These can be better handled if instead of the nine-point stencil the stencil as described in [13] is used.

Appendix A

Stability for Euler forward

Fix Δt and choose

$$\begin{aligned} S_i &= e^{x_i},\ i = 1, .., N+1, \\ \Delta S_i &= S_{i+1} - S_i = e^{x_{min}+i\Delta x} - e^{x_{min}+(i-1)\Delta x} = e^{x_{min}+(i-1)\Delta x}(e^{\Delta x} - 1). \end{aligned} \tag{A.1}$$

where

$$\begin{aligned} \Delta x &= \log(y), \\ y &= -\frac{1}{2}a + \sqrt{\frac{1}{4}a^2 - 1}, \\ a &= -2 + \frac{\Delta t \sigma^2 n}{1 - \Delta t(r-q)},\ n \geq 1. \end{aligned}$$

This approach also guarantees a stable method, which we will proof.

Substitution of (3.9) into (3.5) and using Theorem 2.1 gives for the stability condition for p_i

$$\begin{aligned} & 1 - \Delta t(r-q) + \Delta t \frac{\sigma^2}{(e^{\Delta x} - 1)^2 e^{-\Delta x}} \geq 0 \\ \rightarrow \quad & \Delta t \sigma^2 \geq (e^{\Delta x} - 1)^2 e^{-\Delta x}(-1 + \Delta t(r-q)) \end{aligned} \tag{A.2}$$

Rewriting this equation gives the first statement in Lemma 3.1.
If $-1 + \Delta t(r-q) < 0$, which is especially the case if $r - q > 0$, and holds for $\Delta t \rightarrow 0$, this equation can be rewritten to

$$(e^{\Delta x} - 1)^2 e^{-\Delta x} \geq \frac{\Delta t \sigma^2}{-1 + \Delta t(r-q)}$$

This holds if

$$(e^{\Delta x} - 1)^2 e^{-\Delta x} = \frac{n \Delta t \sigma^2}{-1 + \Delta t(r-q)} \tag{A.3}$$

where $n \geq 1$.

Solving (A.3) gives the stepsize for the x-grid for which the Euler Explicit Forward method will still be stable.

$$\begin{aligned} (e^{\Delta x} - 1)^2 e^{-\Delta x} &= \frac{n \Delta t \sigma^2}{-1 + \Delta t(r-q)} \\ e^{\Delta x} - 2 + e^{-\Delta x} &= \frac{n \Delta t \sigma^2}{-1 + \Delta t(r-q)} \\ e^{\Delta x} - 2 - \frac{n \Delta t \sigma^2}{-1 + \Delta t(r-q)} + e^{-\Delta x} &= 0 \\ e^{2\Delta x} + (-2 - \frac{n \Delta t \sigma^2}{-1 + \Delta t(r-q)}) e^{\Delta x} + 1 &= 0 \end{aligned} \tag{A.4}$$

This equation can easily be solved yielding:

$$\begin{aligned}
&S_i = e^{x_{min}+(i-1)\Delta x},\ i = 1,..,N+1\\
&\Delta x = \log(y),\\
&y = -\frac{1}{2}a + \sqrt{\frac{1}{4}a^2 - 1},\\
&a = -2 + \frac{\Delta t \sigma^2 n}{1-\Delta t(r-q)},\ n \geq 1.
\end{aligned}$$

which proofs the second statement in Lemma 3.1.

This equation only makes sense if $\frac{1}{4}a^2 - 1 \geq 0$ and if $y > 0$.

$$\begin{aligned}
&\frac{1}{4}a^2 - 1 \geq 0,\\
&\frac{1}{4}(-2 + \frac{\Delta t \sigma^2 n}{1-\Delta t(r-q)})^2 - 1 \geq 0,\\
&1 - \frac{\Delta t \sigma^2 n}{1-\Delta t(r-q)} + \frac{1}{4}(\frac{\Delta t \sigma^2 n}{1-\Delta t(r-q)})^2 - 1 \geq 0,\\
&\underbrace{\frac{\Delta t \sigma^2 n}{1-\Delta t(r-q)}}_{part1}\underbrace{(-1 + \frac{1}{4}\frac{\Delta t \sigma^2 n}{1-\Delta t(r-q)})}_{part2} \geq 0.
\end{aligned}$$

Part 1 and part 2 are both smaller than zero and therefore the inequality holds. The second restriction is the positiveness for y, which is always satisfied if $a \leq 0$.

$$\begin{aligned}
&a \leq 0,\\
&-2 + \frac{\Delta t \sigma^2 n}{1-\Delta t(r-q)} \leq 0,\\
&\Delta t \leq \frac{2}{\sigma^2 n + 2(r-q)}.
\end{aligned} \tag{A.5}$$

The last equation allows large values for Δt, so this criterion will always be satisfied. The same thing can be done for the upper and lower diagonal elements. Applying the transformation and requiring that the diagonal elements will always be above zero yields the following constraint to keep a_{i-1} and a_{i+1} positive, which leads to the last statement in Lemma 3.1

$$\log(\frac{r-q}{\sigma^2 + (r-q)}) \leq \Delta x \leq \log(\frac{\sigma^2 + r - q}{r-q}).$$

Appendix B

Exact formula zero coupon bond

In this section an exact solution for a simple zero coupon bond under the Hull-White interest rate model is derived. The exact solution will be used as a reference for the numerical results and later on this exact solution will be used for the transformation to scale out the '$-rV$' term.

A simple zero coupon bond is a contract for which the holder pays a certain premium at the beginning of the period and receives 1 at the end of that period. The pricing equation for the zero coupon bond under the Hull-White interest rate model is given by (4.1).

We assume that the solution of (4.1) can be written in the form

$$V(r,t) = e^{A(t)+rB(t)}, \tag{B.1}$$

with final condition

$$V(r,T) = 1.$$

Substitution of (B.1) in (4.1) gives

$$\begin{aligned}&(A'(t) + rB'(t))V(r,t) + (\Theta(t) - a(t)r)B(t)V(r,t),\\&+\frac{1}{2}\sigma_r^2(t)B^2(t)V(r,t) - rV(r,t) = 0.\end{aligned} \tag{B.2}$$

Rearranging terms gives

$$V(r,t)([A'(t) + \Theta(t)B(t) + \frac{1}{2}\sigma_r^2(t)B^2(t)] + r[B'(t) - aB(t) - 1]) = 0,$$

and since this must hold for any $V(r,t)$ it follows that

$$[A'(t) + \Theta(t)B(t) + \frac{1}{2}\sigma_r^2(t)B^2(t)] + r[B'(t) - aB(t) - 1] = 0.$$

This equation is valid for all r and so both terms between brackets have to be zero, which leads to the system with two unknowns $A(t)$ and $B(t)$

$$A'(t) + \Theta(t)B(t) + \frac{1}{2}\sigma_r^2(t)B^2(t) = 0, \tag{B.3}$$

$$B'(t) - a(t)B(t) - 1 = 0. \tag{B.4}$$

The final condition for $V(r,T)$ is given by (B.2). Since neither $A(t)$ nor $B(t)$ are functions of r it follows that $A(t)$ and $B(t)$ are both zero on T.

$$A(T) = 0, \tag{B.5}$$

$$B(T) = 0. \tag{B.6}$$

Using an appropriate integrating factor, the solution of (B.4) is given by

$$B(t) = \frac{-\int_t^T e^{\int_u^T a(s)ds}du}{e^{\int_t^T a(s)ds}}. \tag{B.7}$$

Equation (B.3) can be solved by using $-\int_t^T A'(t)dt = A(t) - A(T) = A(t)$ (the last equivalence holds due to the endcondition (B.5): $A(T) = 0$). So $A(t)$ can be computed by rearranging (B.3) to $A'(t) = -\Theta(t)B(t) - \frac{1}{2}\sigma_r^2 B^2(t)$ and integrating from T to t.

$$A(t) = -\int_t^T [-\Theta(s)B(s) - \frac{1}{2}\sigma_r^2(s)B^2(s)]ds. \tag{B.8}$$

$$\tag{B.9}$$

The exact solution of the zero coupon bond under the Hull-White model (4.1) is given by

$$\begin{aligned} V(r,t) &= e^{A(t)+rB(t)}, && \text{(B.10)} \\ B(t) &= \frac{-\int_t^T e^{\int_u^T a(s)ds} du}{e^{\int_t^T a(s)ds}}, && \text{(B.11)} \\ A(t) &= -\int_t^T [-\Theta(s)B(s) - \frac{1}{2}\sigma_r^2(s)B^2(s)]ds. && \text{(B.12)} \end{aligned}$$

Appendix C

Derivation of exact solution for an exchange option

An exchange option is a right (not an obligation) to exchange a quantity q_1 of an underlier S_1 for a quantity q_2 of an underlier S_2 on $t = T$. The payoff of an exchange option is given by

$$f(S_1, S_2) = \max(\alpha_1 S_1 - \alpha_2 S_2, 0). \tag{C.1}$$

Following [21], we derive in this appendix an exact solution of problem (5.1) together with (C.1).

The following transformation is applied:

$$V(S_1, S_2, t) = \alpha_1 S_2 H(\xi, t), \tag{C.2}$$

with

$$\xi = \frac{S_1}{S_2}.$$

Substituting (C.2) into (5.1) reduces the dimension of the equation, making it easier to solve. The partial derivatives in the new coordinate become:

$$\begin{aligned}
\frac{\partial V}{\partial t} &= \alpha_1 S_2 \frac{\partial H}{\partial t} \\
\frac{\partial V}{\partial S_1} &= \frac{\partial \alpha_1 S_2 H}{\partial S_1} = \frac{\partial \alpha_1 S_2 H}{\partial \xi}\frac{\partial \xi}{\partial S_1} = \alpha_1 S_2 \frac{\partial H}{\partial \xi}\frac{1}{S_2} = \alpha_1 \frac{\partial H}{\partial \xi}, \\
\frac{\partial V}{\partial S_2} &= \frac{\alpha_1 S_2 \partial H}{\partial S_2} = \alpha_1 H + \alpha_1 S_2 \frac{\partial H}{\partial \xi}\frac{\xi}{\partial S_2} = \alpha_1 H + \alpha_1 S_2 \frac{\partial H}{\partial \xi}\frac{-S_1}{S_2^2} = \alpha_1 H - \alpha_1 \xi \frac{\partial H}{\partial \xi}, \\
\frac{\partial^2 V}{\partial S_1^2} &= \frac{\partial(\alpha_1 \frac{\partial H}{\partial \xi})}{\partial \xi}\frac{\partial \xi}{\partial S_1} = \alpha_1 \frac{\partial^2 H}{\partial \xi^2}\frac{1}{S_2}, \\
\frac{\partial^2 V}{\partial S_2^2} &= \frac{\partial(\alpha_1 H - \alpha_1 \xi \frac{\partial H}{\partial \xi})}{\partial \xi}\frac{\partial \xi}{\partial S_2} = \alpha_1 \xi^2 \frac{\partial^2 H}{\partial \xi^2}\frac{1}{s_1^2}, \\
\frac{\partial^2 V}{\partial S_1 \partial S_2} &= \frac{\partial}{\partial S_2}(\alpha_1 \frac{\partial H}{\partial \xi}) = \alpha_1 \frac{\frac{\partial H}{\partial \xi}}{\partial \xi}\frac{\partial \xi}{\partial S_2} = -\alpha_1 \xi \frac{\partial^2 H}{\partial \xi^2}\frac{1}{S_2}.
\end{aligned}$$

Substituting this into (5.1) gives

$$\begin{aligned}
&\alpha_1 S_2 \frac{\partial H}{\partial t} + \frac{1}{2}\sigma_1^2 S_1^2 \alpha_1 \frac{\partial^2 H}{\partial \xi^2}\frac{1}{S_2} - \sigma_1 \sigma_2 S_1 S_2 \rho_{12} \alpha_1 \xi \frac{\partial^2 H}{\partial \xi^2}\frac{1}{S_2} + \frac{1}{2}\sigma_2^2 S_2^2 \alpha_1 \xi^2 \frac{\partial^2 H}{\partial \xi^2}\frac{1}{S_2} \\
&+(r - q_1) S_1 \alpha_1 \frac{\partial H}{\partial \xi} + (r - q_2) S_2 (\alpha_1 H - \alpha_1 \xi \frac{\partial H}{\partial \xi}) - r \alpha_1 S_2 H = 0.
\end{aligned} \tag{C.3}$$

Divide by $\alpha_1 S_2$ to obtain

$$\begin{aligned}
&\frac{\partial H}{\partial t} + \frac{1}{2}\sigma_1^2 \xi^2 \frac{\partial^2 H}{\partial \xi^2} - \sigma_1 \sigma_2 \rho_{12} \xi^2 \frac{\partial^2 H}{\partial \xi^2} + \frac{1}{2}\sigma_2^2 \xi^2 \frac{\partial^2 H}{\partial \xi^2} \\
&+(r - q_1)\xi \frac{\partial H}{\partial \xi} + (r - q_2)(H - \xi \frac{\partial H}{\partial \xi}) - rH = 0.
\end{aligned} \tag{C.4}$$

Reordering terms gives:

$$\frac{\partial H}{\partial t} + \xi^2(\frac{1}{2}\sigma_1^2 - \sigma_1\sigma_2\rho_{12} + \frac{1}{2}\sigma_2^2)\frac{\partial^2 H}{\partial \xi^2} + \xi(q_2 - q_1)\frac{\partial H}{\partial \xi} - q_2 H = 0. \tag{C.5}$$

Writing $\sigma' = \sqrt{\frac{1}{2}\sigma_1^2 - \sigma_1\sigma_2\rho_{12} + \frac{1}{2}\sigma_2^2}$ yields the following one-dimensional equation

$$\frac{\partial H}{\partial t} + \xi^2\sigma'\frac{\partial^2 H}{\partial \xi^2} + \xi(q_2 - q_1)\frac{\partial H}{\partial \xi} - q_2 H = 0.$$

Transforming the payoff results in

$$H(\xi, T) = \max(\xi - \frac{\alpha_2}{\alpha_1}, 0). \tag{C.6}$$

Equation (C.6) with payoff (C.6) looks like a call option, where the interest r is now replaced by q_2 and the dividend q is replaced by q_1. The exact solution for this initial value problem will be similar to the exact solution of a call option. For a derivation of the exact solution of the Black–Scholes equation see [15]. So the solution to (C.6) with initial value (C.6) is

$$\xi e^{-(q_1)(T-t)}N(d1) - e^{-(q_2)(T-t)}N(d2),$$

with

$$d1 = \frac{ln(\frac{\xi}{\frac{\alpha_2}{\alpha_1}} + (q_2 - q_1 + \frac{\sigma'}{2}))}{\sigma'\sqrt{T-t}},$$
$$d2 = \frac{ln(\frac{\xi}{\frac{\alpha_2}{\alpha_1}} + (q_2 - q_1 - \frac{\sigma'}{2}))}{\sigma'\sqrt{T-t}}.$$

Transforming this equation back gives the exact solution for an exchange option

$$V(S_1, S_2, t) = \alpha_1 S_1 e^{-q_1(T-t)}N(d_1') - \alpha_2 S_2 e^{-q_2(T-t)}N(d_2')$$

where

$$\begin{aligned} d_1' &= \frac{ln(\frac{\alpha_1 S_1}{\alpha_2 S_2}) + (q_2 - q_1 + \frac{1}{2}\sigma'^2)(T-t)}{\sigma'\sqrt{T-t}}, \\ d_2' &= \frac{ln(\frac{\alpha_1 S_1}{\alpha_2 S_2}) + (q_2 - q_1 - \frac{1}{2}\sigma'^2)(T-t)}{\sigma'\sqrt{T-t}}. \\ \sigma' &= \sqrt{\sigma_1^2 - 2\rho_{12}\sigma_1\sigma_2 + \sigma_2^2}. \end{aligned} \tag{C.7}$$

In Figure (C.1) and (C.2) resp. the payoff and the solution on $t = 0$ are plotted. The parameters used are:

$q_1 = 1$, $D_1 = 0.07$, $\sigma_1 = 0.25$, $S_{01} = 105$

$q_2 = 2$, $D_2 = 0.05$, $\sigma_2 = 0.2$, $S_{02} = 50$

$\rho_{12} = 0.5$, $r = 0.1$, $t = 0$, $T = 5$.

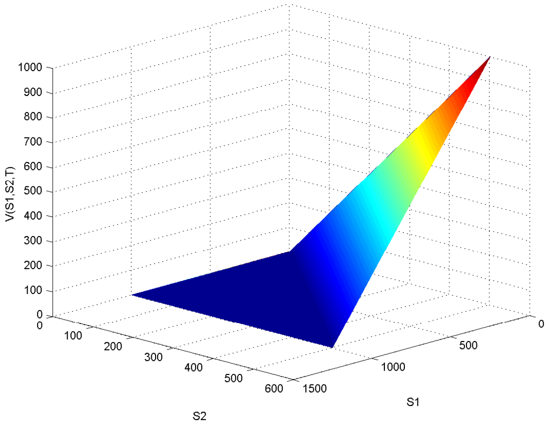

Figure C.1: payoff exchange option.

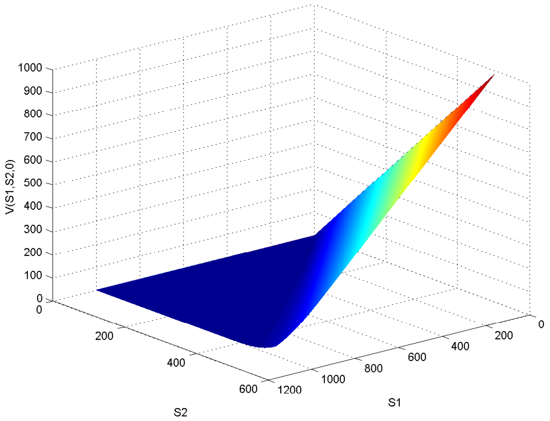

Figure C.2: value of exchange option at t = 0.

Bibliography

[1] F. Black, *From Black-Scholes to black holes*, 1992, Risk Magazine Ltd, Londen.

[2] http://www.newlearner.com/courses/hts/bec4a/ecoho16.

[3] J. Gatheral, *The Volatility Surface*, 2006, John Wiley & Sons, Inc., New Jersey.

[4] N Moodley, *The Heston Model: A Practical Approach*, 2005, Master's Thesis.

[5] S. Shreve, *Stochastic calculus and finance*, 1997, Lecture notes.

[6] D. Tavella and C. Randall: Pricing Financial Instruments, The Finite Difference method, 2000, John Wiley & Sons, Inc, New York.

[7] T. Björk, Arbitrage Theory in Continuous Time, 2001, Oxford University Press, USA.

[8] H. Windcliff, P.A. Forsyth, K.R. Vetzal, *Analysis of the stability of the linear boundary condition for the Black–Scholes equation*, 2004, J. Comp. Fin. **8**(4), 62–92.

[9] J.G. Verwer, B.P. Sommeijer, W. Hundsdorfer : *RKC time-stepping for advection-diffusion-reaction problems*, 2004, J. Comp. Phys., **201**, 61–79.

[10] E. Gaarder Haug, The Complete Guide To Option Pricing Formulas, 1998, McGraw-Hill, New York.

[11] J.C. Hull, Options, Futures and other derivatives, 1997, Prentice-Hall Inc.

[12] R.L. Burden, J. Douglas Faires, Numerical Analysis, 2001, Brooks/Cole, USA.

[13] P. Wesseling, Principles of Computational Fluid Dynamics, 2001, Springer-Verlag, Berlin.

[14] C. Leentvaar, *Numerical solution of the Black-Scholes equation with a small number of grid points*, 2003, Master's Thesis.

[15] P. Wilmott, J. Dewynne, S. Howison, Option Pricing, 1993, Oxford Financial Press, Oxford.

[16] S. Gerschgorin, *Über die Abgrenzung der Eigenwerte einer Matrix*, 1931, Isv. Akad. Nauk. USSR Ser. Mat., **7**, 749–754.

[17] D. Brigo and F. Mercurio, Interest Rate Models. Theory and Practice, 2001, Springer, Berlin.

[18] R. Kangro, R. Nicolaides, Far field boundary conditions for Black–Scholes equations, 2000, Siam J. Numer. Anal, **28**, 1357-1368.

[19] J. Hull and A. White, The pricing of options on assets with stochastic volatilities, 1987, J. Finance, **42**(2), 281–300.

[20] J.C.Cox, J.E. Ingersoll,Jr and S.A. Ross: A Theory of the Term Structure of Interest Rates, 1985,Econometrica, **53**(no. 2-March), 385–407.

[21] W. Margrabe, The Value of an Option to Exchange One Asset for Another, 1978, Journal of Finance, **33**, 177–86.