Finite Difference Methods in Derivatives Pricing under Stochastic Volatility Models.

Graaf, C.S.L. de

C.S.L. de Graaf

Finite Difference Methods in Derivatives Pricing under Stochastic Volatility Models.

Master thesis, defended on September 27, 2012

Thesis advisor: B. Koren

Specialisation: Applied Mathematics

Mathematisch Instituut,
Universiteit Leiden

Acknowledgements

This dissertation would not have been possible without several people who helped me in the preparation and completion of this study.

First, my gratitude goes to Barry Koren. Especially for introducing me to numerical methods and encouraging me to work at CWI. For me CWI, was a very pleasant environment. It has been very stimulating to work among others and the regular ping-pong matches formed a welcome distraction. This is why I would like to thank all the colleagues at CWI and in particular Kees Oosterlee for proposing me this subject, which fitted my expectations like a glove.

Furthermore I want to thank the whole DR&V team of Rabobank International and in particular Erik van Raaij for making me feel welcome, providing me a proper desk to do my research and carefully read my thesis updates. Also thanks to Tim Dijkstra for checking my codes and giving me good remarks. A special thanks goes to Natalia Borovykh for the excellent guidance and quick replies to all my emails.

Contents

1 Introduction

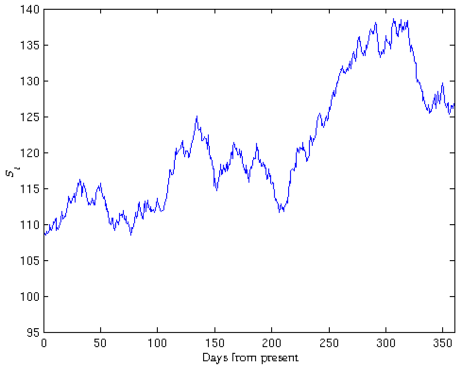
In option pricing, modelling the stock price behaviour is the key concept. In general, these processes are assumed to be stochastic. The well known Black Scholes model uses geometric Brownian motion to implement this feature:

$$dS_t = rS_t dt + \sigma S_t dW_t,$$

where S_0 denotes the currently observed price of the stock, S_t denotes the price of a stock at time t. Here W_t is a Wiener process. The price of this stock in the future is determined by two factors:

- A drift term r.
- And a stochastic term denoted by σ.

The drift term r can be seen as the risk free interest rate, while the stochastic term σ can be seen as the volatility of the stock. Figure 1.0a shows a generated price process of a single stock. The Black-Scholes model assumes the volatility of the stock to be constant. With this important assumption, closed form solutions of the values of European call and put options are derived. The volatility can be chosen in such a way that the model reproduces reasonable market prices. This volatility is called the implied volatility. A major drawback of this procedure is that this implied volatility depends on specifications in the contract of the options. This unrealistic feature can be resolved using so called local volatility models or stochastic volatility. Local volatility models assume the volatility to be a function of time and stock price, calibrated in such a way that the model is consistent with market data, see [11]. In an article by Hagan, Kumar, Lesniewski and Woodward [8] it is claimed that local volatility models do not correspond with the dynamic behaviour of market data. Some well known examples of stochastic volatility models are introduced by Heston [1], Hull and White [12] and Hagan et al [8]. These models assume that also the volatility V_t follows a stochastic process, and have the following general form:



1.0a B-S price process for a stock with $\sigma = 0.02$, $r = 0.03$ and $S_0 = 108.2$ over 360 days.

$$\begin{cases} dS_t &= \mu_S(S_t, V_t, t)dt + \sigma_S(S_t, V_t, t)dW_t^1, \\ dV_t &= \mu_V(S_t, V_t, t)dt + \sigma_V(S_t, V_t, t)dW_t^2, \\ dW_t^1 dW_t^2 &= \rho dt. \end{cases} \tag{1.1}$$

The last line denotes the correlation between the stock price S_t and its volatility V_t. The functions μ_S, μ_V, σ_S and σ_V are determined by the model used. In the models mentioned above there is more than one unknown parameter to be determined. Similar as the volatility in the Black Scholes case, these parameters can be calibrated using market data. To this end the SABR model is introduced. For some models, vanilla options can be priced in an analytic way. However, for more exotic options an analytic formula is not available and other methods need to be used to approximate the prices. To this end the Monte Carlo method is widely used. This method samples many paths of a stock price process using the processes defined in the model. By averaging over a large number of paths, a price is calculated. In the case of stochastic volatility models, this method needs to create two processes and can therefore become computationally expensive. Another method makes use of the fact that the price of an option in a stochastic volatility model can be represented by a two dimensional convection diffusion partial differential equation (PDE). In physics these PDEs are common and well studied. The finite difference method (FDM) is a proven numerical procedure to obtain accurate approximations to the relevant PDE. In this thesis this method is applied to the Heston model and the SABR model. The options being priced are vanilla call- and put options and the more exotic up-and out Barrier options.

2 Derivation of a PDE from a stochastic process

Using Feynman-Kac [13] one can derive that the price of an option or derivative is the solution of a PDE. The general stochastic volatility process (1.1) is used. Assuming $\mu_S = rS_t$, $\sigma_S = \sigma S_t$, $\mu_V = 0$ and $\sigma_V = 0$, the general model reduces to the Black Scholes model. It is assumed that the price of an option or a derivative U with underlying asset S with volatility V is a function of the price of this underlying asset, the volatility of this underlying asset and time, so in notation: $U = U(S, V, t)$. Using Itô's formula the stochastic differential process for this price is given by:

$$\begin{aligned} dU =& \frac{\partial U}{\partial t}dt + \frac{\partial U}{\partial S}dS_t + \frac{\partial U}{\partial V}dV_t \\ &+ \frac{1}{2}\frac{\partial^2 U}{\partial S^2}(dS_t)^2 + \frac{\partial^2 U}{\partial S \partial V}dS_t dV_t + \frac{1}{2}\frac{\partial^2 U}{\partial V^2}(dV_t)^2, \end{aligned} \tag{2.1}$$

with Box algebra:

$$\left[\begin{array}{cccc} & dt & dW_t^1 & dW_t^2 \\ dt & 0 & 0 & 0 \\ dW_t^1 & 0 & dt & \rho(t)dt \\ dW_t^2 & 0 & \rho(t)dt & dt \end{array}\right]. \tag{2.2}$$

Substituting equation (1.1) in (2.1) and using the box-algebra one derives

$$\begin{aligned} dU = & \left[\frac{1}{2}\sigma_V^2 \frac{\partial^2 U}{\partial V^2} + \sigma_S \sigma_V \rho \frac{\partial^2 U}{\partial S \partial V} + \frac{1}{2}\sigma_S^2 \frac{\partial^2 U}{\partial S^2} \right. \\ & \left. + \mu_V \frac{\partial U}{\partial V} + \mu_S \frac{\partial U}{\partial S} + \frac{\partial U}{\partial t}\right] dt \\ & + \sigma_S \frac{\partial U}{\partial S} dW_t^1 + \sigma_V \frac{\partial U}{\partial V} dW_t^2. \end{aligned} \tag{2.3}$$

For notational convenience, let $\mathcal{L}(\cdot)$ define the differential operator in the square brackets:

$$dU = \mathcal{L}(U)dt + \sigma_S \frac{\partial U}{\partial S} dW_t^1 + \sigma_V \frac{\partial U}{\partial V} dW_t^2. \tag{2.4}$$

The dynamics of a self-financing portfolio Π consisting of one derivative U, $-\Delta_1$ shares of the underlying stock S and $-\Delta_2$ units of another derivative U_2, will be:

$$\begin{aligned} d\Pi =& dU - \Delta_1 dS - \Delta_2 dU_2 \\ =& \left[\mathcal{L}(U)dt + \sigma_S \frac{\partial U}{\partial S} dW_t^1 + \sigma_V \frac{\partial U}{\partial V} dW_t^2\right] \\ &- \Delta_1 \left[\mu_S dt + \sigma_S dW_t^1\right] \\ &- \Delta_2 \left[\mathcal{L}(U_2)dt + \sigma_S \frac{\partial U_2}{\partial S} dW_t^1 + \sigma_V \frac{\partial U_2}{\partial V} dW_t^2\right]. \end{aligned} \tag{2.5}$$

Rearranging terms results in:

$$\begin{aligned} d\Pi =& \left[\mathcal{L}(U) - \Delta_1 \mu_S - \Delta_2 \mathcal{L}(U_2)\right] dt \\ &+ \left[\sigma_S \frac{\partial U}{\partial S} - \Delta_1 \sigma_S - \Delta_2 \sigma_S \frac{\partial U_2}{\partial S}\right] dW_t^1 \\ &+ \left[\sigma_V \frac{\partial U}{\partial V} - \Delta_2 \sigma_V \frac{\partial U_2}{\partial V}\right] dW_t^2. \end{aligned} \tag{2.6}$$

For Π to be a risk free self-financing portfolio, Δ_1 and Δ_2 must satisfy the following system:

$$\begin{aligned}
\sigma_S \frac{\partial U}{\partial S} - \Delta_1 \sigma_S - \Delta_2 \sigma_S \frac{\partial U_2}{\partial S} &= 0, \\
\sigma_V \frac{\partial U}{\partial V} - \Delta_2 \sigma_V \frac{\partial U_2}{\partial V} &= 0.
\end{aligned}$$

Solving this yields:

$$\Delta_2 = \left(\frac{\partial U}{\partial V} \right) / \left(\frac{\partial U_2}{\partial V} \right), \tag{2.7a}$$

$$\Delta_1 = \frac{\partial U}{\partial S} - \frac{\partial U_2}{\partial S} \left(\frac{\partial U}{\partial V} \right) / \left(\frac{\partial U_2}{\partial V} \right). \tag{2.7b}$$

To have absence of arbitrage the return of the constructed portfolio must be equal to the following risk-free investment:

$$d\Pi = r\Pi dt = r \left[U - \Delta_1 S - \Delta_2 U_2 \right] dt. \tag{2.8}$$

Expressions (2.6) and (2.8) can be equated, and substituting Δ_1 and Δ_2 results in:

$$\frac{\left[\mathcal{L}(U) - rU \right] dt}{\frac{\partial U}{\partial V}} = \frac{\left[\mathcal{L}(U_2) - rU_2 \right] dt}{\frac{\partial U_2}{\partial V}}.$$

Because this equation must hold for all derivatives U and U_2 independently, the right hand side is said to be equal to some function $\lambda(S, V, t)$. Choosing this function equal to zero, concludes this derivation:

$$\begin{aligned}
&\frac{1}{2}\sigma_V^2 \frac{\partial^2 U}{\partial V^2} + \sigma_S \sigma_V \rho \frac{\partial^2 U}{\partial S \partial V} + \frac{1}{2}\sigma_S^2 \frac{\partial^2 U}{\partial S^2} \\
&+ \mu_V \frac{\partial U}{\partial V} + \mu_S \frac{\partial U}{\partial S} + \frac{\partial U}{\partial t} - rU = 0.
\end{aligned}$$

For notational convenience the capitals and some subscripts can be dropped, resulting in a more elegant form of the general PDE:

$$\begin{aligned}
&\frac{1}{2}\sigma_v^2 \frac{\partial^2 u}{\partial v^2} + \sigma_s \sigma_v \rho \frac{\partial^2 u}{\partial s \partial v} + \frac{1}{2}\sigma_s^2 \frac{\partial^2 u}{\partial s^2} \\
&+ \mu_v \frac{\partial u}{\partial v} + \mu_s \frac{\partial u}{\partial s} + \frac{\partial u}{\partial t} - ru = 0.
\end{aligned} \tag{2.9}$$

2.1 Initial conditions

In option pricing, at maturity the value of the stock and its price process are known. The value of the derivative considered depends on this stock price (or process), and is thus known at maturity. This is called the final condition and can be denoted as: $u(s, v, T) = \phi(s, v)$. To sell or buy the derivative, its value at time $t = 0$ is needed so the reversed price process is considered. Therefore $\tau = T - t$ is introduced such that $u(s, v, r, \tau) = u(s, v, r, T - t)$, and $\frac{\partial u}{\partial \tau} = -\frac{\partial u}{\partial t}$ and the PDE becomes:

$$\begin{aligned}
\frac{\partial u}{\partial \tau} = &\frac{1}{2}\sigma_v^2 \frac{\partial^2 u}{\partial v^2} + \sigma_s \sigma_v \rho \frac{\partial^2 u}{\partial s \partial v} + \frac{1}{2}\sigma_s^2 \frac{\partial^2 u}{\partial s^2} \\
&+ \mu_v \frac{\partial u}{\partial v} + \mu_s \frac{\partial u}{\partial s} - ru,
\end{aligned} \tag{2.10}$$

with *initial* condition $u(s, v, 0) = \phi(s, v)$. As mentioned before, this (now) initial condition for the PDE is determined by the type of pay-off of the option. In this thesis two types of options are considered. A European call option gives the holder the right to buy a given asset at a prescribed maturity date T for a prescribed strike price $K > 0$. Using this fact the following pay-off function is used:

$$\phi(s, v) = \max(0, s - K). \tag{2.11}$$

An up-and-out call option gives the holder the right to buy a given asset (with price process S_τ) at a prescribed maturity date T for a prescribed strike price $K > 0$, only then when the price process S_τ does not exceed a prescribed barrier B. Using this fact the following pay-off function is used:

$$\phi(s, v) = \begin{cases} \max(0, s - K), & \text{when } s < B, \\ 0 & \text{otherwise.} \end{cases} \tag{2.12}$$

The restriction that the process can not hit the boundary B during time to maturity will be incorporated in the boundary conditions, described in section 4.

2.2 Heston PDE

The Heston model was first introduced in 1993 by S.L. Heston [1], and is a popular extension of the well known Black-Scholes formula. This model is an extension in the sense that the volatility also follows a stochastic process:

$$\begin{cases} dS_t & = rS_t dt + S_t \sqrt{V_t} dW_t^1, \\ dV_t & = \kappa(\eta - V_t)dt + \sigma\sqrt{V_t} dW_t^2, \\ dW_t^1 dW_t^2 & = \rho dt, \end{cases} \tag{2.13}$$

with the prescribed time zero (currently observed) values S_0 and V_0. Starting from (1.1) the Heston model uses the following coefficients: $\mu_s(s, v, t) = rs, \sigma_s(s, v, t) = \sqrt{v}s, \mu_v(s, v, t) = \kappa(\eta - v)$ and $\sigma_v(s, v, t) = \sigma\sqrt{v}$ such that the earlier derived general PDE (2.9) becomes:

$$\begin{aligned} &\frac{1}{2}\sigma^2 v \frac{\partial^2 u}{\partial v^2} + \sigma s v \rho \frac{\partial^2 u}{\partial s \partial v} + \frac{1}{2} v s^2 \frac{\partial^2 u}{\partial s^2} \\ &\quad + \kappa(\eta - v)\frac{\partial u}{\partial v} + rs\frac{\partial u}{\partial s} + \frac{\partial u}{\partial t} - ru = 0. \end{aligned} \tag{2.14}$$

An important feature in the Heston model is the so-called Feller condition:

$$\frac{2\kappa\eta}{\sigma^2} > 1. \tag{2.15}$$

When this condition is fulfilled the stochastic process of the volatility is strictly positive. For a detailed description, see [16]. From a computational point of view, when condition is not satisfied the paths in Monte Carlo simulations can go below zero. In PDE sense, the condition determines the behaviour close to the boundary at $v = 0$.

2.3 SABR PDE

The SABR model was introduced by Hagan et. al. in 2002 [8] and stands for: Stochastic Alpha Beta and Rho-model. Following earlier literature the volatility is denoted as α_t and the stock price process is assumed to be described as:

$$\begin{cases} df_t = \alpha_t f_t^{\beta} dW_t^1, \\ d\alpha_t = \sigma \alpha_t dW_t^2, \\ dW_t^1 dW_t^2 = \rho dt, \end{cases} \tag{2.16}$$

again with prescribed currently observed values f_0 and α_0. This model is defined on the forward price of the stock. When one wants to look at the stochastic process for the spot price, this can be derived using Itô calculus with the discounted price $S_t = D(t,T)f_t := e^{-r(T-t)}f_t$:

$$\begin{cases} dS_t = rS_t dt + D^{(1-\beta)}\alpha_t S_t^{\beta} dW_t^1, \\ d\alpha_t = \sigma \alpha_t dW_t^2, \\ dW_t^1 dW_t^2 = \rho dt. \end{cases}$$

Similar as in the Heston model, the price of the option is denoted as $u(s,\alpha,t)$ and because now $\mu_s(s,\alpha,t) = rs$, $\sigma_s(s,\alpha,t) = D^{(1-\beta)}\alpha s^{\beta}$, $\mu_v(s,\alpha,t) = 0$ and $\sigma_v(s,\alpha,t) = \sigma\alpha$ the general partial differential equation (2.9) becomes:

$$\frac{1}{2}\alpha^2 s^{2\beta} D^{2(1-\beta)} \frac{\partial^2 u}{\partial s^2} + \rho\sigma s^{\beta} D^{(1-\beta)} \alpha^2 \frac{\partial^2 u}{\partial s \partial \alpha} + \frac{1}{2}\sigma^2 \alpha^2 \frac{\partial^2 u}{\partial \alpha^2} \\ + \frac{\partial u}{\partial t} + rs\frac{\partial u}{\partial s} - ru = 0. \tag{2.17}$$

3 Finite difference schemes

The finite difference method for partial differential equations is relatively straightforward. First the region $\Omega \in \mathbb{R}^n$ on which the partial differential equation is defined needs to be discretized. This results in a finite grid, denoted as $\overline{\Omega} \in \Omega$, that approximates the continuous space in discrete grid points. With this discrete space, every continuous function $u \in C(\Omega)$ can be approximated with function values of u on $\overline{\Omega}$, let this approximation of u be denoted as $\overline{u}$. Essential to the quality of this approximation is the number of grid points to be used. When more grid points are used there are more points where the function value is approximated. The finite difference method will approach $\overline{u}$ with the help of approximations of the partial derivatives of $\overline{u}$. In this thesis, first the discretization in space direction will be derived. When considering an asset price process with stochastic volatility the space will consist of s and v. With this discretization a finite number of ordinary differential equations are constructed. Subsequently, the discretization in time will be discussed. With the help of these discretizations and with the initial solution of a partial differential equation including boundary conditions, a solution can be constructed.

3.1 Space discretization

First an equidistant grid in both the price as the volatility direction is applied. Later on a non-uniform grid will be used. The uniform case can easily be derived from the non-uniform case. Let m_1 denote the number of grid points to be used in s-direction and m_2 the number of grid points to be used in v-direction, such that $0 = s_0 < s_1, \ldots, s_{m_1} = S_{\max}$ and $0 = v_0 < v_1, \ldots, v_{m_2} = V_{\max}$. Next, the differences Δs_i and Δv_j can be defined as

$$\Delta s_i = s_i - s_{i-1} \text{ and } \Delta v_j = v_j - v_{j-1}. \tag{3.1}$$

In the uniform case, the discrete grid points will be distributed equidistantly in s and v direction such that

$$\Delta s = \frac{S_{\max}}{m_1} \text{ and } \Delta v = \frac{V_{\max}}{m_2}$$

will denote the mesh widths and grid point (s_i, v_j) will equal $(i\Delta s, j\Delta v)$. The spatial domain is thus subdivided in the following grid points:

$$\mathcal{G} = \{(s_i, v_j) : 0 \leq i \leq m_1, 0 \leq j \leq m_2\}. \tag{3.2}$$

The function value at grid point (s_i, v_j) at time t is denoted as $u_{i,j}(t)$ (note that for notational convenience less-relevant sub- or superscripts will be dropped). Let δ_s (or δ_v) denote the differential operator w.r.t. s (or v respectively). All finite difference schemes are derived using Taylor expansions. These expansions are used to express the function value at, for example, (s_{i+1}, v_j), namely $u_{i+1,j}$, in terms of function values at (s_i, v_j) denoted as $u_{i,j}$. In formula one has:

$$u_{i+1,j} = u_{i,j} + \Delta s_{i+1} \left(\frac{\partial u}{\partial s}\right)_{i,j} + \frac{\Delta s_{i+1}^2}{2!} \left(\frac{\partial^2 u}{\partial s^2}\right)_{i,j} + \frac{\Delta s_{i+1}^3}{3!} \left(\frac{\partial^3 u}{\partial s^3}\right)_{i,j} + \ldots \tag{3.3}$$

Because computers can only handle a finite number of terms, the Taylor series needs to be truncated. This is done via the so called "big $\mathcal{O}$" notation, defined as follows:

Definition 3.1. Let f(x) and g(x) be two functions defined on some subset of the real numbers. Then

$$f(x) = \mathcal{O}(g(x))$$

if and only if there exists a positive real number M and a real number x_0 such that

$$|f(x)| \leq Mg(x)| \text{ for all } x > x_0.$$

In practice this notation gives one the ability to truncate (3.3) to:

$$u_{i+1,j} = u_{i,j} + \Delta s_{i+1} \left(\frac{\partial u}{\partial s}\right)_{i,j} + \frac{\Delta s_{i+1}^2}{2!} \left(\frac{\partial^2 u}{\partial s^2}\right)_{i,j} + \mathcal{O}(\Delta s_{i+1}^3), \tag{3.4}$$

because it holds that

$$\left| \frac{\Delta s_{i+1}^3}{3!} \left(\frac{\partial^3 u}{\partial s^3}\right)_{i,j} + \frac{\Delta s_{i+1}^4}{4!} \left(\frac{\partial^4 u}{\partial s^4}\right)_{i,j} + \ldots \right| \leq M \left|\Delta s_{i+1}^3\right| \text{ when } \Delta s_{i+1} \to 0.$$

In words one says that (3.4) approximates (3.3) up to third order accuracy. So, in general, $\mathcal{O}(\Delta x^p)$ assembles all terms that have leading term equal to or lower than Δx^p. When $\Delta x \to 0$ this term will decrease to 0 faster for higher values of p. The finite difference scheme approximates the derivative in every grid point (s_i, v_j) using finite differences, e.g. the difference between neighbouring points (s_{i-1}, v_j), (s_i, v_j) and (s_{i+1}, v_j). First the symmetric central approach is treated.

3.1.1 Central difference scheme

The central character of the scheme lies in the fact that the scheme uses two direct neighbours. So, to approximate the value of the first derivative in point (s_i, v_j) in s direction, the Taylor expansions of $u_{i+1,j}$, $u_{i,j}$ and $u_{i-1,j}$ are used:

$$\begin{aligned}
\beta_{+1} u_{i+1,j} &= \beta_{+1} \left(u_{i,j} + \Delta s_{i+1} \left(\frac{\partial u}{\partial s}\right)_{i,j} + \frac{{\Delta s_{i+1}}^2}{2} \left(\frac{\partial^2 u}{\partial s^2}\right)_{i,j} + \mathcal{O}({\Delta s_{i+1}}^3) \right), \\
\beta_0 u_{i,j} &= \beta_0 u_{i,j}, \\
\beta_{-1} u_{i-1,j} &= \beta_{-1} \left(u_{i,j} - \Delta s_i \left(\frac{\partial u}{\partial s}\right)_{i,j} + \frac{{\Delta s_i}^2}{2} \left(\frac{\partial^2 u}{\partial s^2}\right)_{i,j} + \mathcal{O}({\Delta s_i}^3) \right).
\end{aligned}$$

Now we can choose β_{+1}, β_0 and β_{-1} such that the leading term cancels, the second derivative remains and the second order terms also cancel to get a second order approximation of the derivative:

$$\left.\begin{array}{l}
\beta_{+1} + \beta_0 + \beta_{-1} = 0, \\
\beta_{+1} \Delta s_{i+1} + \beta_{-1} \Delta s_i = 1, \\
\beta_{+1} \frac{\Delta s_{i+1}^2}{2} + \beta_{-1} \frac{\Delta s_i^2}{2} = 0.
\end{array}\right\} \Rightarrow
\begin{array}{l}
\beta_{+1} = \frac{\Delta s_i}{\Delta s_{i+1}(\Delta s_i + \Delta s_{i+1})}, \\
\beta_0 = \frac{-\Delta s_i + \Delta s_{i+1}}{\Delta s_i \Delta s_{i+1}}, \\
\beta_{-1} = \frac{-\Delta s_{i+1}}{\Delta s_i(\Delta s_i + \Delta s_{i+1})}.
\end{array} \tag{3.5}$$

In the notationally more elegant uniform case $\Delta s_i = \Delta s_j = \Delta s, \forall i, j$, this simply becomes:

$$u_{i+1,j} - u_{i-1,j} = 2\Delta s \left(\frac{\partial u}{\partial s}\right)_{i,j} + \mathcal{O}(\Delta s^3) \iff \tag{3.6}$$

$$\left(\frac{\partial u}{\partial s}\right)_{i,j} = \frac{u_{i+1,j} - u_{i-1,j}}{2\Delta s} + \mathcal{O}(\Delta s^2). \tag{3.7}$$

The same can be done in v direction. For the second derivative w.r.t. s the function value at $u_{i,j}$ is also used:

$$
\begin{array}{rl}
u_{i+1,j} = & u_{i,j} + \Delta s \left(\frac{\partial u}{\partial s}\right)_{i,j} + \frac{\Delta s^2}{2}\left(\frac{\partial^2 u}{\partial s^2}\right)_{i,j} + \frac{\Delta s^3}{3!}\left(\frac{\partial^3 u}{\partial s^3}\right)_{i,j} + \mathcal{O}(\Delta s^4),\\
-2\cdot u_{i,j} = & -2\cdot u_{i,j}\\
u_{i-1,j} = & u_{i,j} - \Delta s \left(\frac{\partial u}{\partial s}\right)_{i,j} + \frac{\Delta s^2}{2}\left(\frac{\partial^2 u}{\partial s^2}\right)_{i,j} - \frac{\Delta s^3}{3!}\left(\frac{\partial^3 u}{\partial s^3}\right)_{i,j} + \mathcal{O}(\Delta s^4) + \\
\hline
u_{i+1,j} - 2u_{i,j} + u_{i-1,j} = & \Delta s^2 \left(\frac{\partial^2 u}{\partial s^2}\right)_{i,j} + \mathcal{O}(\Delta s^4) \iff \\
\left(\frac{\partial^2 u}{\partial s^2}\right)_{i,j} = & \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{\Delta s^2} + \mathcal{O}(\Delta s^2).
\end{array}
$$

In the uniform case, the following well known set of central finite difference approximations will be used for internal mesh points:

$$\delta_s\left[u_{i,j}\right] \approx \frac{u_{i+1,j} - u_{i-1,j}}{2\Delta s}, \tag{3.8a}$$

$$\delta_v\left[u_{i,j}\right] \approx \frac{u_{i,j+1} - u_{i,j-1}}{2\Delta v}, \tag{3.8b}$$

$$\delta_s^2\left[u_{i,j}\right] \approx \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{\Delta s^2}, \tag{3.8c}$$

$$\delta_v^2\left[u_{i,j}\right] \approx \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{\Delta v^2}. \tag{3.8d}$$

In addition, by linearity of the operators, the cross derivative $\frac{\partial^2 u}{\partial s \partial v}$ can be approximated by subsequently applying (3.8a) and (3.8b):

$$
\begin{aligned}
\delta_{s,v}\left[u_{i,j}\right] = \delta_s\left[\delta_v\left[u_{i,j}\right]\right] \approx \delta_s\left[\frac{u_{i,j+1} - u_{i,j-1}}{2\Delta v}\right] &= \frac{\delta_s[u_{i,j+1}] - \delta_s[u_{i,j-1}]}{2\Delta v},\\
&\approx \frac{\frac{u_{i+1,j+1} - u_{i-1,j+1}}{2\Delta s} - \frac{u_{i+1,j-1} - u_{i-1,j-1}}{2\Delta s}}{2\Delta v},\\
&= \frac{u_{i+1,j+1} - u_{i-1,j+1} - u_{i+1,j-1} + u_{i-1,j-1}}{4\Delta v \Delta s}.
\end{aligned} \tag{3.8e}
$$

3.1.2 One sided schemes

Because the central scheme has some drawbacks which will be discussed later, and because the scheme is not applicable to the entire domain, also one sided schemes are used. The earlier defined space discretization implies that at the boundaries of this domain there are no two direct neighbours to approximate the derivative at this point. This is why at a boundary, central difference cannot be applied. All grid points $u_{i,0}$ do not have two direct neighbours in v-direction. For these boundary points, so called one-sided finite difference schemes can be applied. For example at the $v = 0$ boundary, the first derivative can be approximated with the help of the one-sided forward difference scheme. Again, this approximation is derived using the Taylor expansions and to keep the second order behaviour, three solution values $u_{i,0}, u_{i,1}$ and

$u_{i,2}$ are used:

$$\begin{aligned} \gamma_0 u_{i,0} + \gamma_1 u_{i,1} + \gamma_2 u_{i,2} = u_{i,0}\gamma_0 + &\gamma_1(u_{i,0} + \Delta v \left(\frac{\partial u}{\partial v}\right)_{i,0} + \frac{\Delta v^2}{2}\left(\frac{\partial^2 u}{\partial v^2}\right)_{i,0} + \mathcal{O}(\Delta v^3)), \\ &+\gamma_2(u_{i,0} + 2\Delta v \left(\frac{\partial u}{\partial v}\right)_{i,0} + \frac{(2\Delta v)^2}{2}\left(\frac{\partial^2 u}{\partial v^2}\right)_{i,0} + \mathcal{O}(\Delta v^3)), \\ &= u_{i,0}(\gamma_0 + \gamma_1 + \gamma_2) + \left(\frac{\partial u}{\partial v}\right)_{i,0}(\gamma_1 \Delta v + 2\gamma_2 \Delta v) \\ &+ \left(\frac{\partial^2 u}{\partial v^2}\right)_{i,0}(\gamma_1 \frac{\Delta v^2}{2} + 2\gamma_2 \Delta v^2) + \mathcal{O}(\Delta v^3). \end{aligned}$$

Now γ_1, γ_2 and γ_3 can be chosen to obtain a second order approximation of $\left(\frac{\partial u}{\partial v}\right)_{i,0}$:

$$\left.\begin{array}{l} \gamma_0 + \gamma_1 + \gamma_2 = 0, \\ \gamma_1 \Delta v + 2\gamma_2 \Delta v = 1, \\ \gamma_1 \frac{\Delta v^2}{2} + 2\gamma_2 \Delta v^2 = 0. \end{array}\right\} \Rightarrow \gamma_0 = \frac{-3}{2\Delta v}, \gamma_1 = \frac{4}{2\Delta v} \text{ and } \gamma_2 = \frac{-1}{2\Delta v},$$

copying the notation from above and adding an index to emphasize the forward character, this operator is denoted as:

$$\delta_v^+ [u_{i,0}] \approx \frac{-3u_{i,0} + 4u_{i,1} - u_{i,2}}{2\Delta v}. \tag{3.9}$$

In general one sided scheme can also be applied to points that are not on the boundary. Because of some drawbacks of the central scheme that will be discussed in section 7, a one sided alternative is used to approximate derivatives at interior grid points. When the coefficients μ_s or μ_v become negative, the backward difference scheme will be used for the first order derivatives. This is done to alleviate spurious oscillations caused by the possible negative sign of these coefficients. For example in the Heston case, one has $\mu_v = \kappa(\eta - v)$ which clearly becomes negative for $v > \eta$. The disturbing effect of using a central scheme when the sign is negative will be shown later on. The derivation is similar as in the forward situation:

$$\delta_v^- [u_{i,j}] \approx \frac{u_{i,j-2} - 4u_{i,j-1} + 3u_{i,j}}{2\Delta v}. \tag{3.10}$$

Although it is not used in this thesis, the one sided second order accurate approximations of the second derivative is given below:

$$\begin{aligned} \delta_v^{2+} [u_{i,j}] &\approx \frac{2u_{i,j} - 5u_{i,j+1} + 4u_{i,j+2} - u_{i,j+3}}{\Delta v^2}, \\ \delta_v^{2-} [u_{i,j}] &\approx \frac{-u_{i,j-3} + 4u_{i,j-2} - 5u_{i,j-1} + 2u_{i,j}}{\Delta v^2}. \end{aligned}$$

the non-uniform versions can be derived similar as in (3.5) using the appropriate Taylor series.

3.1.3 Non-uniform difference schemes

The non-uniform difference schemes can be derived similar as done in (3.5). Following [4], one has:

$$\delta_s^- \left[u_{i,j}\right] = \left(\frac{\partial u}{\partial s}\right)_{i,j} \approx \alpha_{-2} u_{i-2,j} + \alpha_{-1} u_{i-1,j} + \alpha_0 u_{i,j}, \tag{3.11a}$$

$$\delta_s \left[u_{i,j}\right] = \left(\frac{\partial u}{\partial s}\right)_{i,j} \approx \beta_{-1} u_{i-1,j} + \beta_0 u_{i,j} + \beta_{+1} u_{i+1,j}, \tag{3.11b}$$

$$\delta_s^+ \left[u_{i,j}\right] = \left(\frac{\partial u}{\partial s}\right)_{i,j} \approx \gamma_0 u_{i,j} + \gamma_{+1} u_{i+1,j} + \gamma_{+2} u_{i+2,j}, \tag{3.11c}$$

$$\delta_s^2 \left[u_{i,j}\right] = \left(\frac{\partial^2 u}{\partial s^2}\right)_{i,j} \approx \delta_{-1} u_{i-1,j} + \delta_0 u_{i,j} + \delta_{+1} u_{i+1,j}, \tag{3.11d}$$

where

$$\begin{array}{rlrlrl}
\alpha_{-2} = & \frac{\Delta s_i}{\Delta s_{i-1}(\Delta s_{i-1}+\Delta s_i)} & \alpha_{-1} = & \frac{-\Delta s_{i-1}-\Delta s_i}{\Delta s_{i-1}\Delta s_i} & \alpha_0 = & \frac{\Delta s_{i-1}+2\Delta s_i}{\Delta s_i(\Delta s_{i-1}+\Delta s_i)}, \\
\beta_{-1} = & \frac{-\Delta s_{i+1}}{\Delta s_i(\Delta s_i+\Delta s_{i+1})} & \beta_0 = & \frac{-\Delta s_i+\Delta s_{i+1}}{\Delta s_i\Delta s_{i+1}} & \beta_{+1} = & \frac{\Delta s_i}{\Delta s_{i+1}(\Delta s_i+\Delta s_{i+1})}, \\
\gamma_0 = & \frac{-2\Delta s_{i+1}-\Delta s_{i+2}}{\Delta s_{i+1}(\Delta s_{i+1}+\Delta s_{i+2})} & \gamma_{+1} = & \frac{\Delta s_{i+1}+\Delta s_{i+2}}{\Delta s_{i+1}\Delta s_{i+2}} & \gamma_{+2} = & \frac{-\Delta s_{i+1}}{\Delta s_{i+2}(\Delta s_{i+1}+\Delta s_{i+2})}, \\
\delta_{-1} = & \frac{2}{\Delta s_i(\Delta s_i+\Delta s_{i+1})} & \delta_0 = & \frac{-2}{\Delta s_i\Delta s_{i+1}} & \delta_{+1} = & \frac{2}{\Delta s_{i+1}(\Delta s_i+\Delta s_{i+1})}.
\end{array}$$

These schemes are similar in v-direction and again the mixed derivative can be approximated by applying (3.11b) consecutive in v- and s-direction. Other than in the uniform case, the order of accuracy depends on the differences that are used at the specific grid point. To be able to do a similar accuracy study for non-uniform grids, these non-uniform grids must satisfy a certain smoothness condition. A part of this condition is that for all i there is a Δs and a positive real constant C_1 such that

$$\Delta s_i \leq C_1 \Delta s.$$

Here Δs stems from the construction of the non-uniform grid. This will be treated in more detail in section 6. Having such a Δs, similar as in the uniform case, one can state the accuracy of these schemes. In this sense, these derived schemes are second order ($\mathcal{O}(\Delta s^2)$) accurate approximations.

3.1.4 System of ODEs

Because all the spatial derivatives are expressed in terms of discrete function values, the partial differential equation can be expressed in terms of $u_{i,j}$ and $\left(\frac{\partial u}{\partial t}\right)_{i,j}$. Together with the initial value, this leads to an initial value problem for a system of ordinary differential equations (ODEs). To simplify the notation let the solution vector $(u_1, u_2, \ldots, u_n)^T$ be denoted by $U(t)$, its derivative w.r.t. t by $U'(t)$, the coefficient matrix by A and the vector that is determined by the boundary conditions by R, then, together with the initial condition the following system of ordinary differential equations is derived:

$$\begin{array}{rcl}
U'(t) & = & AU(t) + R \quad 0 \leq t \leq T, \\
U(0) & = & f(x).
\end{array} \tag{3.12}$$

This procedure can be extended to more than one space dimension. In case of $m_1 + 1$ and $m_2 + 1$ grid points in s and v direction respectively A will be a square matrix of size $(m_1 + 1)(m_2 + 1) \times$

$(m_1+1)(m_2+1)$ and R an $(m_1+1)(m_2+1)\times 1$ vector determined by the boundary conditions. In the higher dimensional case, the solution vector needs to be ordered in a convenient way. In this thesis the following lexicographic ordering is used for the solution vector U:

$$U = \left(u_{0,0}, \cdots, u_{0,m_2}, \cdots, \cdots, u_{i,0}, \cdots, u_{i,m_2}, \cdots, \cdots, u_{m_1,0}, \cdots, u_{m_1,m_2}\right)^T.$$

For coding reasons (index 0 does not exist in Matlab) we let the indices run from 1 to (m_1+1) and (m_2+1) respectively, where $u^k_{1,1} = u(s_1, v_1, t_k) = u(0, 0, t_k)$. To get a better insight in the procedure, the following one dimensional example is helpful.

Example 3.2. Define the following problem on $[0, 1]$:

$$\begin{cases} \frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2}, & \\ u(t,0) = b, & t \in [0,T] \\ \frac{\partial u}{\partial x}(t,1) = c, & \\ u(0,x) = f(x). & \end{cases} \tag{3.13}$$

First, the spatial domain can be uniformly discretized in n intervals of length $\Delta x := \frac{1}{n}$ as follows: $0 = x_1 < x_2 < \ldots < x_n < x_{n+1} = 1$. Notice that because of the Dirichlet boundary at $x = 0$, the function value at $x = 0$ is already known, so instead of $n+1$ unknown function values, there are only n unknown function values: u_i for $i = 2, \ldots, n+1$, or in vector notation, $(u_2, u_3, \ldots, u_{n+1})^T$. Because of this Dirichlet boundary at $x_1(= 0)$, the second derivative at $x_1(= 0)$ is not needed. For the points x_i for $i = 2, \ldots, n$ the second order derivative w.r.t. x can be approximated using the central scheme:

$$(u_i)_{xx} \approx \frac{u_{i+1} - 2u_i + u_{i-1}}{\Delta x^2}, \tag{3.14}$$

or in matrix notation:

$$\begin{pmatrix} (u_2)_{xx} \\ (u_3)_{xx} \\ \vdots \\ \vdots \\ \vdots \\ (u_{n+1})_{xx} \end{pmatrix} \approx \frac{1}{\Delta x^2} \begin{pmatrix} k & l & & & & \\ 1 & -2 & 1 & & & \\ 0 & 1 & -2 & 1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & 1 & -2 & 1 \\ & & & & q & r \end{pmatrix} \begin{pmatrix} u_2 \\ u_3 \\ \vdots \\ \vdots \\ \vdots \\ u_{n+1} \end{pmatrix}. \tag{3.15}$$

For the derivative at grid point x_2 the central scheme uses function values $u(t, x_1)$, $u(t, x_2)$ and $u(t, x_3)$ and because the function value at $x = 0$ equals b, it holds that:

$$(u_2)_{xx} \approx \frac{u_3 - 2u_2 + u_1}{\Delta x^2} = \frac{u_3 - 2u_2}{\Delta x^2} + \frac{1}{\Delta x^2} b,$$

such that $k = -2$ and $l = 1$ and the vector $(\frac{b}{\Delta_x^2}, 0, \ldots, 0)^T$ is added to approximation (3.15). At the $x = 1$ boundary, the central scheme needs the function value of the non existing "ghost"-point u_{n+2}. Using the information of the Neumann condition, this function value can be approximated. This is done with the help of linear extrapolation. The continuous condition $u_x(t, 1) = c$ can be translated into the discrete variant using the earlier discussed central scheme for the first derivative: $u_x(t, 1) \approx \frac{u_{n+2} - u_n}{2\Delta x}$, this implies: $u_{n+2} \approx 2\Delta x c + u_n$. Using this to approximate the second derivative at $x_{n+1} = 1$ yields:

$$(u_{n+1})_{xx} \approx \frac{u_n - 2u_{n+1} + u_{n+2}}{\Delta x^2} \approx \frac{2u_n - 2u_{n+1} + 2\Delta x c}{\Delta x^2},$$

So $q = 2$ and $r = -2$ and the $\frac{2\Delta xc}{\Delta x^2}$-term is stored in the vector: $(\frac{b}{\Delta x^2}, 0, \ldots, 0, \frac{2c}{\Delta x})^T$. Resulting in the following second order approximation of u_{xx}:

$$\begin{pmatrix} (u_2)_{xx} \\ (u_3)_{xx} \\ \vdots \\ \vdots \\ \vdots \\ (u_{n+1})_{xx} \end{pmatrix} \approx \frac{1}{\Delta x^2} \begin{pmatrix} -2 & 1 & & & & \\ 1 & -2 & 1 & & & \\ 0 & 1 & -2 & 1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & 1 & -2 & 1 \\ & & & & 2 & -2 \end{pmatrix} \begin{pmatrix} u_2 \\ u_3 \\ \vdots \\ \vdots \\ \vdots \\ u_n \end{pmatrix} + \begin{pmatrix} \frac{b}{\Delta_x^2} \\ 0 \\ \vdots \\ \vdots \\ 0 \\ \frac{2c}{\Delta_x} \end{pmatrix}. \tag{3.16}$$

3.2 Time discretization

With the space discretization above and a starting vector, the solution at any time instance can be calculated using a stepping procedure. This can be done for example using the θ-method which is a combination of an implicit method and an explicit method. Similar as in the space discretization the time dimension is discretized. In this thesis maturity time T is set to 1, the number of discrete grid points in time direction is denoted as T_N and the finite difference in time direction is denoted as Δt. The later derived ADI scheme uses a first order accurate approximation for the time derivative. Again this is derived using Taylor series as follows:

$$\begin{aligned} u^{k+1} &= u^k + \frac{\Delta t}{1!}\left(\frac{\partial u}{\partial t}\right)^k + \frac{\Delta t^2}{2!}\left(\frac{\partial^2 u}{\partial t^2}\right)^k + \mathcal{O}(\Delta t^3), \\ \Rightarrow \left(\frac{\partial u}{\partial t}\right)^k &= \frac{u^{k+1} - u^k}{\Delta t} + \mathcal{O}(\Delta t). \end{aligned} \tag{3.17}$$

Let $\mathcal{D}_{\Delta s,\Delta v}(u^k_{i,j})$ denote a finite difference operator in v and s. For example for the general PDE (2.9), its defined as:

$$\begin{aligned} \mathcal{D}_{\Delta s,\Delta v}(u^k_{i,j}) :=& \frac{1}{2}\sigma_v^2\delta_v^2\left[u^k_{i,j}\right] + \sigma_s\sigma_v\rho\delta_{sv}\left[u^k_{i,j}\right] + \frac{1}{2}\sigma_s^2\delta_s^2\left[u^k_{i,j}\right] \\ &+ \mu_v\delta_v\left[u^k_{i,j}\right] + \mu_s\delta_s\left[u^k_{i,j}\right] - ru^k_{i,j}. \end{aligned} \tag{3.18}$$

, then the Euler forward method uses the following time discretization:

$$\frac{u^{k+1}_{i,j} - u^k_{i,j}}{\Delta t} = \mathcal{D}_{\Delta s,\Delta v}(u^k_{i,j}),$$

which results in the following explicit expression:

$$u^{k+1}_{i,j} = u^k_{i,j} + \Delta t\mathcal{D}_{\Delta s,\Delta v}(u^k_{i,j}).$$

The Euler backward method discretizes the time derivative in a similar way, but the space derivatives are taken at time level $k+1$ instead of k:

$$\frac{u^{k+1}_{i,j} - u^k_{i,j}}{\Delta t} = \mathcal{D}_{\Delta s,\Delta v}(u^{k+1}_{i,j}),$$

resulting in the following implicit formula:

$$u^{k+1}_{i,j} - \Delta t\mathcal{D}_{\Delta s,\Delta v}(u^{k+1}_{i,j}) = u^k_{i,j}.$$

The implicit property of the backward scheme lies in the fact that the solution at the next time level, $u_{i,j}^{k+1}$ is given as an unknown in an equation, while in the Euler forward case $u_{i,j}^{k+1}$ is given explicitly. Now, starting with an initial solution and using the forward or backward scheme, a linear equation can be constructed which calculates the value of u at any grid point at the next time step. Let U^k denote the solution vector at $t = k$. Such that in matrix notation one has $U'(t = k) \approx \frac{U^{k+1}-U^k}{\Delta_t}$ in the forward case. Now equation (3.12) can be fully expressed in U^{k+1} and U^k such that in the forward case one has:

$$\begin{aligned}\frac{U^{k+1} - U^k}{\Delta_t} &= AU^k + R,\\ U^0 &= f(x),\end{aligned}$$

and in the backward case:

$$\begin{aligned}\frac{U^{k+1} - U^k}{\Delta_t} &= AU^{k+1} + R,\\ U^0 &= f(x).\end{aligned}$$

In either cases, the solution at a next time level can be obtained by a matrix multiplication and adding a vector:

$$\begin{aligned}U^{k+1} &= QU^k + G,\\ U^0 &= f(x),\end{aligned} \tag{3.19}$$

where Q is a matrix and G a vector both depending on the method used. The most important difference between these methods is that the backward scheme requires a matrix inversion to determine Q and G, while with the help of the forward scheme the solution at the next time level can be obtained without a matrix inversion. In literature, these methods are mostly referred to as implicit (Euler backward) or explicit methods (Euler forward). Both methods have advantages and disadvantages. The implicit method is unconditionally stable, but the expensive matrix inversion is needed. The explicit method is straightforward and no matrix inversion is needed, but this method is not unconditionally stable. A combination of these two methods, so called Implicit-Explicit methods (IMEX-methods), can be used.

3.3 θ-method

A widely applied IMEX-scheme is the θ-method. This method combines the forward and backward scheme with the help of a scaling parameter $\theta \in [0, 1]$:

$$\frac{u_{i,j}^{k+1} - u_{i,j}^{k}}{\Delta_t} = \theta \mathcal{D}_{\Delta s,\Delta v}(u_{i,j}^{k+1}) + (1-\theta)\mathcal{D}_{\Delta s,\Delta v}(u_{i,j}^{k})$$

For $\theta = 1$ the equation is fully implicit, for $\theta = 0$ the equation is fully explicit. The ADI method, to be introduced in the next section, uses both the implicit and the explicit method. In matrix notation the θ-scheme becomes:

$$\begin{aligned}\frac{1}{\Delta_t}\left(U^{k+1} - U^k\right) &= \theta\left(AU^{k+1} + R\right) + (1-\theta)\left(AU^k + R\right) \Leftrightarrow\\ U^{k+1} &= U^k + \Delta_t\left[\theta\left(AU^{k+1} + R\right) + (1-\theta)\left(AU^k + R\right)\right].\end{aligned} \tag{3.20}$$

Together with the initial solution, any solution can be calculated using time stepping.

3.4 The Greeks

In finance, the Greeks play an important role. Quoting Wikipedia:

Any trader worth his or her salt will know the Greeks and make a choice of which Greeks to hedge to limit exposure.

Next to the price of an option, one wants to know how the value of this option changes when the market conditions change (e.g. for hedging purposes). The Greeks measure the sensitivity of the option price to a set of particular parameters. In practice, one is usually interested in the sensitivities to the market parameters (e.g. implied volatility, spot, correlation). The model parameters are derived from the market parameters during the calibration procedure. This is why the sensitivities with respect to the model parameters and state variables are also of interest. Luckily mathematics cover this concept in the form of partial derivatives. FDM has an important advantage in computing these partial derivatives. Because the method produces option prices for a whole range of stock prices and volatility values, the partial derivatives with respect to these parameters can be computed by simply using the finite difference schemes derived in this section. The most important is the sensitivity of the option price to a slight change of the price of the underlying stock. This Greek is called Δ^{mp}, where mp stands for model parameters denoting the fact that it is computed with respect to these parameters. This is thus defined as the first derivative with respect to s:

$$\Delta^{\mathrm{mp}} := \frac{\partial u}{\partial s} \tag{3.21a}$$

Next to Δ^{mp}, in this thesis the following Greeks are computed:

$$\Gamma^{\mathrm{mp}} := \frac{\partial^2 u}{\partial s^2}, \tag{3.21b}$$

$$\nu^{\mathrm{mp}} := \frac{\partial u}{\partial \sigma} \tag{3.21c}$$

One keeps in mind that although the last Greek is actually the Greek letter *nu*, it is pronounced as *vega*. Probably because the Greek letter ν looks like the first letter v from volatility and the world of finance recognized a habit in translating the Western letters a, b and z in Latin as *alpha*, *beta* and *zeta*. Because the finite difference method calculates option values for a range of values of s and v, these derivatives can be computed using finite differences once again. For example for Δ, one has:

$$\Delta^{\mathrm{mp}}(s_i, v_j) := \frac{\partial u_{i,j}}{\partial s} = \frac{u_{i-1,j} - u_{i+1,j}}{2\Delta s}, \tag{3.22}$$

for $i = 2, \ldots, m1 - 1$ and $j = 1, \ldots, m_2$. At the boundaries the one sided alternatives from section 3.1.2 can be applied.

4 Boundary conditions

In theory the stock price s and the volatility v can reach a whole range of positive values during the time to maturity. However it is assumed that these values cannot become negative. Therefore there exists a boundary at $s = 0$ and $v = 0$ or $\alpha = 0$ in the SABR case. Besides this, some derivatives have no other conditions on the value of the stock or volatility during the process, so in theory, an infinite domain should be considered. Unfortunately, to implement the finite difference method on a computer, only a finite number of grid points can be considered. To address this problem, artificial boundary conditions are imposed.

4.1 Boundary conditions for a European call option

In a European call option, the stock price and the volatility can attain every possible positive value. This implies that two artificial boundaries are needed. Two types of boundary conditions can be considered, the so called: *Dirichlet*- and *Neumann*-boundary conditions. The *Dirichlet*-boundary condition assumes the *value* of the function at the boundary is known, while the *Neumann*-boundary condition assumes the *normal derivative value* of the function at the boundary to be known. These boundaries should be set far away from the region of interest to minimize the effect on the solution. The spatial domain is restricted to the bounded set: $[0, S_{\max}] \times [0, V_{\max}]$ with fixed values $S_{\max}$ and $V_{\max}$ chosen large enough (for example in [4] $S_{\max}$ and $V_{\max}$ are set at $14K$ and 10 respectively). In the case of a European call option the following boundary conditions are imposed,

$$\textcircled{1} \quad u(0, v, \tau) = 0, \tag{4.1a}$$

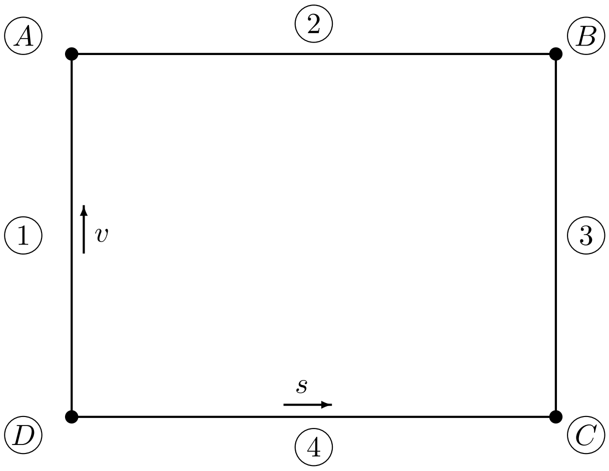
$$\textcircled{2} \quad u(s, V_{\max}, \tau) = s, \tag{4.1b}$$

$$\textcircled{3} \quad \frac{\partial u}{\partial s}(S_{\max}, v, \tau) = 1. \tag{4.1c}$$

Because the derived PDE has two second derivatives in two space directions, four boundary conditions are needed. This comes from the fact that the two second order derivatives give rise to two unknown integration constants. To meet this requirement, at the boundary $v = 0$ it is considered inserting $v = 0$ into the PDE to complete the set of four boundary conditions:

$$\textcircled{4} \quad \frac{\partial u}{\partial \tau}(s, 0, \tau) = \tfrac{1}{2}\sigma_v^2 \tfrac{\partial^2 u}{\partial v^2} + \sigma_s \sigma_v \rho \tfrac{\partial^2 u}{\partial s \partial v} + \tfrac{1}{2}\sigma_s^2 \tfrac{\partial^2 u}{\partial s^2} + \mu_v \tfrac{\partial u}{\partial v} + \mu_s \tfrac{\partial u}{\partial s} - ru \ . \tag{4.1d}$$

The next picture gives a schematic overview:



For the corners these boundary conditions yield:

Ⓐ $\lim\limits_{v\uparrow V_{\max}} u(0,v,\tau) = \lim\limits_{s\downarrow 0} u(s,V_{\max},\tau) = 0,$

Ⓑ from ② one has $u(s,V_{\max},\tau) = s$ which implies $\dfrac{\partial u}{\partial s}(s,V_{\max},\tau) = 1,$

so specifically, at $s = S_{\max}$ it holds that $\dfrac{\partial u}{\partial s}(S_{\max},V_{\max},\tau) = 1,$

Ⓒ from ③ one has $\dfrac{\partial u}{\partial s}(S_{\max},0,\tau) = 1,$ inserting this in ④ yields:

$$\frac{\partial u}{\partial \tau}(S_{\max},0,\tau) = \frac{1}{2}\sigma_v^2\frac{\partial^2 u}{\partial v^2} + \sigma_s\sigma_v\rho\frac{\partial^2 u}{\partial s\partial v} + \frac{1}{2}\sigma_s^2\frac{\partial^2 u}{\partial s^2} + \mu_v\frac{\partial u}{\partial v} + \mu_s - ru,$$

Ⓓ from ① one has $u(0,0,\tau) = 0$ such that ④ becomes:

$$\frac{\partial u}{\partial \tau}(0,0,\tau) = \frac{1}{2}\sigma_v^2\frac{\partial^2 u}{\partial v^2} + \sigma_s\sigma_v\rho\frac{\partial^2 u}{\partial s\partial v} + \frac{1}{2}\sigma_s^2\frac{\partial^2 u}{\partial s^2} + \mu_v\frac{\partial u}{\partial v} + \mu_s\frac{\partial u}{\partial s}.$$

The first Dirichlet boundary condition (4.1a) is obvious, when the stock price tends to zero also the option value will tend to zero. For the second Neumann boundary condition (4.1c) one reasons as follows. For large stock values the change of the option value in time will decrease. So the value of the stock will be equal to the stock value minus the discounted strike price. Because the limiting behaviour of an option is independent of the type of model that is considered, the limiting behaviour can also be shown using the traditional Black-Scholes model:

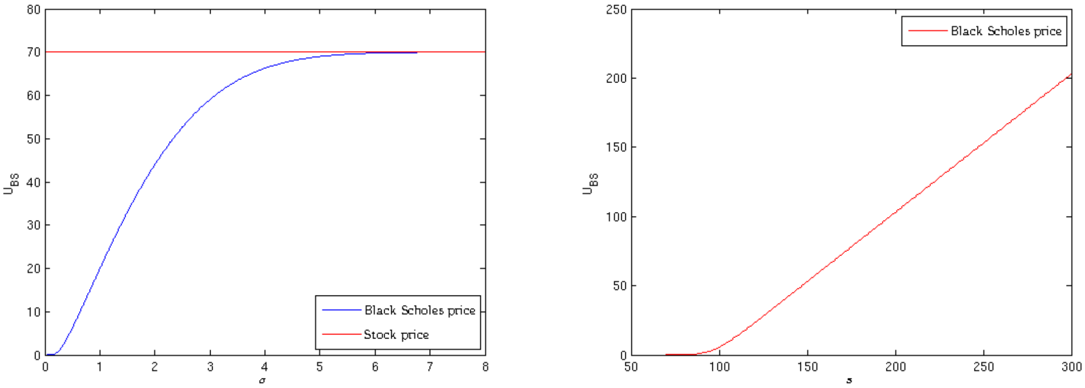
$$u(S,\tau) = N(d_1)S - N(d_2)Ke^{-r(T-\tau)},$$
$$d_{1,2} = \frac{\ln\left(\frac{S}{K}\right) + (r \pm \frac{\sigma^2}{2})(T-\tau)}{\sigma\sqrt{T-\tau}},$$

where σ denotes the volatility. Clearly for $S \to \infty$ we have that $d_1 \to \infty$ and $d_2 \to \infty$ and because $N(\cdot)$ denotes the cumulative distribution function, $N(d_{1,2}) \to 1$, and so $u(S,\tau) \to (S - Ke^{-r(T-\tau)})$. The reason we use a Neumann boundary condition instead of this Dirichlet condition has to do with the volatility boundary. When we look at the behaviour of the option value in the Black Scholes case for $\sigma \to \infty$, we get: $d_1 \to \infty$ and $d_2 \to -\infty$, resulting in $N(d_1) \to 1$ and $N(d_2) \to 0$ such that $u(S,\tau) = S$. To avoid discontinuity at the point where the boundaries coincide, we choose to use the Neumann boundary condition at $s = S_{\max}$. To further substantiate these choices, a closed form solution of the Black Scholes formula is plotted for large values of σ and s in figures 4.1a and 4.1b.

4.1.1 Boundary at $v = 0$

Note that the last condition isn't really a boundary condition because it involves a time derivative. The Feller condition 2.15) determines the behaviour of the solution close to $v = 0$, and in both cases the option value satisfies this boundary condition. This can be treated as a PDE by itself. In the Heston and the SABR model the coefficients ensure that some terms vanish and the special boundary condition at $v = 0$ (or $\alpha = 0$ respectively) (4.1d) attains a more attractive form. In the Heston case:

$$\frac{\partial u}{\partial \tau}(s,0,\tau) = \kappa\eta\frac{\partial u}{\partial v}(s,0,\tau) + rs\frac{\partial u}{\partial s}(s,0,\tau) - ru(s,0,\tau), \tag{4.2}$$



4.1a B-S price for range of volatilities σ, clearly convergent to S_0. $K = 100, S_0 = 70$ and $r = 0.03$

4.1b B-S price for range of stock prices S_0, clearly $\frac{\partial U_{BS}}{\partial S} = 1$ for large values of S_0. $K = 100, \sigma = 0.1$ and $r = 0.03$

and in the SABR case, the condition simplifies to:

$$\frac{\partial u}{\partial \tau}(s,0,\tau) = rs\frac{\partial u}{\partial s}(s,0,\tau) - ru(s,0,\tau). \tag{4.3}$$

Note that although the artificial boundary condition at $v = 0$ for the Heston model and even more at $\alpha = 0$ for the SABR model look easy, an exact solution is not available. When there is an exact solution, one should apply this at the boundary. Another important observation is the lack of a second derivative at these boundaries. This implies that no one-sided finite difference approximation of the second derivative is needed.

4.2 Boundary conditions for an up-and-out call option

A more exotic option compared to the European call option studied above is the up-and-out call option. An up-and-out call option introduces a new exercise rule. In the case of an up-and-out barrier option, when the underlying stock price reaches a pre-set barrier price B the option is extinguished. The boundary condition at that pre-set level is not artificial, but stems from the product, so an up and out barrier option implies only one artificial boundary. Boundary condition (4.1c) must be replaced by:

$$u(B, v, t) = 0. \tag{4.4a}$$

To avoid conflict with other boundary conditions, also condition (4.1b) needs to be adjusted. Following [4] this boundary can be set equal to:

$$\frac{\partial u}{\partial v}(s, V_{\max}, t) = 0. \tag{4.4b}$$

The model now only consists of homogeneous boundary conditions. Next to that the boundary at $s = B = S_{\max}$ is part of the model and cannot be set far from the strike price K (e.g. $S_{\max} = 14K$) to lower the contribution of the error caused by this artificial boundary condition.

5 ADI method

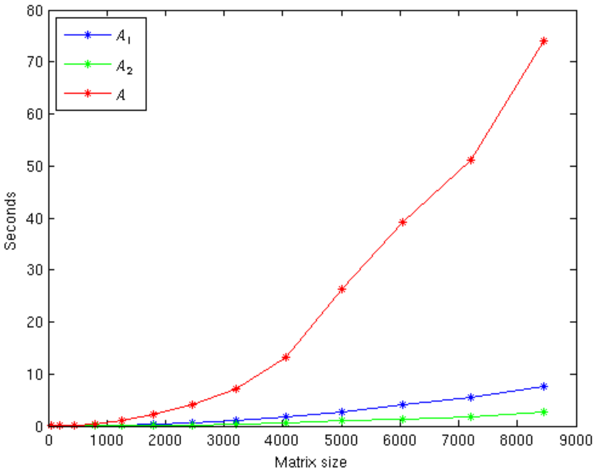
The popular standard θ method can be computationally very demanding. Increasing the number of grid points in space direction will give a more detailed result, but at the same time the system of equations that needs to be solved will increase as well. In the two dimensional case, increasing the number of grid points by 10 in both directions will increase the number of equations by a factor $10^2 = 100$. Because the θ-method involves implicit steps, the matrix in (3.20) needs to be inverted every time step, this can be very demanding depending on the structure of A. To optimize the structure of this matrix, the ADI method is introduced. ADI stands for Alternating Direction Implicit. From this name it is clear that the method treats one direction implicitly per step. The ADI method splits the matrix A:

$$A = A_0 + A_1 + A_2,$$

where A_0 corresponds to the mixed derivatives, A_1 to the derivatives in the s direction and A_2 to the derivatives in the v direction, the $ru_{i,j}$-term is distributed over A_1 and A_2. Similarly the vector R is split up:

$$R = R_0 + R_1 + R_2$$

again corresponding with the same directions as the matrices. In the scheme one time step is split up in sub steps. In every sub step one direction is treated implicitly and the other directions explicitly, in the next sub step, the next direction is treated implicitly and the other ones explicitly and so on. In every sub step only one direction matrix needs to be inverted, which has small bandwidth. When one only treats the matrices A_1 and A_2 implicitly, the implicit steps in this method can be done fast because these matrices are sparse and the non zero entries are close to the diagonal. See figure 5.0a, where the time to invert the matrices A, A_1 and A_2 are plotted against the number of grid points M. Note that one ADI time step consists of two sub steps in which A_1 and A_2 are inverted, so for a fair comparison the inverting times of A_1 and A_2 should be added. In theory, it can be chosen to treat every direction implicitly, but to enhance computational time it is better to choose only directions that result in sparse matrices. The mixed derivative matrix, is not sparse and is therefore not treated in an implicit fashion.



5.0a Matrix inversion of A, A_1 and $A2$ in seconds.

N.B. *The non zero entries of A_0 are far from the diagonal and that is why in all ADI schemes the A_0 part is treated in an explicit fashion.*

Several variations of ADI schemes have been introduced. The simplest ADI scheme is the Douglas scheme, discussed next.

5.1 Douglas scheme

The ADI method treats all space directions in an implicit and an explicit fashion. Other than in the θ-scheme, it creates intermediate solutions Y_i for $i = 1, \ldots, n$. To calculate these intermediate solutions, one direction is treated implicitly, and the others explicitly. Then to calculate the next intermediate solution, the next space direction is treated implicitly while all others are treated explicitly. This is proceeded until all directions have been treated implicitly once. For example in the two dimensional case, the starting point of the ADI scheme is the following fully explicit forward in time scheme:

$$\frac{U^{k+1} - U^k}{\Delta t} = AU^k + R.$$

Splitting the matrix in sub matrices: $A = A_0 + A_1 + A_2$ leads to:

$$\frac{U^{k+1} - U^k}{\Delta t} = (A_0 + A_1 + A_2)\, U^k + R. \tag{5.1}$$

When one chooses to treat one direction partly implicitly and partly explicitly, the following formula is used:

$$\frac{U^{k+1} - U^k}{\Delta t} = (A_0 + (1-\theta)A_1 + A_2)\, U^k + \theta A_1 U^{k+1} + R. \tag{5.2}$$

When all $k+1$ terms are assembled on the right hand side an all the k terms on the left hand side, this leads to:

$$(I - \theta\Delta t A_1)U^{k+1} = (I + \Delta t A_0 + (1-\theta)\Delta t A_1 + \Delta t A_2)\, U^k + \Delta t R, \tag{5.3a}$$

and treating the other direction partly implicitly leads to:

$$(I - \theta\Delta t A_2)U^{k+1} = (I + \Delta t A_0 + \Delta t A_1 + (1-\theta)\Delta t A_2)\, U^k + \Delta t R. \tag{5.3b}$$

What the ADI scheme actually does, is one fully explicit step. Then it does a correction step such that actually one direction is done implicitly and the rest still explicitly. In the next step the other direction is corrected implicitly. Therefore the intermediate steps are often referred to as correction steps. As stated the first step Y_0 is a fully explicit step:

$$Y_0 = (I + \Delta t A_0 + \Delta t A_1 + \Delta t A_2)\, U^k + \Delta t R. \tag{5.4}$$

Now, to correct the first direction implicitly, Y_1 is defined as follows:

$$\begin{aligned} Y_1 &= Y_0 - \theta\Delta t A_1 U^k + \theta\Delta t A_1 Y_1 \Rightarrow \\ (I - \theta\Delta t A_1)Y_1 &= Y_0 - \theta\Delta t A_1 U^k. \end{aligned} \tag{5.5a}$$

Inserting (5.4) in (5.5a) results in (5.3a). Hence, till now only one direction is corrected implicitly. The next step will correct the other direction implicitly by repeating the above procedure:

$$\begin{aligned} Y_2 &= Y_1 - \theta\Delta t A_2 U^k + \theta\Delta t A_2 Y_2 \Rightarrow \\ (I - \theta\Delta t A_2)Y_2 &= Y_1 - \theta\Delta t A_2 U^k. \end{aligned} \tag{5.5b}$$

Now both directions are corrected implicitly and the above derived sub steps form the Douglas method:

$$\begin{aligned} Y_0 &= U^k + \Delta t\left(AU^k + R\right), \\ Y_1 &= Y_0 + \theta\Delta t\left(A_1Y_1 - A_1U^k\right), \\ Y_2 &= Y_1 + \theta\Delta t\left(A_2Y_2 - A_2U^k\right), \\ U^{k+1} &= Y_2. \end{aligned} \tag{5.6}$$

The great advantage of doing these sub steps is the smaller bandwidth of the inverted matrices. Also these expensive matrix inversions can be done in advance and do not have to be repeated every time step. In the code belonging to this thesis, the method coded as follows:

$$\begin{aligned} Y_0 &= U^k + \Delta t\left(AU^k + R\right), \\ Y_1 &= (I - \theta\Delta tA_1)^{-1}\left(Y_0 - \theta\Delta tA_1U^k\right), \\ Y_2 &= (I - \theta\Delta tA_2)^{-1}\left(Y_1 - \theta\Delta tA_2U^k\right), \\ U^{k+1} &= Y_2. \end{aligned} \tag{5.7}$$

5.2 Convergence of the Douglas scheme

The ADI scheme actually is a partly implicit and explicit (IMEX) scheme . The main difference between this method and the earlier discussed θ-method is the sub steps. In this sub section it is shown that these sub steps approximate the normal single step θ-scheme up to $\mathcal{O}(\Delta t^2)$. For splitted matrices $A = A_0 + A_1 + A_2$ the θ-method comes down to:

$$\frac{U^{k+1} - U^k}{\Delta t} = (A_0 + (1-\theta)A_1 + (1-\theta)A_2)\,U^k + \theta A_1U^{k+1} + \theta A_2U^{k+1} + R \Rightarrow \tag{5.8}$$

$$U^{k+1} = U^k + \Delta tA_0U^k + (1-\theta)\Delta t(A_1 + A_2)U^k + \theta\Delta t(A_1 + A_2)U^{k+1} + \Delta tR. \tag{5.9}$$

Assembling all U^{k+1} at the left hand side and all U^k on the right hand side, this comes down to:

$$(I - \theta\Delta tA_1 - \theta\Delta tA_2)U^{k+1} = (I + \Delta tA_0 + (1-\theta)\Delta tA_1 + (1-\theta)\Delta tA_2)U^k + \Delta tR$$

The important observation is that the left hand side above almost equals the product $(I - \theta\Delta tA_1)(I - \theta\Delta A_2)U^{k+1}$. The only difference is a $\theta^2\Delta t^2A_1A_2U^{k+1}$ term of order Δt^2. Adding this term on both sides yields:

$$\begin{aligned} (I - \theta\Delta tA_1)(I - \theta\Delta tA_2)U^{k+1} =& (I + \Delta tA_0 + (1-\theta)\Delta tA_1 + (1-\theta)\Delta tA_2)U^k \\ &+ \theta^2\Delta t^2A_1A_2U^{k+1} + \Delta tR \\ =& (I + \Delta tA_0 + (1-\theta)\Delta tA_1 + (1-\theta)\Delta tA_2)U^k \\ &+ \Delta tR + \mathcal{O}(\Delta t^2). \end{aligned} \tag{5.10}$$

Taking a better look at the Douglas scheme (5.6), one realises that it only uses one sub step. Written in a more compact form, by eliminating Y_2 and Y_0 the scheme is denoted as:

$$(I - \theta\Delta tA_1)Y_1 = (I + \Delta tA_0 + (1-\theta)\Delta tA_1 + \Delta tA_2)U^k + \Delta tR, \tag{5.11a}$$

$$(I - \theta\Delta tA_2)U^{k+1} = Y_1 - \theta\Delta tA_2U^k. \tag{5.11b}$$

Now eliminating Y_1 in equation (5.11), by substituting (5.11b) in (5.11a) this yields:

$$\begin{aligned}(I-\theta\Delta tA_1)\left((I-\theta\Delta tA_2)U^{k+1}+\theta\Delta tA_2U^k\right)=&(I+\Delta tA_0+(1-\theta)\Delta tA_1+\Delta tA_2)U^k\\&+\Delta tR\end{aligned}$$

Again, assembling all $k+1$ terms to the left an all k terms to the right hand side, this is equivalent to:

$$\begin{aligned}(I-\theta\Delta tA_1)(I-\theta\Delta tA_2)U^{k+1}=&(I+\Delta tA_0+(1-\theta)\Delta tA_1+\Delta tA_2)U^k\\&-(I-\theta\Delta tA_1)\theta\Delta tA_2U^k+\Delta tR\\=&(I+\Delta tA_0+(1-\theta)\Delta tA_1+\Delta tA_2)U^k\\&-\theta\Delta tA_2U^k+\Delta tR+\mathcal{O}(\Delta t^2)\\=&(I+\Delta tA_0+(1-\theta)\Delta tA_1+(1-\theta)\Delta tA_2)U^k\\&+\Delta tR+\mathcal{O}(\Delta t^2).\end{aligned}$$

Concluding that this method indeed approximates the one step θ-scheme in (5.10). For $\theta=\frac{1}{2}$, the θ-scheme is of second order, and known as the Crank-Nicolson method.

5.3 The Craig-Sneyd scheme

In a similar way the Craig Sneyd scheme applies 5 intermediate solutions. After the first set of correction steps a new starting solution is used to apply the same procedure again:

$$\begin{aligned}Y_0&=U^k+\Delta t\left(AU^k+R\right),\\Y_1&=Y_0+\theta\Delta t\left(A_1Y_1-A_1U^k\right),\\Y_2&=Y_1+\theta\Delta t\left(A_2Y_2-A_2U^k\right),\\Y_3&=Y_0+\frac{1}{2}\Delta tA_0\left(Y_2-U^k\right),\\Y_4&=Y_3+\theta\Delta t\left(A_1Y_4-A_1U^k\right),\\Y_5&=Y_4+\theta\Delta t\left(A_2Y_5-A_2U^k\right),\\U^{k+1}&=Y_5.\end{aligned}\tag{5.12}$$

In addition the modified Craig-Sneyd scheme even further improves the new starting solution by adding an extra correction step:

$$\begin{aligned}
Y_0 &= U^k + \Delta t \left(AU^k + R \right), \\
Y_1 &= Y_0 + \theta \Delta t A_1 \left(Y_1 - U^k \right), \\
Y_2 &= Y_1 + \theta \Delta t A_2 \left(Y_2 - U^k \right), \\
Y_3 &= Y_0 + \theta \Delta t A_0 \left(Y_2 - U^k \right), \\
Y_4 &= Y_3 + (\frac{1}{2} - \theta) A \Delta t \left(Y_2 - U^k \right), \\
Y_5 &= Y_4 + \theta \Delta t A_1 \left(Y_5 - U^k \right), \\
Y_6 &= Y_5 + \theta \Delta t A_2 \left(Y_6 - U^k \right), \\
U^{k+1} &= Y_6.
\end{aligned}$$

5.4 The Hunsdorfer-Verwer scheme

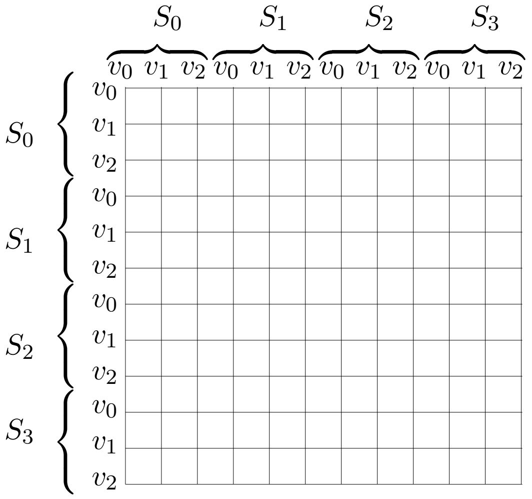
The last ADI scheme suggested in the in ‘t Hout paper is the Hunsdorfer-Verwer scheme:

$$\begin{aligned}
(I - \theta \Delta t A_1) Y_1 &= (I + \Delta t A_0 + (1-\theta)\Delta t A_1 + \Delta t A_2) U^k + \Delta t R, \\
(I - \theta \Delta t A_2) Y_2 &= Y_1 - \theta \Delta t A_2 U^k, \\
Y_3 &= (I - \theta \Delta t A_1) Y_1 + \theta \Delta t A_1 U^k \\
&\quad + \frac{1}{2} \Delta t A (Y_2 - U^k), \\
(I - \theta \Delta t A_1) Y_4 &= Y_3 - \theta \Delta t A_1 Y_2, \\
(I - \theta \Delta t A_2) U^{k+1} &= Y_4 - \theta \Delta t A_2 Y_3.
\end{aligned}$$

In the paper by Haentjens and in ’t Hout [4] this scheme is concluded to give the best results in terms of accuracy and order of convergence. This method is also applicable for a wide range of θ values. This is why, in this thesis, this scheme is applied most often.

5.5 Construction of the matrices

As mentioned before, in higher dimensions the solution vector needs to be ordered in a convenient way. This ordering fully determines the structure of the matrices. The lexicographic ordering results in matrices that are sparse and concentrated close to the diagonal. The space is discretized as in (3.2). The PDE (2.9) stemming from a stochastic volatility model is two dimensional, therefore the ADI scheme requires three matrices A_0, A_1 and A_2. The A_0-matrix stems from all mixed derivative terms, the A_1-matrix from derivative terms with respect to s and the A_2-matrix from derivative terms with respect to v. Next to that the $-rU$ term is evenly distributed over A_1 and A_2. All matrices have the following form (shown here for $m_1 = 3$ and $m_2 = 2$):



This PDE is a so-called two dimensional second order PDE, it consists of first and second derivatives w.r.t. s and v. To increase the robustness of the method, it is convenient to construct A_1 and A_2 from two distinct matrices corresponding to the first and second derivative:

$$A_1 = A_s + A_{ss} - \frac{1}{2}rI \text{ and } A_2 = A_v + A_{vv} - \frac{1}{2}rI,$$

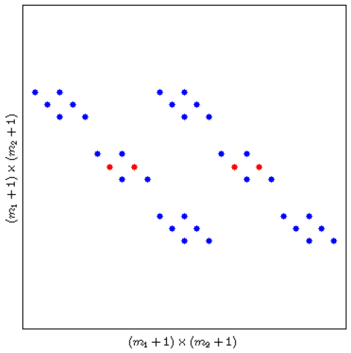
where the subscript indicates the first or second derivative. In the following paragraphs the construction of the specific matrices is treated in detail. Examples for a small number of grid points $m_1 = m_2 = 4$ are given on a uniform grid, but can be applied on a non-uniform grid in exactly the same way. For the clarity of the examples the coefficients are adopted from the Heston PDE (2.14) where $\mu_s = rs, \sigma_s = s\sqrt{v}, \mu_v = \kappa(\eta - v)$ and $\sigma_v = \sigma\sqrt{v}$, but these derivations are applicable in more general cases.

5.5.1 Constructing A_0

In the Heston PDE the coefficient in front of the $\frac{\partial^2}{\partial s \partial v}$-term ($\sigma_s \sigma_v \rho$) equals $\sigma s v \rho$, so for $s = 0$ or for $v = 0$ the mixed derivative is not needed. Furthermore, from the boundary conditions (4.1c) and (4.1b) one can conclude $\frac{\partial^2 u}{\partial s \partial v}(S_{\max}, v, t) = \frac{\partial^2 u}{\partial s \partial v}(s, V_{\max}, t) = 0$ for any s, v and t. The discrete approximation for the mixed derivative is:

$$\sigma s v \rho \frac{u_{i+1,j+1} + u_{i-1,j-1} - u_{i-1,j+1} - u_{i+1,j-1}}{4 \Delta v \Delta s},$$

so for the evaluation in one point, four terms are used. All the terms are far off-diagonal. Because of the lexicographic ordering the coefficient that belongs to $u_{i-1,j-1}$ is $m_2 + 1 + 1$ entries away from the diagonal. Due to this inconvenient property, this matrix is treated in an explicit fashion every iteration. Figure 5.5a displays the non-zero entries of the matrix $A_0 = A_{sv}$. The red dots denote coefficients belonging to the derivative of $u_{2,2}$.



5.5a A_0 consisting of $(m_1 + 1) = 5$ blocks of size $(m_2 + 1) = 5$

5.5.2 Constructing A_1

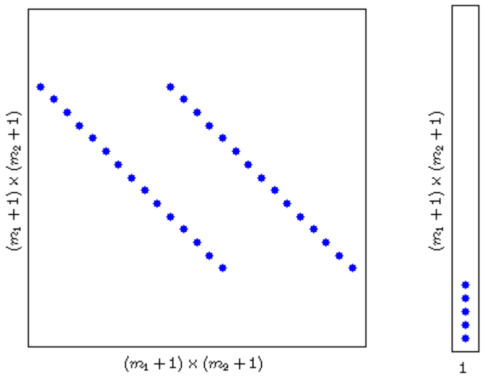
As mentioned, A_1 can be constructed from A_s and A_{ss}. For the first and the second derivative in s direction a central scheme is used. Because of our lexicographic ordering, the non-zero diagonals in the A_s and A_{ss} matrix will not be consecutive, but there will be exactly $(m_2 + 1)$ zeros between them. Beginning with the first derivative, the interior points are discretized as:

$$r s_i \frac{u_{i+1,j} - u_{i-1,j}}{2 \Delta s}.$$

Because of the Dirichlet boundary conditions at $s = 0$, no derivative is needed there. In the Heston call option case, at $s = S_{\max}$ the derivative is given by the Neumann boundary condition, such that the matrix A_s has no inputs at its last rows, and the known value is stored in the vector R_s. So the interior for A_s and R_s will be of the form:

$$A_s = \begin{pmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \frac{s_1}{2\Delta s} I & \mathbf{0} & \frac{-s_1}{2\Delta s} I & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \frac{s_2}{2\Delta s} I & \mathbf{0} & \frac{-s_2}{2\Delta s} I & 0 \\ \mathbf{0} & \mathbf{0} & \frac{s_3}{2\Delta s} I & \mathbf{0} & \frac{-s_3}{2\Delta s} I \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{pmatrix} \text{ and } R_s = \begin{pmatrix} \vec{0} \\ \vec{0} \\ \vec{0} \\ \vec{0} \\ \vec{I} \end{pmatrix},$$

where I denotes the $(m_2+1)\times(m_2+1)$ identity matrix, $\mathbf{0}$ denote $(m_2+1)\times(m_2+1)$ zero matrices, $\vec{0}$ $(m_2+1)\times 1$ zero vectors and $\vec{I}$ an $(m_2+1)\times 1$ vector. See figure 5.5b for an appealing graphic notation.



5.5b Matrix A_s and its corresponding vector R_s

For the second derivative with respect to s the structure will equal the matrix A_s. The interior points are discretized as:

$$\frac{1}{2}(\sigma_s)^2_{i,j}\frac{u_{i+1,j}-2u_{i,j}+u_{i-1,j}}{\Delta s^2}=\frac{1}{2}v_j s_i^2\frac{u_{i+1,j}-2u_{i,j}+u_{i-1,j}}{\Delta s^2}.$$

Again the block determined by the Dirichlet boundary condition at $s=0$ can be taken equal to zero. However, because of the Neumann boundary condition at $s=S_{\max}$, (4.1c), $\frac{\partial^2 u}{\partial s^2}$ still needs to be approximated. The first derivative is given and to use this information in the approximation of the second derivative a so called “ghost”- point at $s_{m_1+2,j}$ is introduced. The function value $u_{m_1+2,j}$ at this point can be linearly extrapolated using the boundary condition:

$$\begin{aligned}\frac{\partial u}{\partial s}&=\frac{u_{m_1+2,j}-u_{m_1,j}}{2\Delta s}=1\Leftrightarrow\\ u_{m_1+2,j}&=2\Delta s+u_{m_1,j}.\end{aligned}\tag{5.14}$$

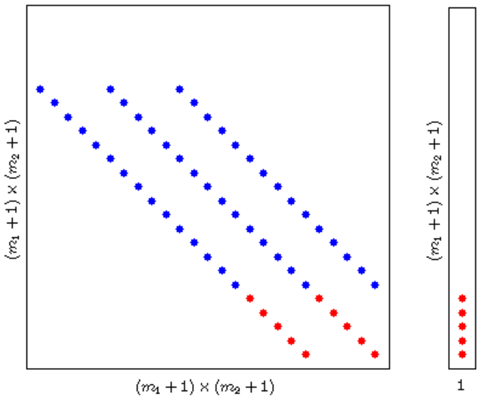
With the help of this obtained ghost point, the central scheme at s_{m_1} can be used by substituting (5.14) in (3.8c). Resulting in the following discretization:

$$\frac{1}{2}(\sigma_s)^2_{m_1+1,j}\frac{u_{m_1+2,j}-2u_{m_1+1,j}+u_{m_1,j}}{\Delta s^2}=\frac{1}{2}v_j s^2_{m_1+1}\frac{2\Delta s-2u_{m_1+1,j}+2u_{m_1,j}}{\Delta s^2}.$$

The factor stemming from the linear extrapolation needs to be stored in the vector R_{ss}. The matrix thus looks like:

$$A_{ss}=\begin{pmatrix}\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{0}\\ \frac{s_1^2}{2\Delta s^2}\mathbf{V}&\frac{-2s_1^2}{2\Delta s^2}\mathbf{V}&\frac{s_1^2}{2\Delta s^2}\mathbf{V}&\mathbf{0}&\mathbf{0}\\ \mathbf{0}&\frac{s_2^2}{2\Delta s^2}\mathbf{V}&\frac{-2s_2^2}{2\Delta s^2}\mathbf{V}&\frac{s_2^2}{2\Delta s^2}\mathbf{V}&\mathbf{0}\\ \mathbf{0}&\mathbf{0}&\frac{s_3^2}{2\Delta s^2}\mathbf{V}&\frac{-2s_3^2}{2\Delta s^2}\mathbf{V}&\frac{s_3^2}{2\Delta s^2}\mathbf{V}\\ \mathbf{0}&\mathbf{0}&\mathbf{0}&\frac{2s_4^2}{2\Delta s^2}\mathbf{V}&\frac{-2s_4^2}{2\Delta s^2}\mathbf{V}\end{pmatrix}\text{ and }R_{ss}=\begin{pmatrix}\vec{0}\\ \vec{0}\\ \vec{0}\\ \vec{0}\\ \vec{0}\\ \frac{2\Delta s}{2\Delta s^2}\vec{v}\end{pmatrix},$$

where $\vec{v}$ denotes the vector with grid points v_0 to v_{m_2} as entries and $\mathbf{V}$ denotes a one diagonal matrix with the vector $\vec{v}$ on its main diagonal. Figure 5.5c shows the non-zero entries of the matrix A_{ss} and vector R_{ss}. The red dots denote the entries related to the boundary condition.



5.5c Matrix A_{ss} and its corresponding vector R_{ss}

Because $A_1 = A_s + A_{ss} - \frac{1}{2}rI$ and $R_1 = R_s + R_{ss}$ eventually one has:

$$A_1 = \begin{pmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \frac{s_1^2}{2\Delta s^2}\mathbf{V} + \frac{s_1}{2\Delta s}I & \frac{-2s_1^2}{2\Delta s^2}\mathbf{V} - \frac{1}{2}rI & \frac{s_1^2}{2\Delta s^2}\mathbf{V} - \frac{s_1}{2\Delta s}I & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \frac{s_2^2}{2\Delta s^2}\mathbf{V} + \frac{s_2}{2\Delta s}I & \frac{-2s_2^2}{2\Delta s^2}\mathbf{V} - \frac{1}{2}rI & \frac{s_2^2}{2\Delta s^2}\mathbf{V} - \frac{s_2}{2\Delta s}I & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \frac{s_3^2}{2\Delta s^2}\mathbf{V} + \frac{s_3}{2\Delta s}I & \frac{-2s_3^2}{2\Delta s^2}\mathbf{V} - \frac{1}{2}rI & \frac{s_3^2}{2\Delta s^2}\mathbf{V} - \frac{s_3}{2\Delta s}I \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \frac{2s_4^2}{2\Delta s^2}\mathbf{V} & \frac{-2s_4^2}{2\Delta s^2}\mathbf{V} - \frac{1}{2}rI \end{pmatrix},$$

$$\text{and } R_1 = \begin{pmatrix} \vec{0} \\ \vec{0} \\ \vec{0} \\ \vec{0} \\ \vec{0} \\ \frac{2\Delta s}{2\Delta s^2}\vec{v} + \vec{I} \end{pmatrix}.$$

5.5.3 Constructing A_2

In the lexicographic ordering the v space is leading, so the matrices A_v and A_{vv} consist of m_1+1 blocks of matrices of size $(m_2+1)\times(m_2+1)$. First consider A_v consisting of terms from the first order v-derivatives. Unlike the previous examples the differences in v-direction use not only the central scheme. The coefficient $\mu_v = \kappa(\eta - v)$ can become negative. When this is the case the backward scheme is applied. Because also the forward scheme is needed at $v = 0$ a single block matrix, for some fixed s-grid point i, uses the following three finite differences:

- the central finite difference:

$$\kappa(\eta - v_j)\frac{u_{i,j+1} - u_{i,j-1}}{2\Delta v},$$

- the forward finite difference at $v = 0$ boundary:

$$\kappa(\eta - v_j)\frac{-3u_{i,0} + 4u_{i,1} - u_{i,2}}{2\Delta v},$$

- the backward finite difference at the $v > \eta$ region

$$\kappa(\eta - v_j)\frac{u_{i,j-2} - 4u_{i,j-1} + 3u_{i,j}}{2\Delta v}.$$

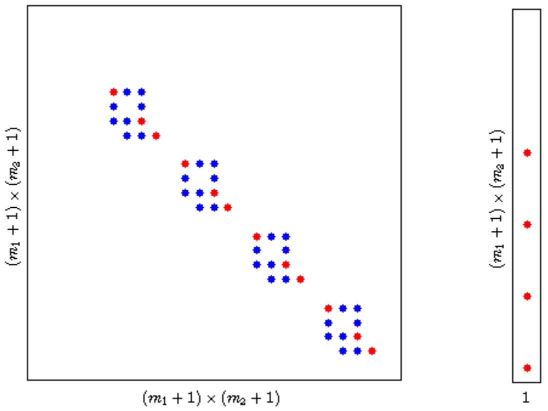
The Dirichlet condition at $s = 0$ (4.1a) and $v = V_{\max}$ (4.1b) are given such that no entries are needed at the lines corresponding to $s = s_0$ and $v = V_{\max}$. The non-homogeneous term in the boundary condition at $u(s, V_{\max}, \tau) = s$ is stored in the vector R_v. So the interior for A_v and R_v will be of the form:

$$A_v = \begin{pmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{B}_v & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{B}_v & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{B}_v & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{B}_v \end{pmatrix} \text{ and } R_v = \begin{pmatrix} \vec{0} \\ s_1\vec{e}_5 \\ s_2\vec{e}_5 \\ s_3\vec{e}_5 \\ s_4\vec{e}_5 \end{pmatrix},$$

where $\vec{e}_5$ denotes an $(m_2 + 1) \times 1$ vector with a 1 at index 5 and $\mathbf{B}_v$ an $(m_2 + 1) \times (m_2 + 1)$ matrix with the following structure:

$$B_v = \begin{pmatrix} \frac{\kappa(\eta - v_0)}{2\Delta v} & \frac{\kappa(\eta - v_0)}{2\Delta v} & \frac{\kappa(\eta - v_0)}{2\Delta v} & 0 & 0 \\ \frac{\kappa(\eta - v_1)}{2\Delta v} & 0 & \frac{\kappa(\eta - v_1)}{2\Delta v} & 0 & 0 \\ \frac{\kappa(\eta - v_2)}{2\Delta v} & \frac{\kappa(\eta - v_2)}{2\Delta v} & \frac{\kappa(\eta - v_2)}{2\Delta v} & 0 & 0 \\ 0 & \frac{\kappa(\eta - v_3)}{2\Delta v} & \frac{\kappa(\eta - v_3)}{2\Delta v} & \frac{\kappa(\eta - v_3)}{2\Delta v} & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \text{ and } s_1\vec{e}_5 = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ s_1 \end{pmatrix}.$$

Figures 5.5d show the structure of A_v and R_v in a more appealing way.



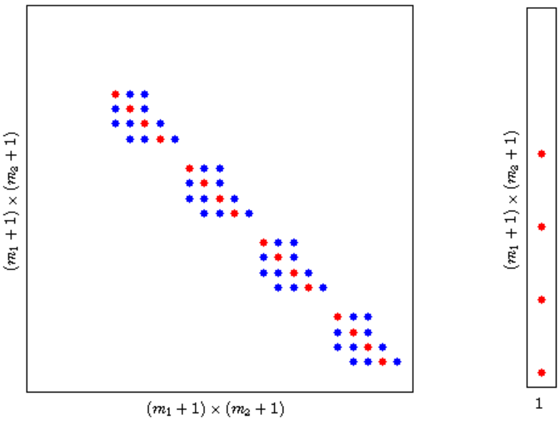
5.5d A_v and R_v

The red dots denote the diagonal to emphasize the backward, forward or central scheme used in the particular row. Because of the uniform scheme the central scheme only uses two points.

In this example v ranges from 0 to 1 in 4 steps, so $\Delta v = \frac{1}{4}$ and $v_i = \frac{i}{4}$. Because $\eta = 0.26$, the method uses the backward scheme from v_3 on. To substantiate the reason for using the backward scheme some numerical tests are shown in chapter 8. The second derivative with respect to v at $v = 0$ is not needed in the PDE because the coefficient of $\frac{\partial^2 u}{\partial v^2}$ ($\sigma_v^2 = \frac{1}{2}\sigma^2 v$) equals zero at $v = 0$. Next to that the second derivative at $v = V_{\max}$ is assumed to be equal to zero. This is reasonable because the price is assumed not to differ for extreme values of v. Because the Dirichlet boundary condition at $v = V_{\max}$ is already incorporated in R_s, the vector R_{vv} remains zero. Similar as in the A_v case, one has:

$$A_{vv} = \frac{\sigma^2}{2}\begin{pmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{B}_{vv} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{B}_{vv} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{B}_{vv} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{B}_{vv} \end{pmatrix} \text{ where } \mathbf{B}_{vv} = \frac{1}{\Delta v^2}\begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ v_1 & -2v_1 & v_1 & 0 & 0 \\ 0 & v_2 & -2v_2 & v_2 & 0 \\ 0 & 0 & v_3 & -2v_3 & v_3 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

Concluding this subsection, figure 5.5e schematically shows the resulting non-zero entries of $A_2 = A_v + A_{vv} - \frac{1}{2}rI$ and R_2.



5.5e A_2 consisting of 5 non-zero diagonals and R_2.

5.6 Boundary conditions for ADI

All ADI schemes use an intermediate solution. Logically the boundary conditions for the intermediate solution are closely related to the boundary conditions for the normal solution. Especially when the boundary conditions are independent of time the boundary conditions are similar. The PDEs studied in this thesis al have time independent boundary conditions. For this case one thus can bluntly apply the same boundary conditions for the intermediate solution:

N.B. *For a PDE with time independent boundary conditions, the intermediate solutions satisfy the boundary conditions from the original PDE.*

For completeness, this section shows what can be done when boundary conditions are time dependent. To illustrate the procedure the Douglas scheme is considered (5.6). Assume the normal heat equation with Neumann, Dirichlet and a time dependent boundary condition:

$$
\begin{aligned}
&\tfrac{\partial u}{\partial t} = \tfrac{\partial^2 u}{\partial x^2} + \tfrac{\partial^2 u}{\partial y^2},\\
&u(0,y,t) = 0, \qquad u(x,0,t) = 0, \qquad u(x,y,0) = f(x,y),\\
&u_x(1,y,t) = 1, \quad u(x,1,t) = h(t).
\end{aligned}
$$

The ADI method makes use of two finite difference approximations δ_x^2 and δ_y^2. As seen in section 3, Neumann conditions are always set with the help of a vector, and no finite difference approximations are needed there. These Neumann conditions are independent of time, so they can be adopted by these intermediate solutions. In the case of the time dependent condition at $y = 1$, more care is needed. In discrete notation at one grid point the Finite Difference scheme looks as follows:

$$\delta_t[u_{i,j}^k] = \delta_x^2[u_{i,j}^k] + \delta_y^2[u_{i,j}^k],$$

and the Douglas scheme with intermediate solutions w_0, w_1 and w_2 is represented as:

$$
\begin{aligned}
(w_0)_{i,j} &= u^k + \Delta t\left(\delta_x^2[u_{i,j}^k] + \delta_y^2[u_{i,j}^k]\right),\\
(w_1)_{i,j} &= (w_0)_{i,j} + \theta\Delta t\left(\delta_x^2[(w_1)_{i,j}] - \delta_x^2[u_{i,j}^k]\right),\\
(w_2)_{i,j} &= (w_1)_{i,j} + \theta\Delta t\left(\delta_y^2[(w_2)_{i,j}] - \delta_y^2[u_{i,j}^k]\right),\\
u_{i,j}^{k+1} &= (w_2)_{i,j}\,.
\end{aligned}
\tag{5.15}
$$

In this case only one intermediate solution is of interest, because for the intermediate solution $(w_0)_{i,j}$ and $(w_2)_{i,j}$, the boundary conditions are directly adopted from $u_{i,j}^k$ or $u_{i,j}^{k+1}$ respectively. Rewriting (5.15) in terms of $u_{i,j}^k$, $u_{i,j}^{k+1}$ and $(w_1)_{i,j}$, yields:

$$
\begin{aligned}
\left(1 - \theta\Delta t\delta_x^2\right)(w_1)_{i,j} &= u^k + \Delta t\left((1-\theta)\delta_x^2[u_{i,j}^k] + \delta_y^2[u_{i,j}^k]\right),\\
\left(1 - \theta\Delta t\delta_y^2\right)u_{i,j}^{k+1} &= (w_1)_{i,j} - \theta\Delta t\delta_y^2[u_{i,j}^k].
\end{aligned}
$$

From these equations all intermediate boundary conditions can be derived. By rewriting one obtains:

$$\theta\Delta t\delta_x^2\,(w_1)_{i,j} = \left(1 - \theta\Delta t\delta_y^2\right)u_{i,j}^{k+1} - u_{i,j}^k - \Delta t\left((1-\theta)\delta_x^2[u_{i,j}^k] + (1-\theta)\delta_y^2[u_{i,j}^k]\right).$$

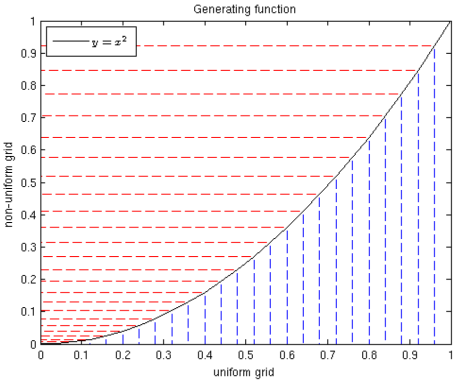
The ADI scheme copies the Dirichlet boundary conditions as an average of the boundary conditions at time level k and $k+1$. These boundary conditions are for example implied with the help of a ghost point described in section 3. This procedure can be extended to more general cases where the boundary conditions can be more sophisticated. For a detailed description see [9]. This method can be applied for more intermediate solutions.

6 Extension of the FDM

An important improvement of FDM, is applying a non-uniform grid. A non-uniform grid can be created to be more fine (or coarse) in a particular region. This section closely follows the derivation done by Kluge [5]. To construct non-uniform grids, one usually uses a grid generating function g, defined on the domain that needs to be adjusted. This function needs to be continuously differentiable, bijective and monotonically increasing. Let our domain be $\Omega = [0,1]$ and let $0 = x_0 < x_1 < \ldots < x_n = 1$ be its uniform discretization. Using $g : [0,1] \to [0,1]$ the non-uniform grid can be defined as:

$$y_i := cg(x_i) + d \text{ for } i = 1, 2, \ldots, n,$$

where c and d are scaling parameters. Below an example for $g(x_i) := x_i^2$ is shown: Clearly there are many more grid points in the neighbourhood of 0 in the non-uniform case then in the uniform case. A more logical approach to develop a grid generating function is to first look at the ratio between two adjacent points in the non-uniform grid and the uniform grid:



$$y_{i+1} - y_i = cg(x_{i+1}) - cg(x_i) \approx cg'(x_i)\Delta x = cg'(g^{-1}(y_i))\Delta x.$$

Using this approximation, the distance ratio function is defined as:

$$r(y) := g'(g^{-1}(y)) \text{ and so } g'(x) = r(g(x)).$$

This ODE can simply be solved using the separation of variables technique:

$$\int_0^x \frac{g'(z)}{r(g(z))} dz = \int_0^x dz,$$
$$\int_0^{g(x)} \frac{1}{r(y)} dy = x. \tag{6.1}$$

With the help of this distance ratio function the generating function can be calculated implicitly. Also we derive the necessary condition: $\int_0^1 \frac{1}{r(y)} = 1$, because $g(1) = 1$.

6.1 European Call option grid for the Heston PDE

The Heston PDE consists of an s-grid referring to the stock price and a v-grid referring to the volatility. Both these directions have particular characteristics which imply a grid refinement in a particular region. In the following two paragraphs a motivation is given for a specific grid, in s-direction and in v-direction.

6.1.1 Grid refinement in s-direction

When a European call option is considered, an increase of density needs to be imposed at the strike price K. A reasonable density function could be:

$$r(y) := \sqrt{c^2 + p^2(K-y)^2}.$$

At $y = K$ the distance between to adjacent points is approximately c. Moving away from K the distances increase. The function gets close to linear when y is getting large: $r(y) \approx |py|$. The scalar p is used to scale the function such that it satisfies $\int_0^1 \frac{1}{r(y)} = 1$. Substituting this distance ratio function into the expression derived above yields:

$$\int_0^{g(x)} \frac{1}{\sqrt{c^2 + p^2(K-y)^2}} dy = x.$$

First one can substitute $z = p(y-K)$, such that $\frac{dz}{dy} = p \Rightarrow dy = \frac{1}{p} dz$ and the indefinite integral becomes:

$$\frac{1}{p} \int \frac{1}{\sqrt{c^2 + z^2}} dz.$$

Next substituting the hyperbolic function $z = c \sinh\theta$ and $\frac{dz}{d\theta} = c\cosh\theta \Rightarrow dz = c\cosh\theta d\theta$ and, using $1 = \cosh^2\theta - \sinh^2\theta$, this yields:

$$\begin{aligned}
\frac{1}{p} \int \frac{1}{\sqrt{c^2 + z^2}} dz &= \frac{1}{p} \int \frac{1}{\sqrt{c^2\left(1 + \sinh^2\theta\right)}} c\cosh\theta d\theta \\
&= \frac{1}{p} \int \frac{1}{c\cosh\theta} c\cosh\theta d\theta = \frac{1}{p} \int 1 d\theta = \frac{1}{p}\theta \\
&= \frac{1}{p} \sinh^{-1} \frac{z}{c} \\
&= \frac{1}{p} \sinh^{-1} \frac{p(y-K)}{c}.
\end{aligned}$$

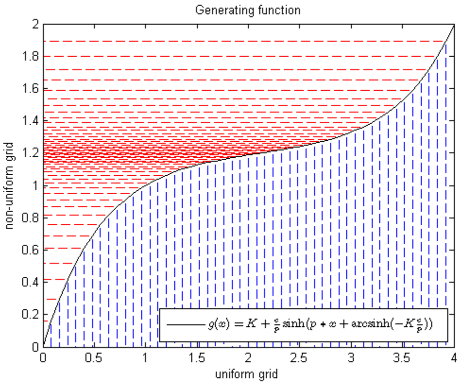
In explicit form, from (6.1) one concludes:

$$\begin{aligned}
&\int_0^{g(x)} \frac{1}{\sqrt{c^2 + p^2(K-y)^2}} dy = x \\
\Rightarrow &\frac{1}{p} \sinh^{-1}\left(\frac{p(g(x)-K)}{c}\right) - \frac{1}{p} \sinh^{-1}\left(\frac{-pK}{c}\right) = x \\
&\Rightarrow g(x) = K + \frac{c}{p} \sinh\left(px + \sinh^{-1}\left(\frac{-pK}{c}\right)\right). \qquad (6.2)
\end{aligned}$$

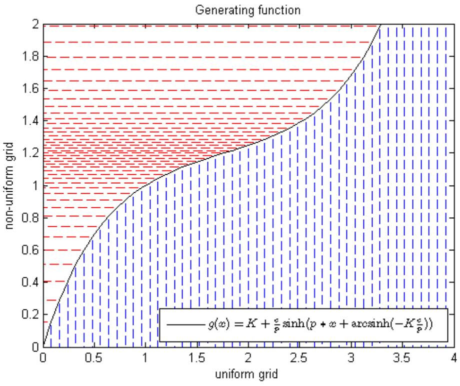
The scalar p can be calculated with the help of the condition $g(1) = 1$. This can be done by simply applying Newton-Raphson iteration to (6.2). Two examples for $c = 0.1$ and $c = 0.2$ ($K = 1.2$) are shown in figure 6.1a and 6.1b

The value of c behaves as a scaling parameter of the density of the non-uniform grid near K. Now to apply the non-uniform grid, with generating function $g(\xi) = K + \frac{c}{p} \sinh\xi$, first equidistant points $\xi_{\min} = \xi_0 < \xi_1 < \ldots < \xi_{m_1} = \xi_{\max}$ are defined by:

$$\begin{aligned}
\xi_i &= \sinh^{-1}\left(\frac{-pK}{c}\right) + i\Delta\xi, \text{ where} \\
\Delta\xi &= \frac{1}{m_1}\left(\sinh^{-1}\left((S_{\max} - K)\frac{p}{c}\right) - \sinh^{-1}\left(\frac{-pK}{c}\right)\right).
\end{aligned}$$



6.1a Non-uniform grid for $c = 0.1$, $p = 8.4216$



6.1b Non-uniform grid for $c = 0.2$, $p = -1.718$

By construction $g(0) = 0$ and $g(\xi_{\max}) = S_{\max}$ and the non-uniform grid $0 = s_0 < s_1 < \ldots < s_{m_1} = S_{\max}$ can be constructed via g:

$$s_i = g(\xi_i) = K + \frac{c}{p}\sinh(\xi_i)\,.$$

The non-uniform grid in the paper of K.J. in 't Hout and T. Haentjens [4], is somewhat adjusted. Instead of an increased density at a point, an interval $[S_{\text{left}}, S_{\text{right}}]$ containing K is introduced in which the intervals are uniform. Following the paper, one can choose:

$$S_{\text{left}} = \max\{\frac{1}{2}, e^{-rT}\}K \text{ and } S_{\text{right}} = K.$$

Outside this interval the function follows the generating function described in the previous example. Therefore the following parameters are introduced:

$$\begin{aligned}
\xi_{\min} = \xi_0 &= \sinh^{-1}\left(\frac{-pS_{\text{left}}}{c}\right),\\
\xi_{\text{int}} &= \frac{p(S_{\text{right}} - S_{\text{left}})}{c},\\
\xi_{\max} = \xi_{m1} &= \xi_{\text{int}} + \sinh^{-1}\left(\frac{p(S_{\max} - S_{\text{right}})}{c}\right).
\end{aligned}$$

Clearly the following holds: $\xi_{\min} < 0 < \xi_{\text{int}} < \xi_{\max}$. Now the non-uniform grid $0 = s_0 < s_1 < \ldots < s_{m_1} = S_{\max}$ can be constructed again via a uniform grid of $m_1 + 1$ points between $\xi_{\min}$ and $\xi_{\max}$: $\xi_{\min} = \xi_0 < \xi_1 < \ldots < \xi_{m_1} = \xi_{\max}$ and the function g:

$$\begin{aligned}
g(\xi_i) &=: s_i \quad i = 0, \ldots, m_1, \text{ where}\\
g(\xi_i) &= \begin{cases} S_{\text{left}} + \frac{c}{p}\sinh(\xi_i) & \text{if } \xi_{\min} \le \xi_i < 0,\\ S_{\text{left}} + \frac{c}{p}\xi_i & \text{if } 0 \le \xi_i \le \xi_{\text{int}},\\ S_{\text{right}} + \frac{c}{p}\sinh(\xi_i - \xi_{\text{int}}) & \text{if } \xi_{\text{int}} < \xi_i \le \xi_{\max}. \end{cases}
\end{aligned}$$

Although p is a parameter depending on the choice of the scaling parameter c, in the end only the ratio ($d_1 := \frac{c}{p}$) enters the formula. Experiments show that one can use this ratio d_1 as a scaling parameter, where smaller values of d_1 result in a higher density in $[S_{\text{left}}, S_{\text{right}}]$. Figure 6.1d shows how s is divided over the grid choosing $m_1 = 40$, $K = 100$ and $d_1 = \frac{K}{20} = 5$. In the European call option case the refinement around strike gives better results then when a uniform grid is used. In the case of an up-and-out call option, the initial condition is not only discontinuous at strike price K, but also at pre-set barrier level $B := S_{\text{max}}$, because the option directly loses its value when it reaches this level. Because of this, one can expect that when the stock price is between strike price K and barrier value B, the option price will reach its maximum. Motivated by this, one can choose to extend the interval, in which the differences are densely uniform, to include the entire interval $[K, B]$. For example

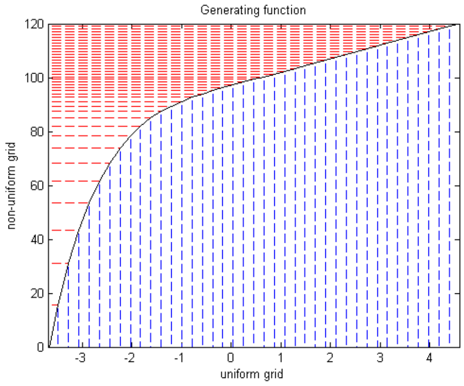
$$S_{\text{left}} = \max\{\frac{3}{4}, e^{-rT}\}K \text{ and } S_{\text{right}} = B.$$

Similar as in the European case we construct the values ξ_{min} and ξ_{max} as:

$$\xi_{\text{min}} = \xi_0 = \sinh^{-1}\left(\frac{-pS_{\text{left}}}{c}\right),$$
$$\xi_{\text{max}} = \frac{p(B - S_{\text{left}})}{c}.$$

Again it holds that $\xi_{\text{min}} < 0 < \xi_{\text{max}}$ and a uniform grid of $m_1 + 1$ points between ξ_{min} and ξ_{max} can be constructed. Using these grid points together with a function g results in:

$$g(\xi_i) =: s_i \quad i = 0, \ldots, m_1, \text{ where}$$
$$g(\xi_i) = \begin{cases} S_{\text{left}} + \frac{c}{p}\sinh(\xi_i) & \text{if } \xi_{\text{min}} \le \xi_i < 0, \\ S_{\text{left}} + \frac{c}{p}\xi_i & \text{if } 0 \le \xi_i \le \xi_{\text{max}}. \end{cases}$$

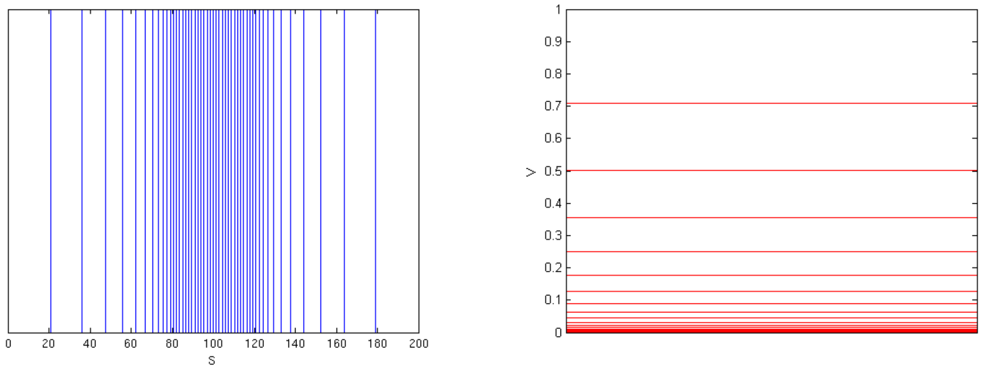


6.1c Non-uniform grid for up-and-out Barriers in s direction

6.1.2 Grid refinement in v-direction

For the v direction more points at the boundary $v = 0$ are desired and for larger values of v the mesh can be less dense. The European and up-and-out pay-off do not make any difference. Let m_2 be the number of points to be considered and d_2 a scaling parameter. Define equidistant points $\eta_0 < \eta_1 < \ldots < \eta_{m_2}$, given by $\eta_j = j \cdot \Delta\eta$, with $\Delta\eta = \frac{1}{m_2}\sinh^{-1}\left(\frac{V_{\text{max}}}{d_2}\right)$, for $j = 0, \ldots, m_2$. Now the mesh $0 = v_0 < v_1 < \ldots < v_{m_2} = V_{\text{max}}$ is defined by: $v_j = d_2 \sinh(\eta_j) \quad j = 0, \ldots, m_2$ and the resulting grid is shown in figure 6.1e. These two meshes are smooth in the sense that there are real constants C_0, C_1 and C_2 such that:

$$C_0\Delta\xi \le \Delta s_i \le C_1\Delta\xi \text{ and } |\Delta s_{i+1} - \Delta s_i| \le C_2(\Delta\xi)^2.$$



6.1d Non-uniform grid in s direction

6.1e Non-uniform grid in v direction

6.2 Grid shifting

A drawback when pricing options with FDM on a refined grid is that the specific option price for a predefined combination of stock price and volatility (s_0, v_0) is not a grid point in the resulting non-uniform grid. Especially when one wants to compare the FDM result with an obtained result by Monte Carlo solution, only one option price for a specific stock price and volatility is needed. A first solution is to bilinear interpolate the option price with the help of grid points closest by (s_0, v_0). See appendix A.5 for the derivation. Because this operation is only first order accurate, a loss of accuracy is expected. A better idea is to shift the obtained grid in such a way that the set of grid points in the shifted non-uniform grid contains (s_0, v_0). This method is known as grid shifting, and is treated in [14]. In the non-uniform grid in v-direction discussed above, the value of $V_{\max}$ can be increased with value y_v such that the new grid contains v_0. The boundary condition at $V_{\max}$ is artificial, so without any harm it can be set at $V_{\max} + y_v$, with $y_v > 0$, instead. In this case $V_{\max}$ holds as a lower limit for the value of the used boundary. For the s-grid a similar procedure can be applied. In section 6.1.1 the s-grid is refined such that it has an equidistant subinterval $[S_{\text{left}}, S_{\text{right}}]$ containing K. In this case, S_{right} can be adjusted with scaling parameter y_s. In this thesis, only grid shifting in v direction is applied.

6.3 Smoothing

Because the initial condition derived from the type of option considered is a discontinuous function, high frequency errors are expected around these discontinuities (e.g. around strike K and/or barrier B). The problems are most visible in the computation of the Greeks. For more details see [14] and [4]. A popular solution to this phenomenon is Rannacher time stepping. This method is introduced in combination with the Crank-Nicolson scheme (θ scheme for $\theta = \frac{1}{2}$), but can applied in a more general sense. The method implies two fully implicit steps at $\tau = 0$ before the ADI scheme is applied for all other time steps. This first initial step has a damping effect. Because of computational reasons, in this thesis, (similar as in [4]), it is chosen to apply Douglas scheme for $\theta = 1$ at $\tau = 0$. From $\tau = \Delta\tau$ on the intended ADI scheme with chosen θ is applied. So instead of two fully implicit steps, two fully implicit *sub steps* are taken. The damping effect comes from the implicitness of the time steps, so it is similar when the implicit sub steps are applied.

7 Analysis of FDM

This section has as an objective to find necessary and sufficient conditions for a given finite difference scheme to be a "good" approximation to a PDE. To be defined as "good", three criteria must be satisfied, namely: stability, convergence and consistency. Simply speaking, stability requires that the finite difference solution does not exponentially grow per time step. Convergence requires the solution to converge to a solution and consistency means that the solution to which it converges actually is the desired solution. A finite difference scheme is called consistent if the discretization considered actually approaches the solution. The most common approach to convergence is via consistency and stability in combination with the Lax Theorem. This theorem allows one to prove convergence of a scheme via stability and consistency, which are usually easier to show. In the following subsections these three properties will be treated individually. To this end, the spatial terms in a partial differential equation is viewed as an operator $\mathbf{L}(\cdot)$, such that (2.10) becomes:

$$\begin{aligned}\frac{\partial u}{\partial \tau} &= \mathbf{L}(u), \\ u(0,s,v) &= \phi(s,v).\end{aligned} \tag{7.1}$$

From section 3.2, the discrete approximation of (7.1) in a forward fashion states:

$$\begin{aligned}\frac{u_{i,j}^{k+1}-u_{i,j}^{k}}{\Delta\tau} &= \mathcal{D}_{\Delta s,\Delta v}(u_{i,j}^{k}), \\ u_{i,j}^{0} &= \phi_{i,j}.\end{aligned} \tag{7.2}$$

7.1 Consistency

As mentioned consistency makes sure the FDM approximation actually approaches the desired solution of the PDE. In a formal definition:

Definition 7.1. The finite difference scheme (7.2) is point wise consistent with the partial differential equation (7.1) if for any function $u(\tau,s,v)$ the following relationship holds:

$$\left(\frac{\partial}{\partial\tau}-\mathbf{L}\right)u(\tau_k,s_i,v_j)-\left[\frac{u_{i,j}^{k+1}-u_{i,j}^{k}}{\Delta\tau}-\mathcal{D}_{\Delta s,\Delta v}^{\Delta\tau}(u_{i,j}^{k})\right]\to 0 \text{ as } \Delta s,\Delta v,\Delta\tau\to 0. \tag{7.3}$$

To show that a finite difference method is stable all comes down on the use of Taylor series and its truncation errors. All earlier defined finite difference schemes from section 3 ((3.17) first order in time and (3.11) all second order in space) are derived via Taylor series. This allows one to derive the desired consistency by substituting the truncation errors in (7.3):

$$\begin{aligned}&\left(\frac{\partial}{\partial\tau}-\mathbf{L}\right)u(\tau_k,s_i,v_j)-\left[\frac{u_{i,j}^{k+1}-u_{i,j}^{k}}{\Delta\tau}-\mathcal{D}_{\Delta s,\Delta v}^{\Delta\tau}(u_{i,j}^{k})\right] \\ &= \mathcal{O}(\Delta\tau)+\mathcal{O}(\Delta s^2)+\mathcal{O}(\Delta v^2)=\mathcal{O}(\Delta\tau,\Delta s^2,\Delta v^2)\to 0 \text{ for } \Delta\tau,\Delta s,\Delta v\to 0.\end{aligned}$$

7.2 Stability and oscillations

As seen in the previous subsection, discretizing space gives rise to a numerical round off error. This local error is made at every time step and logically one would not want this error to increase in every time step. Finite difference methods can be:

1. Neutrally stable: error does not increase or decrease in doing one time step.
2. Stable: the error decreases in doing one time step.
3. Unstable: the error increases in doing one time step.

A way of checking the stability character of the method is given by Von Neumann analysis. As seen in section 3, to approximate derivatives, one can choose between a one sided approach and a central approach. The size and sign of the coefficients that belong to the derivative play an important role. When one applies the central scheme everywhere instability can cause spurious oscillations. These stem from the discretization of so-called unsteady convection-diffusion equations. These PDEs consist of first and second derivatives with respect to space and a derivative with respect to time. Possible instabilities in these PDEs are caused by individual partial derivative terms in the PDE. In this section an example that is one dimensional in space is studied that shows problems arising from these individual terms. The PDEs that are derived from stochastic volatility models, as seen in section 2, have two dimensions in space. The stabillity problems in this higher dimensional case directly stem from this more insightful one dimensional example.

Example 7.2. The next example stems from a popular physical phenomenon, namely a species concentration that is passively transported (convected and diffused) by flowing water. This species will be carried along by the flow of the water, and at the same time will diffuse. Mathematically the concentration u of the species satisfies the following PDE:

$$\frac{\partial u}{\partial t} = a\frac{\partial u}{\partial x} + b\frac{\partial^2 u}{\partial x^2},$$

where a is the flow speed of the water and b the coefficient for the diffusion of the species in the water. Discretizing this equation using a forward method in time and a central method in space (FTCS) results in:

$$\begin{aligned}\frac{u_j^{k+1} - u_j^k}{\Delta t} &= a\frac{u_{j+1}^k - u_{j-1}^k}{2\Delta x} + b\frac{u_{j+1}^k - 2u_j^k + u_{j-1}^k}{\Delta x^2},\\ u_j^{k+1} &= u_j^k + \frac{a\Delta t}{2\Delta x}(u_{j+1}^k - u_{j-1}^k) + \frac{b\Delta t}{\Delta x^2}(u_{j+1}^k - 2u_j^k + u_{j-1}^k),\\ u_j^{k+1} &= u_{j-1}^k(-\frac{a\Delta t}{2\Delta x} + \frac{b\Delta t}{\Delta x^2}) + u_j^k(1 - 2\frac{b\Delta t}{\Delta x^2}) + u_{j+1}^k(\frac{a\Delta t}{2\Delta x} + \frac{b\Delta t}{\Delta x^2}). \qquad (7.4)\end{aligned}$$

The stability of the scheme can be analysed by the Fourier transform. Let $(\ldots, u_{-1}, u_0, u_1, \ldots)^T$ be an infinite sequence of values. Then the discrete Fourier transform (given in [9]) is defined as:

$$\hat{u}(\xi) = \frac{1}{\sqrt{2\pi}} \sum_{j=-\infty}^{\infty} e^{-\mathrm{i}j\xi} u_j, \qquad \text{for } \xi \in [-\pi, \pi].$$

For notational convenience and following [9] and [10], let $R = \frac{a\Delta t}{\Delta x}$ and $r = \frac{b\Delta t}{\Delta x^2}$. By a simple change of variable one has:

$$\frac{1}{\sqrt{2\pi}} \sum_{j=-\infty}^{\infty} e^{-\mathrm{i}j\xi} u_{j\pm1} = \frac{e^{\pm\mathrm{i}\xi}}{\sqrt{2\pi}} \sum_{m=-\infty}^{\infty} e^{-im\xi} u_m = e^{\pm i\xi} \hat{u}^k(\xi). \qquad (7.5)$$

Substituting (7.5) in (7.4) and expressing the finite difference in discrete Fourier transform, gives us:

$$\begin{aligned}\hat{u}^{k+1}(\xi) &= (r-\frac{R}{2})e^{-\mathrm{i}\xi}\hat{u}^k(\xi) + (1-2r)e^{-\mathrm{i}\xi}\hat{u}^k(\xi) + (r+\frac{R}{2})e^{\mathrm{i}\xi}\hat{u}^k(\xi),\\ &= ((r-\frac{R}{2})e^{-\mathrm{i}\xi} + (1-2r) + (r+\frac{R}{2})e^{\mathrm{i}\xi})\hat{u}^k(\xi),\\ &:= \rho(\xi)\hat{u}^k(\xi).\end{aligned}$$

This $\rho(\xi)$ is called the symbol of the difference scheme. A direct intuition gives an interpretation to this symbol as an amplification factor. To prevent the solution to blow up in time one would want this factor to be smaller than or equal to one. This will give a stable result. This argument is confirmed by [9] and [10], so $|\rho(\xi)| \leq 1$ is required. By definition $e^{\mathrm{i}\xi} = \cos\xi + i\sin\xi$, such that a more convenient expression for $\rho(\xi)$ can be derived:

$$\rho(\xi) = 2r\cos\xi + iR\sin\xi + 1 - 2r.$$

To bound the magnitude of this complex function, $|\rho(\xi)|^2$ is considered. To determine the maximum and minimum, this function can be derived with respect to ξ and set equal to zero:

$$\begin{aligned}&\frac{\partial}{\partial\xi}\left((2r\cos\xi + (1-2r))^2 + R^2\sin^2\xi\right)\\ &= \frac{\partial}{\partial\xi}\left(4r^2\cos^2\xi + 4r(1-2r)\cos\xi + (1-2r)^2 + R^2(1-\cos^2\xi)\right)\\ &= -8r^2\cos\xi\sin\xi - 4r(1-2r)\sin\xi + 2R^2\cos\xi\sin\xi = 0.\\ &\Rightarrow \xi = 0, \xi = \pm\pi \text{ and } \cos\xi = \frac{2r(1-2r)}{R^2-4r^2}.\end{aligned}$$

For $\xi = 0$ the modulus of ρ equals 1. This is why no restriction on R and r can be derived, for $\xi = \pm\pi$ the modulus equals:

$$|\rho(\pm\pi)|^2 = (-2r + (1-2r))^2 = (1-4r)^2.$$

So $r \leq \frac{1}{2}$ suffices to meet the bound of the magnitude. For the last potential maximum, there are two cases.
Case 1: $R^2 \leq 4r^2$
Using the earlier derived inequality $r \leq \frac{1}{2}$, the fraction can be bounded: $\frac{2r(1-2r)}{R^2-4r^2} = \frac{2r-4r^2}{R^2-4r^2} \geq 1 \Rightarrow r > \frac{1}{2}$ which violates the assumption $r \leq \frac{1}{2}$, so:

$$\begin{aligned}|\rho|^2 &= 4r^2\left(\frac{2r(1-2r)}{R^2-4r^2}\right)^2 + 4r(1-2r)\frac{2r(1-2r)}{R^2-4r^2} + (1-2r)^2 + R^2\left(1-\left(\frac{2r(1-2r)}{R^2-4r^2}\right)^2\right)\\ &\leq 4r^2 + 4r(1-2r) + (1-2r)^2 = 4r^2 + 4r - 8r^2 + 1 - 4r + 4r^2 = 1.\end{aligned}$$

So no further restrictions are derived and the first stability condition equals $\frac{|R|}{2} \leq r \leq \frac{1}{2}$.
Case 2: $4r^2 < R^2$
First, let $\frac{2r(1-2r)}{R^2-4r^2} \geq 1$. This inequality is equivalent to $r \geq \frac{R^2}{2}$, so the final stability condition in this case reads: $\frac{R^2}{2} \leq r \leq \frac{|R|}{2}$. Now let $\frac{2r(1-2r)}{R^2-4r^2} < 1$, parallel to the other case this inequality is

equivalent to $r < \frac{R^2}{2}$. Substituting $\cos\xi = \frac{2r(1-2r)}{R^2-4r^2}$ into $|\rho(\xi)|^2$ leads to:

$$\begin{aligned}
|\rho(\xi)|^2 &= (1-2r)^2 + R^2 + 4r(1-2r)\left(\frac{2r(1-2r)}{R^2-4r^2}\right) + (4r^2-R^2)\left(\frac{2r(1-2r)}{R^2-4r^2}\right)^2 \\
&= \frac{(1-2r)^2(R^2-4r^2+4r^2)+R^2(R^2-4r^2)}{R^2-4r^2} \\
&= \frac{R^2(4r-R^2-1)}{4r^2-R^2}.
\end{aligned}$$

Setting $|\rho(\xi)|^2 \leq 1$ is now equivalent to $R^2(4r-R^2-1) \geq R^2-4r^2$, moving everything to the right hand side, results in $(R^2-2r)^2 \leq 0$, thus for $r < \frac{R^2}{2}$ the scheme will be unstable. To conclude, two conditions are derived, namely:

$$\begin{aligned}
&\textit{Case 1}\text{: for } R^2 \leq 4r^2 \text{ one needs } \frac{|R|}{2} \leq r \leq \frac{1}{2}, \\
&\textit{Case 2}\text{: for } R^2 > 4r^2 \text{ one needs } \frac{R^2}{2} \leq r \leq \frac{|R|}{2}.
\end{aligned}$$

Inserting the original coefficients leads to

$$\begin{aligned}
&\textit{Case 1}\text{: for } \left(\frac{a\Delta t}{\Delta x}\right)^2 \leq 4\left(\frac{b\Delta t}{\Delta x^2}\right)^2 \text{ one needs } \frac{|a|\Delta t}{2\Delta x} \leq \frac{b\Delta t}{\Delta x^2} \leq \frac{1}{2}, \\
&\textit{Case 2}\text{: for } \left(\frac{a\Delta t}{\Delta x}\right)^2 > 4\left(\frac{b\Delta t}{\Delta x^2}\right)^2 \text{ one needs } \frac{a^2\Delta t^2}{2\Delta x^2} \leq \frac{b\Delta t}{\Delta x^2} \leq \frac{|a|\Delta t}{2\Delta x}.
\end{aligned}$$

Al tough the resulting inequalities in the two cases contradict, this method can still be stable. Rewriting *Case 1* and *Case 2* leads to:

$$\textit{Case 1}\text{: } \Delta x^2 \leq \frac{4b^2}{a^2} \text{ and } \textit{Case 2}\text{: } \Delta x^2 > \frac{4b^2}{a^2}$$

The reason why this scheme can still be stable is that, starting from the second case, as $\Delta x \to 0$ eventually *Case 1* will hold. Concluding that the most critical stability inequality is $\frac{|R|}{2} \leq r$. When the original coefficients are substituted this condition equals

$$\frac{|a|\Delta x}{2b} \leq 1 \tag{7.6}$$

and is known as the mesh Péclet number. In addition to stabillity, oscillations can occur when the coefficients in the finite difference scheme have certain values. Let C_1, C_2 and C_3 denote the coefficients in (7.4):

$$u_j^{k+1} = u_{j-1}^k\underbrace{(-\frac{a\Delta t}{2\Delta x}+\frac{b\Delta t}{\Delta x^2})}_{C_1} + u_j^k\underbrace{(1-2\frac{b\Delta t}{\Delta x^2})}_{C_2} + u_{j+1}^k\underbrace{(\frac{a\Delta t}{2\Delta x}+\frac{b\Delta t}{\Delta x^2})}_{C_3}. \tag{7.7}$$

The sign of these coefficients can effect the accuracy of the solution. A stable solution can have spurious oscillations. To see this, consider a stationary solution to the pure convection problem ($b = 0$ and $\frac{\partial u}{\partial t} = 0$). When the central scheme is applied the interior solution points will satisfy:

$$u_{j-1}^k = u_{j+1}^k \quad \text{for } j = 2, \ldots, N_x - 1.$$

The result of this is that the solution points are divided in two types: the even points and the odd points. These form two independent sets of coupled points. The problem is that, for the interior points, there is no direct connection between the odd and the even points. This feature can cause unwanted solution growth and are caused by the the dominance of convection term a over diffusion term b. In this case the mesh Péclet number (7.6) can be violated. PDEs for which these problems occur are said to be *convection-dominated.* In the Heston model, the convection coefficient $\kappa(\eta - v)$, is non constant. Therefore, the convection term can become dominant from a particular value of v on. This is why the central scheme can become unstable from a specific value of v on and the spurious oscillations can occur. As discussed in for example Andersen and Piterbarg [14], these occur when for some grid point j, either $C_1 < 0$ or $C_3 < 0$. From (7.7) it is seen that requiring this coefficients to be strictly positive this is equivalent to:

$$\begin{aligned} &\frac{-a\Delta t}{2\Delta x} + \frac{b\Delta t}{\Delta x^2} \geq 0 \text{ and } \frac{a\Delta t}{2\Delta x} + \frac{b\Delta t}{\Delta x^2} \geq 0 \\ \iff &|a\Delta x| \leq 2b. \end{aligned} \tag{7.8}$$

To avoid this problem a one-sided, so called upwinding scheme can be applied. Instead of discretizing the first derivative in a central way, a one sided discretization is used. Applying this method correctly it will artificially increase the diffusion term, by which the dominance of the convection is reduced. Assume b to be positive. First, when a is negative, a backward upwind scheme is applied:

$$\begin{aligned} \frac{u_j^{k+1} - u_j^k}{\Delta t} &= a\frac{u_j^k - u_{j-1}^k}{\Delta x} + b\frac{u_{j+1}^k - 2u_j^k + u_{j-1}^k}{\Delta x^2} \\ &= a\frac{u_{j+1}^k - u_{j-1}^k}{2\Delta x} - \frac{a\Delta x}{2}\frac{u_{j+1}^k - 2u_j^k + u_{j-1}^k}{\Delta x^2} + b\frac{u_{j+1}^k - 2u_j^k + u_{j-1}^k}{\Delta x^2} \\ &= a\frac{u_{j+1}^k - u_{j-1}^k}{2\Delta x} + \left(b - \frac{a\Delta x}{2}\right)\frac{u_{j+1}^k - 2u_j^k + u_{j-1}^k}{\Delta x^2}. \end{aligned}$$

Secondly, when a is positive the forward upwind scheme is applied:

$$\begin{aligned} \frac{u_j^{k+1} - u_j^k}{\Delta t} &= a\frac{u_{j+1}^k - u_j^k}{\Delta x} + b\frac{u_{j+1}^k - 2u_j^k + u_{j-1}^k}{\Delta x^2} \\ &= a\frac{u_{j+1}^k - u_{j-1}^k}{2\Delta x} + \frac{a\Delta x}{2}\frac{u_{j+1}^k - 2u_j^k + u_{j-1}^k}{\Delta x^2} + b\frac{u_{j+1}^k - 2u_j^k + u_{j-1}^k}{\Delta x^2} \\ &= a\frac{u_{j+1}^k - u_{j-1}^k}{2\Delta x} + \left(b + \frac{a\Delta x}{2}\right)\frac{u_{j+1}^k - 2u_j^k + u_{j-1}^k}{\Delta x^2}. \end{aligned}$$

Clearly when $|a|$ gets larger the upwinding approach will increase the diffusion coefficient and (7.8) is met. A negative consequence of using this upwinding scheme is the loss of accuracy. This can be handled by using a second order one-sided difference scheme like (3.10) and (3.9). Another idea is to use a mixture of the central and upwinding scheme:

$$\frac{u_j^{k+1} - u_j^k}{\Delta t} = \lambda a\frac{u_{j+1}^k - u_{j-1}^k}{2\Delta x} + (1-\lambda)a\frac{u_j^k - u_{j-1}^k}{\Delta x} + b\frac{u_{j+1}^k - 2u_j^k + u_{j-1}^k}{\Delta x^2}. \tag{7.9}$$

This will be shown with the help of some tests in the results section.

7.3 Convergence

As mentioned above, convergence is related to stability and consistency via the Lax Equivalence Theorem. This theorem states as follows:

Theorem 7.3. *(Lax Equivalence Theorem)* A consistent, two level difference scheme for a well-posed linear initial-value problem (of the form (3.19)) is convergent if and only if it is stable.

Thus, when a scheme is consistent, it is convergent when it is stable. This is a convenient property because now one only needs to prove consistency (which can be done via Taylor expansions) and stability (which can be done via Fourier techniques). A full proof of this important theorem can be found in [15].

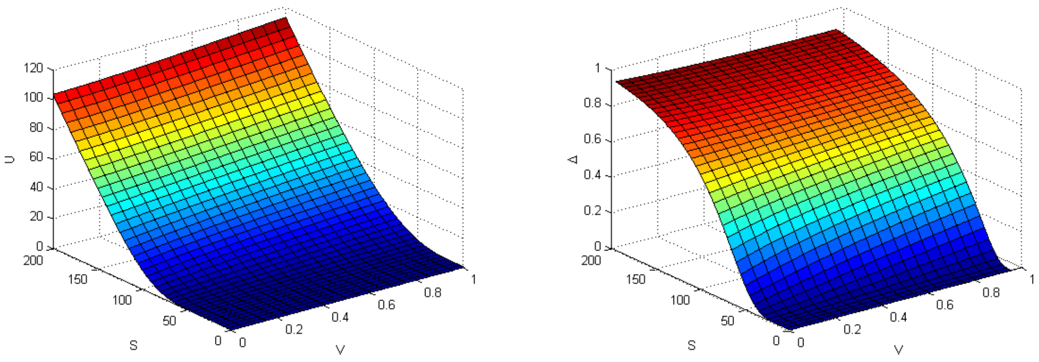
8 Numerical experiments

8.1 Heston with European pay-off

To verify the above derived method, some parameter values are needed. The parameter values introduced in case A, table 1 from [4] are used:

$$\begin{array}{lll} S_{\max} = 200 & \kappa = 3 & \theta = 0.8 \\ V_{\max} = 1 & \eta = 0.12 & r = 0.03 \\ T = 1 & \sigma = 0.041 & \\ K = 100 & \rho = 0.60 & \end{array} \tag{8.1}$$

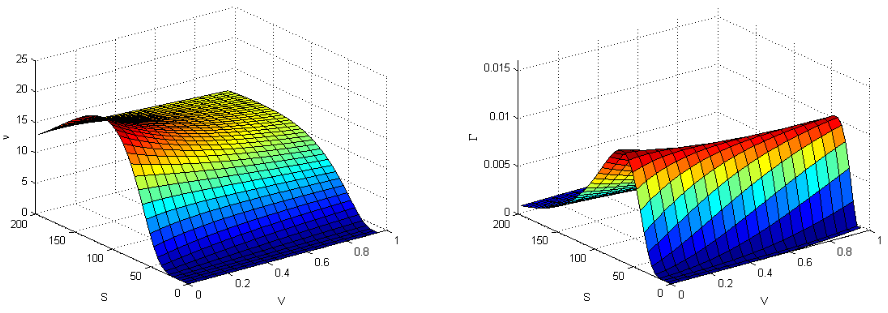
For the first results the Douglas-scheme is used. After that, results for the four different ADI schemes introduced in chapter 5 are compared and from that a preferred scheme is chosen which is used for pricing barrier options and pricing with the SABR model. The price surface is shown below. Next to that, the relation between fluctuation in stock price and the option value is often studied. This is done via the first derivative w.r.t s and is known as $\Delta := \frac{\partial u}{\partial s}$. Having obtained the solution u the derivative w.r.t s can be obtained by the central differences (3.8) used before for the interior points and a one-sided approximation ((3.9) and (3.10)) for the boundary points. The price surface and Δ are shown in figure 8.1a and 8.1b:



8.1a Price surface using Douglas scheme and parameters (8.1)

8.1b $\Delta^{\text{mp}} := \frac{\partial u}{\partial s}$

Next to Δ, the other important Greeks $\Gamma^{\text{mp}} := \frac{\partial^2 u}{\partial s^2}$, and $\nu^{\text{mp}} := \frac{\partial u}{\partial v}$ can be directly computed using a finite difference scheme, see figure 8.1c and 8.1d:



8.1c $\nu^{\text{mp}} := \frac{\partial u}{\partial v}$

8.1d $\Gamma^{\text{mp}} := \frac{\partial^2 u}{\partial s^2}$

8.1.1 Richardson extrapolation

In the end the most important property of the developed scheme is how accurately does it approximate the real price of an option at $t = 0$? As mentioned in section 7, the order of convergence of the scheme can be deduced from the used Taylor expansions. All the ADI schemes use second order central, forward and backward schemes. Hence a second order behaviour is expected in the space direction. Also, from literature, a second order behaviour in time for the Hundsdorfer-Verwer scheme is expected. To test convergence, without computing the exact solution, Richardson extrapolation can be used. In this method it is assumed that the solution of the ADI scheme ($U_h^{\Delta\tau}$ where $h = \sqrt{\Delta s \Delta v}$) and the exact solution (U_{exact}) at $\tau = T$ relate as follows:

$$||U_h^{\Delta\tau}||_{L^\infty} - ||U_{\text{exact}}||_{L^\infty} = Kh^p + L\Delta\tau^q, \tag{8.2}$$

where K and L are constants and q and p are the convergence rates. The error is measured at maturity $\tau = T$ in the L_∞-norm, also called the maximum norm:

$$||U_h^{\Delta\tau}||_{L^\infty} = \max\left(|u_{0,0}|, \cdots, |u_{0,m_2}|, \cdots, \cdots, |u_{m_1,0}|, \cdots, |u_{m_1,m_2}|\right). \tag{8.3}$$

Assuming $e_{\Delta\tau} := L\Delta\tau^q$, equation (8.2) consists of 3 unknowns and the convergence rate p can be calculated using 3 computations of the ADI scheme for h, $\frac{h}{2}$ and $\frac{h}{4}$, resulting in the following set of equations:

$$||U_h^{\Delta\tau}||_{L^\infty} - ||U_{\text{exact}}||_{L^\infty} = Kh^p + e_{\Delta\tau}, \tag{8.4a}$$

$$||U_{\frac{h}{2}}^{\Delta\tau}||_{L^\infty} - ||U_{\text{exact}}||_{L^\infty} = K\left(\frac{h}{2}\right)^p + e_{\Delta\tau}, \tag{8.4b}$$

$$||U_{\frac{h}{4}}^{\Delta\tau}||_{L^\infty} - ||U_{\text{exact}}||_{L^\infty} = K\left(\frac{h}{4}\right)^p + e_{\Delta\tau}. \tag{8.4c}$$

Dividing $2^p \cdot$ ((8.4c)-(8.4b)) by ((8.4b)-(8.4a)) results in:

$$2^p \frac{||U_{\frac{h}{4}}^{\Delta\tau}||_{L^\infty} - ||U_{\frac{h}{2}}^{\Delta\tau}||_{L^\infty}}{||U_{\frac{h}{2}}^{\Delta\tau}||_{L^\infty} - ||U_h^{\Delta\tau}||_{L^\infty}} = 1 \iff$$

$$p = \log\left(\frac{||U_{\frac{h}{2}}^{\Delta\tau}||_{L^\infty} - ||U_h^{\Delta\tau}||_{L^\infty}}{||U_{\frac{h}{4}}^{\Delta\tau}||_{L^\infty} - ||U_{\frac{h}{2}}^{\Delta\tau}||_{L^\infty}}\right) / \log(2).$$

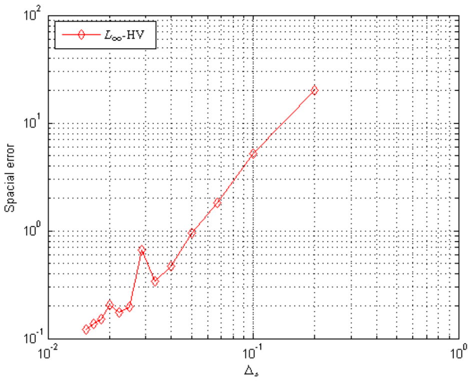
To calculate the convergence in time direction, similar calculations as above result in an expression for q. Next tables show the obtained results, and confirm expectations.

h	$U_h^{\Delta\tau}$	p
$\sqrt{180}$	104.6263	
$\sqrt{70 \cdot 1}$	96.9804	
$\sqrt{35 \cdot 0.5}$	95.3777	2.2542
$\Delta\tau$	$U_h^{\Delta\tau}$	q
$1.3 \cdot 10^{-3}$	96.5052	
$6.25 \cdot 10^{-4}$	96.5052	
$3.125 \cdot 10^{-4}$	96.5052	1.944

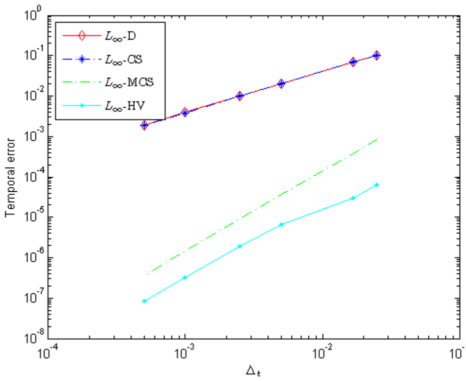
The schemes should give similar results when we test it against an exact solution.

8.1.2 Reference solution

To obtain an exact analytic solution the COS-method is used. The error is measured in the maximum norm over a pre-defined region of interest. In this error calculation the number of grid points in s direction and v direction are taken as $m_1 = 2M$ and $m_2 = M$ respectively where $M = 5, 10, 15, 20, \dots, 65$. Note that in the case of $M = 40$ we already compute $41 \times 81 = 3321$ grid points. To render the error introduced by the restriction of the spatial domain, the boundary is set at $S_{\max} = 14K$ and $V_{\max} = 10$, and the error is calculated in the region of interest: $(\frac{1}{2}K, \frac{3}{2}K) \times (0, 1)$. Because of this smaller region of interest the vector (8.3) that needs to be checked for maxima is smaller. Due to the large values of $S_{\max}$ and $V_{\max}$, Δs and Δv are still relatively large. Possible negative effects of these large differences can be reduced using a non-uniform grid. All the different ADI methods treat the space discretisation in the same way, this is why the value p is expected to be similar. A plot confirms this presumption.



8.1e Spatial error for Hundsdorfer-Verwer scheme, $\theta = 0.8$

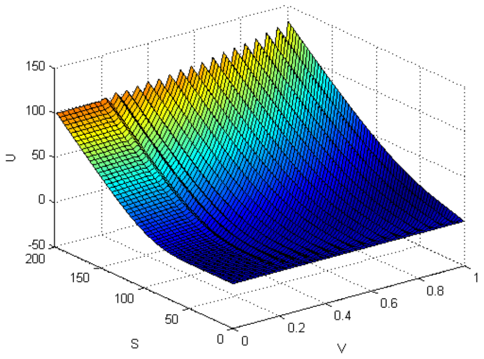


8.1f Temporal error for ADI $\theta = 0.8$

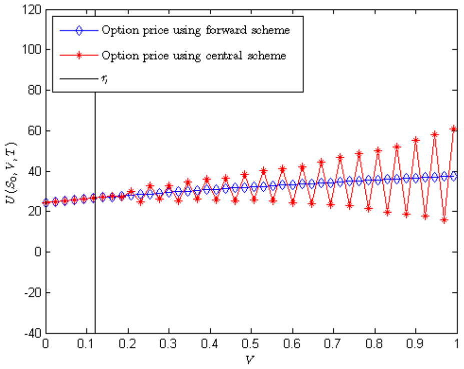
Because the spatial error is still relatively large compared to the analytic solution the COS-method gives us, the error in time direction will be dominated. This can give problems when looking at the behaviour of the error when $\Delta\tau$ decreases. By decreasing the step size in space direction no more improvement is observed. To be able to show the temporal error behaviour for the different methods, a reference solution is used by computing a solution for a very large amount of steps, say $T_N = 20000$ and fixed number of space grid points $M = 50$.. Then the method is tested on this reference solution for $T_N = 40, 60, 200, 400, 1000, 2000$. All the methods are computed with $\theta = 0.8$. Hence all results can be expected to be stable. The Douglas and Craig-Sneyd scheme behave first order in time, while the Modified Craig-Sneyd and the Hunsdorfer-Verwer scheme show a second order behaviour.

8.1.3 Upwinding

The central approach to a convection diffusion problem can give unstable results when the diffusion coefficient becomes negative. These instabilities occur in the form of oscillations. As mentioned, this can be mitigated by using a combination of the central and upwinding discretisation, this implies that in the region where the diffusion coefficient in v direction is positive a central scheme is applied while when this coefficient becomes negative, the one-sided alternative is used.

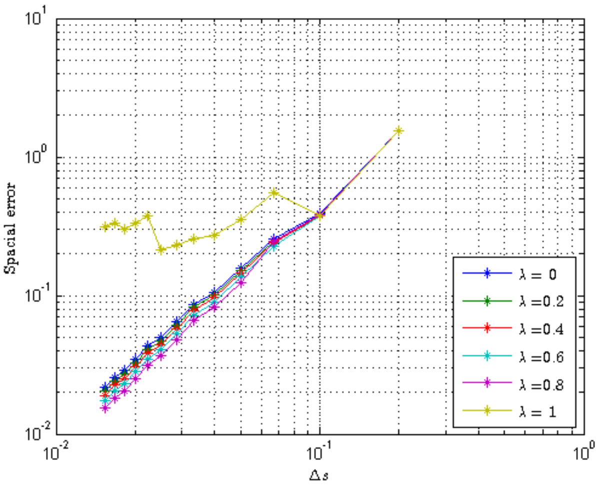


8.1g Price surface when only a central scheme is applied. Oscillations occur when convection coefficient $\kappa(\eta - v)$ becomes negative.



8.1h Oscillations for a fixed asset price $S_0 = 119$.

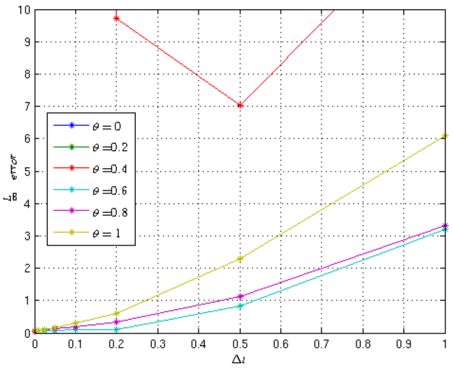
Another option is to use a weighted combination of upwinding and the central scheme. The error in space decreases when a combination is used. However, in the Heston case, one can argue about the significance of this improvement. Figure 8.1i shows the error for a variety of values of λ. The value $\lambda = 0.8$ seems to be most accurate.



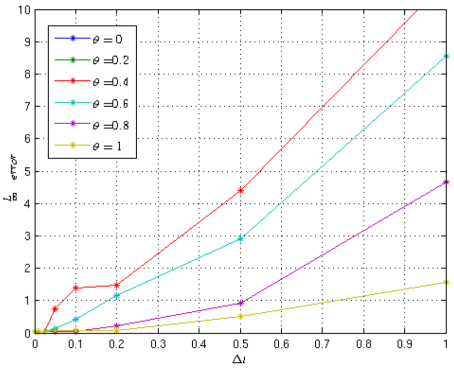
8.1i Error plotted for various values of λ. Oscillations for $\lambda = 1$ (fully central scheme) give rise to a large error.

8.1.4 Values of θ

Stability of the schemes is different for different values of θ. Varying the value of θ can enhance stability. The first derived Douglas ADI scheme has unstable results for $\theta < \frac{1}{2}$. As seen in the next figure 8.1j the error explodes for small values of $\Delta\tau$ when $\theta \leq 0.4$, while the Hundsdorfer-Verwer scheme gives stable results for values of $\theta \geq 0.4$. Because the fully explicit scheme has problems with discontinuities in the initial condition, this method is never stable. The Hundsdorfer-Verwer scheme does seem to be more stable, but for larger values of $\Delta\tau$ the method behaves better for larger values of θ. This can be explained by the fact that for larger values of θ the scheme becomes more and more implicit.



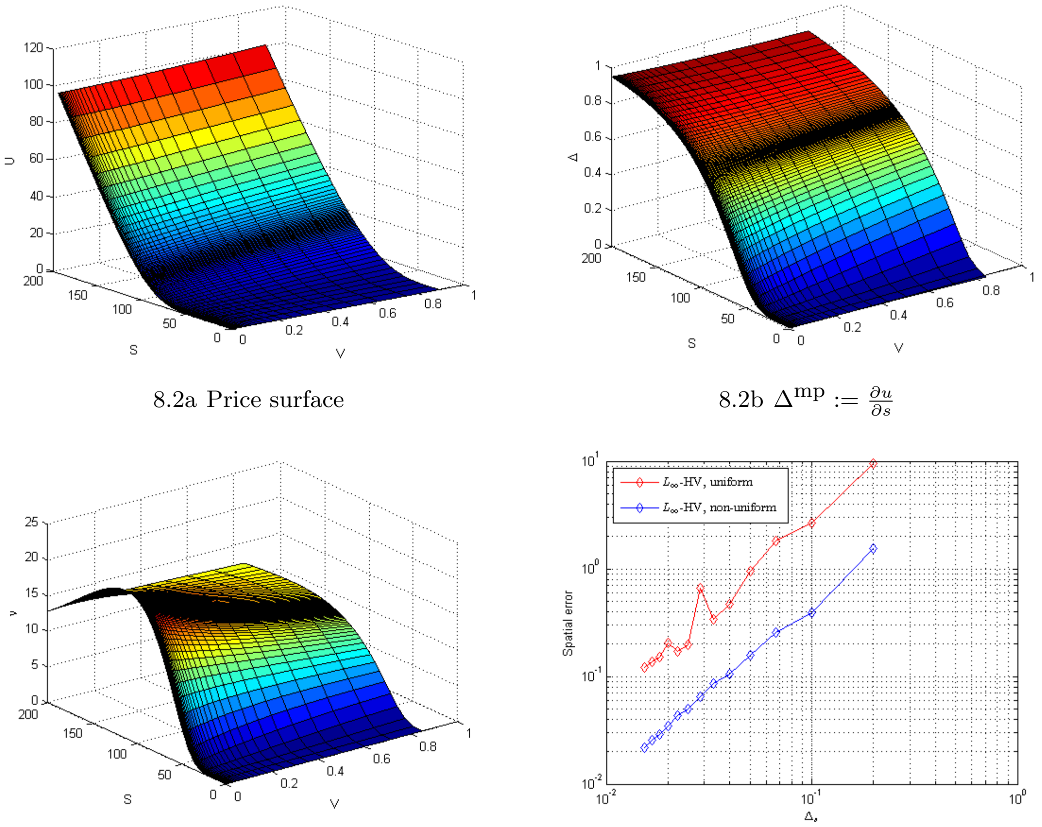
8.1j Douglas scheme for a number of values of θ. Note that for $\theta < 0.4$ the error is larger than 10 and thus not visible on the plot.



8.1k H-V scheme for a number of values of θ. Note that, again, for $\theta < 0.4$ the error larger than 10 and thus not visible on the plot.

8.2 Non-uniform grid

As mentioned above, a non-uniform grid should improve accuracy when we use a similar amount of grid points. Due to the local refinement of the grid less grid points need to be evaluated to reach a level of density at critical regions as $s = K$ and $v = 0$. The figure below shows the price surface and the Greeks (Δ and ν) for the non-uniform case. A direct important observation is that these Greeks do not precisely imply a similar grid refinement. Looking at the error in space, the refined grid significantly improves accuracy over the non-uniform grid. Also the oscillations disappear. Because the grid refinement does not affect the time discretisation, the error in time behaves similarly for uniform and non-uniform space grids.



8.2a Price surface

8.2b $\Delta^{\text{mp}} := \frac{\partial u}{\partial s}$

8.2c $\nu^{\text{mp}} := \frac{\partial u}{\partial v}$

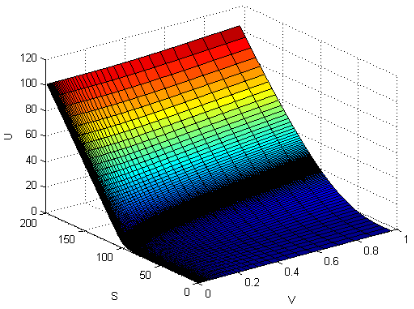
8.2d Spatial error for Hundsdorfer-Verwer scheme on a uniform and non-uniform grid, $\theta = 0.8$

8.3 Feller condition

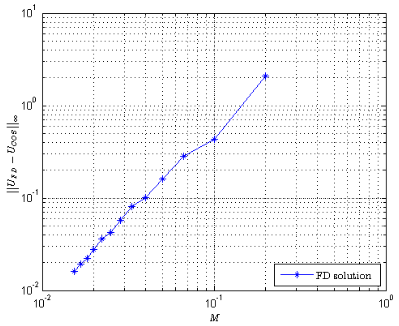
As mentioned before the Feller condition can influence the solution. In a more general sense, when this condition is not fulfilled, the stochastic process for the volatility is not strictly positive. When doing Monte Carlo simulations this can result in paths that can become negative, resulting in less accurate simulations. When applying (4.1d) at the boundary $v = 0$, the solution is claimed to be trustworthy in cases where the Feller condition is satisfied and where it is not satisfied, see [5], [16] and [4]. To test this claim, the following parameter set is used:

$$\begin{array}{lll} S_{\max} = 200 & \kappa = 2 & \theta = 0.8 \\ V_{\max} = 1 & \eta = 0.012 & r = 0.03 \\ T = 1 & \sigma = 0.4 & \\ K = 100 & \rho = 0.60 & \end{array} \tag{8.5}$$

Here, one has $\frac{2\kappa\eta}{\sigma^2} = 0.3 \not> 1$, so Feller is not satisfied. The COS-method is again applied as a reference solution.



8.3a Price surface using FDM. Feller condition is not satisfied.



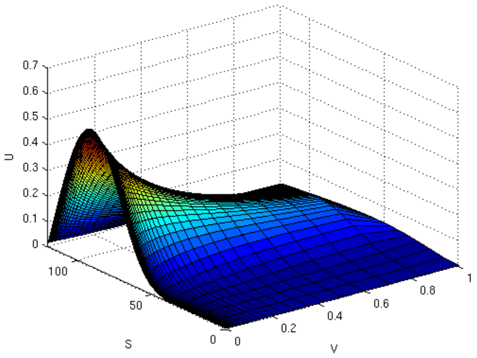
8.3b Error with respect to COS method.

8.4 Up-and-out call option

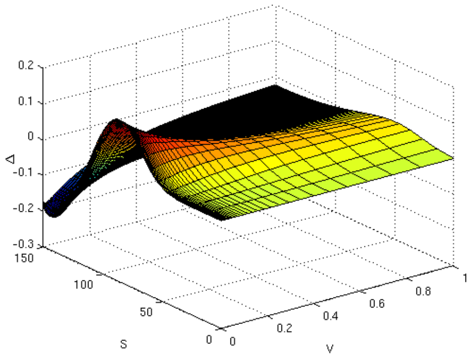
In the case of European up-and-out call options the previous described grid refinement can be used. Next to that the initial solution has an extra discontinuity at the barrier $s = B$. Due to this discontinuity a so called boundary layer is introduced. The solution has a sharp gradient because at $s = B$ it equals zero. Following the theory, upwinding is applied in s direction. In the results the forward scheme is implemented for the first derivative with respect to s. Another enhancement to increase accuracy comes from [4], where it is suggested to use a damping procedure at $\tau = 0$. This damping procedure comes down to applying the Douglas scheme for $\theta = 1$ at $\tau = 0$ and then use the Hundsdorfer-Verwer scheme. Using the parameters from (8.6),

$$\begin{array}{lll} S_{\max} = B = 150 & \kappa = 3 & \theta = 0.8 \\ V_{\max} = 10 & \eta = 0.12 & r = 0.03 \\ T = 1 & \sigma = 0.041 & \\ K = 100 & \rho = 0.60 & \end{array} \tag{8.6}$$

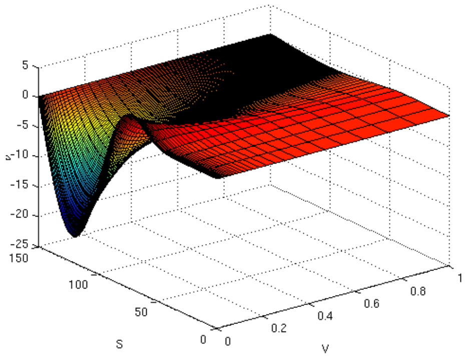
figures 8.4a, 8.4b, 8.4c and 8.4d show the price surface and the Greeks.



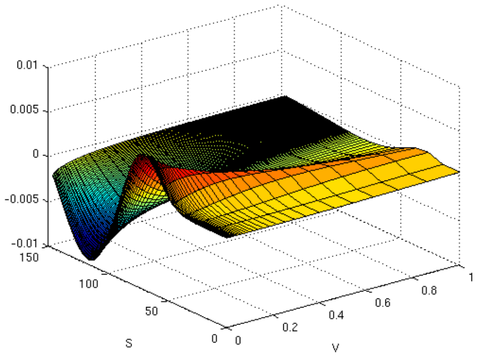
8.4a Price surface



8.4b $\Delta^{\text{mp}} := \frac{\partial u}{\partial s}$



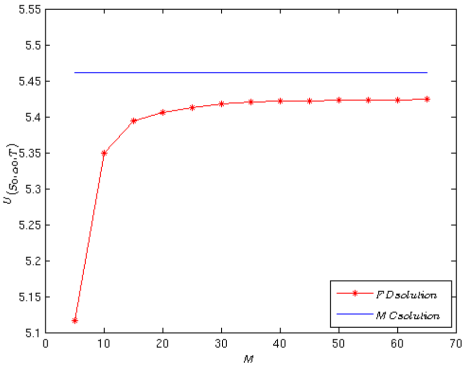
8.4c $\nu^{\text{mp}} := \frac{\partial u}{\partial v}$



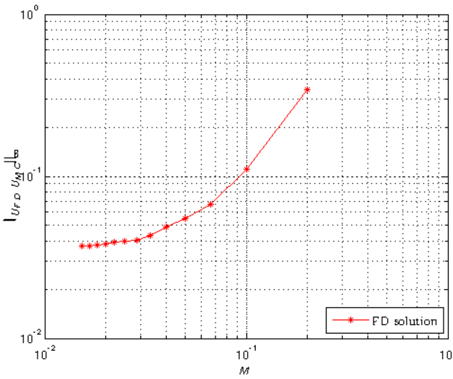
8.4d $\Gamma^{\text{mp}} := \frac{\partial^2 u}{\partial s^2}$

8.4.1 Reference solution

To test the method, it can be compared with solutions obtained by Monte Carlo simulations. In appendix A.4 the used Monte Carlo method is derived. Because of the large computational time of this method it is no longer reasonable to calculate derivative prices for all defined spot and volatility values in the finite difference grid. Instead of this "brute force" approach used before, now the deviation is measured at one specific spot and volatility value. Because the non-uniform grid discretizes the region with help of the tricky sinh function, specific values for the spot and volatility cannot be claimed to on be a grid point beforehand. Bilinear interpolation or grid shifting can be used to approach the function value at the desired grid point. See the appendix A.5 for a derivation of bilinear interpolation. Below is a result for an up-and-out barrier option where the finite difference method is compared to a Monte Carlo solution obtained with 10^6 paths and 10^3 time steps for $s_0 = 108.2$ and $v_0 = 0.01$, with the parameters from (8.6). Here grid shifting in v direction is imposed, figure 8.4e and 8.4f show the results. Clearly the finite difference solution approaches the Monte Carlo solution. Also the interpolation in s direction causes that the FDM solution can be fluctuating. Keeping in mind that the Monte Carlo solution might be accurate only up to 10^{-2} w.r.t the exact price, it can be concluded that the FDM is a good approximation.



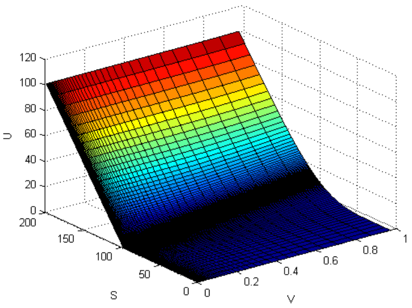
8.4e Non-uniform finite difference solution compared to Monte Carlo with 10^6 paths for barrier option



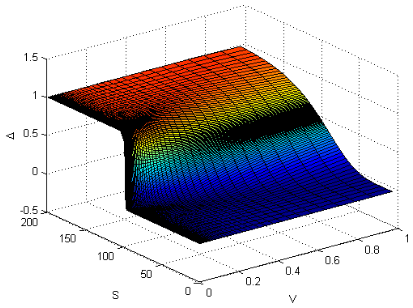
8.4f Non-uniform finite difference solution compared to Monte Carlo with 10^6 paths for barrier option

8.5 SABR European call

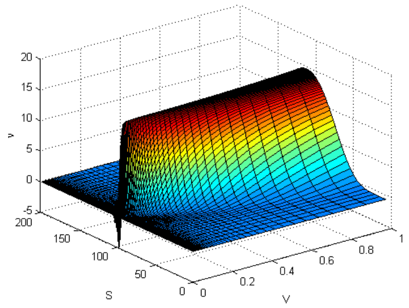
To price European call options the PDE derived in (2.17) is used. This PDE contains a convection and diffusion term in s direction and only a convection term in α direction. At the $\alpha = 0$ boundary there is also lack of convection in α direction. Therefore the discontinuity at $s = K$ at the $\alpha = 0$ boundary will still be present at $\tau = T$. This discontinuity will affect the computation of the Greeks. The parameters used are equal as in the Heston case (8.1). In addition, the discount factor is taken equal to $D(t, T) = e^{-r(T-t)}$ and $\beta = 0.7$:



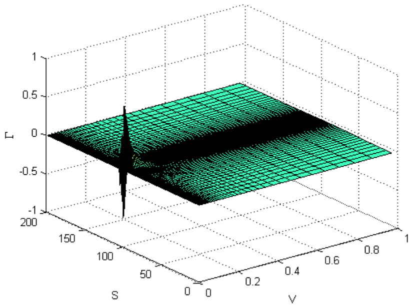
8.5a Price surface



8.5b $\Delta^{\text{mp}} := \frac{\partial u}{\partial s}$



8.5c $\nu^{\text{mp}} := \frac{\partial u}{\partial v}$



8.5d $\Gamma^{\text{mp}} := \frac{\partial^2 u}{\partial s^2}$

8.5.1 Reference solution

As a reference solution the proposed analytic solution in the paper of Hagan, Kumar, Lesniewski and Woodward [8] can be used. Although it is an approximation, it is proven to be accurate for small maturities. Maturities up to one year are assumed small in this sense. The method uses the following derivative price for a European call option:

$$V_{\text{call}} = e^{-r(T-t)}(fN(d_1) - KN(d_2)),$$

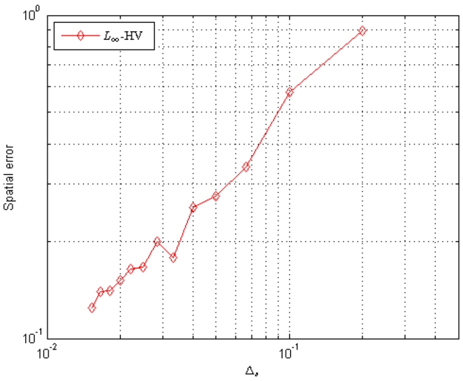
with the d_1 and d_2 similar as in the Black Scholes case:

$$d_{1,2} = \frac{\log \frac{f}{K} \pm \frac{1}{2}\sigma^2 T}{\sigma\sqrt{T}},$$

where the implied volatility is given by:

$$\sigma(f,K) = \frac{\alpha}{(fK)^{(1-\beta)/2}\left[1 + \frac{(1-\beta)^2}{24}\log^2\frac{f}{K} + \frac{(1-\beta)^4}{1920}\log^4\frac{f}{K} + ...\right]}\frac{z}{\chi(z)}$$
$$\cdot \left[1 + \left[\frac{(1-\beta)^2}{24}\frac{\alpha^2}{(fK)^{(1-\beta)}} + \frac{1}{4}\frac{\rho\beta\sigma\alpha}{(fK)^{(1-\beta)/2} + \frac{2-3\rho^2}{24}\sigma^2}\right]T + ...\right].$$

This analytic form can be used to approximate the error in the finite difference scheme:



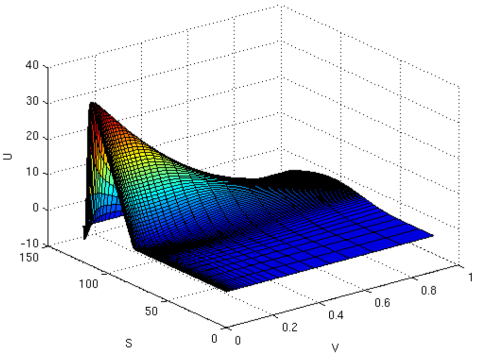
8.5e Error of the FD solution w.r.t. analytic solution of Hagan et al

8.6 SABR barrier up-and-out

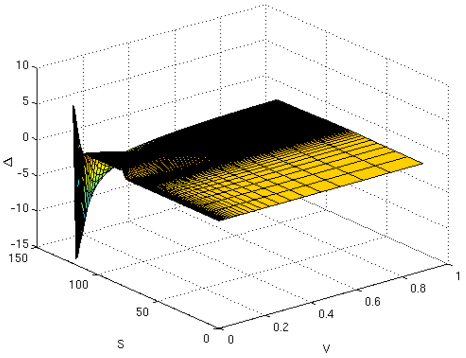
Similar as in the Heston case, pricing an up-and-out barrier with finite differences acquires adjustments in the vectors determined by the boundary conditions. Furthermore, a one sided finite difference should be used in s -direction. The parameters are taken as follows:

$$\begin{array}{lll} S_{\max} = B = 150 & \kappa = 3 & \theta = 0.8 \\ V_{\max} = 10 & \eta = 0.12 & r = 0.03 \\ T = 1 & \sigma = 0.17 & \beta = 1 \\ K = 100 & \rho = 0.11 & D(t,T) = e^{-r(T-t)} \end{array} \tag{8.7}$$

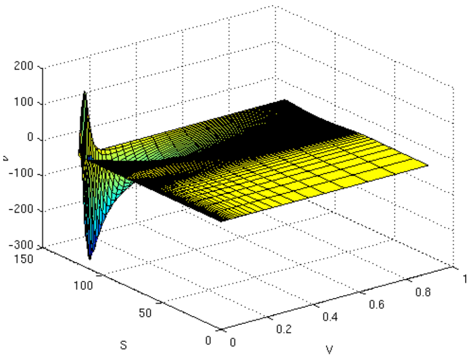
Results for the price surface are shown below. The boundary conditions are similar to the Heston case. Next to that the Greeks are plotted.



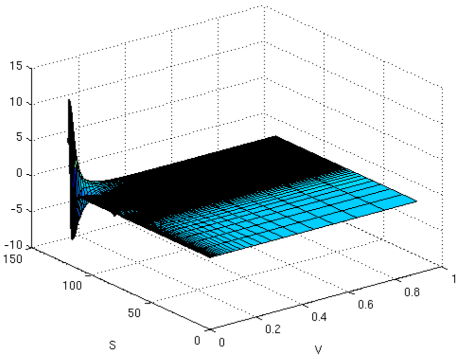
8.6a Price surface



8.6b $\Delta^{\text{mp}} := \frac{\partial u}{\partial s}$



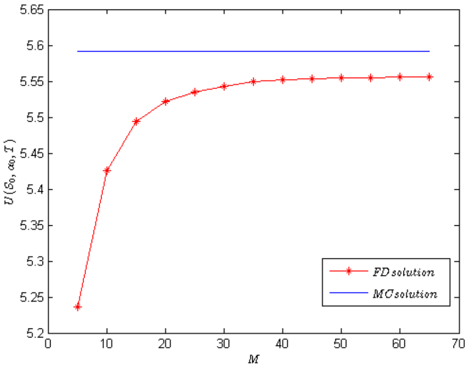
8.6c $\nu^{\text{mp}} := \frac{\partial u}{\partial v}$



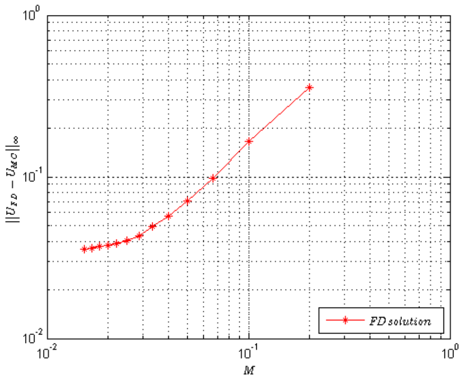
8.6d $\Gamma^{\text{mp}} := \frac{\partial^2 u}{\partial s^2}$

8.6.1 Reference solution

As a reference solution again a Monte Carlo method is used. The number of paths is chosen up to $4 * 10^4$, to make sure that the solution is accurate. Because of this large number of paths the solution is only calculated for one spot and one volatility value, namely $s_0 = 115.2$ and $\alpha_0 = 0.29$. Again the finite difference solution for these spot and volatility values are obtained using a bilinear routine and using the grid shifting procedure from section 6.2. In figure 8.6e and 8.6f the Monte Carlo solution uses $2*10^4$ time steps. Clearly the finite difference solution approaches the Monte Carlo solution. Also the L_∞-norm of the deviation is plotted. We can see that the finite difference method approaches the solution obtained by the Monte Carlo method up to the order of 10^{-2}. In this sense the finite difference method delivers a good approximation.



8.6e Non-uniform finite difference solution compared to Monte Carlo with 10^6 paths for barrier option



8.6f Nonuniform finite difference solution compared to Monte Carlo with $4 * 10^4$ paths and $2 * 10^4$ time steps for barrier option

9 Conclusion

In this thesis, the finite difference method is applied to the Heston model and the SABR model. To benefit from the stable behaviour of implicit schemes, the ADI scheme is applied. This mixture of an implicit explicit scheme, can be seen as "the best of both worlds" where it has the stability characteristics from the implicit scheme, but also limits the computation time of matrix inversion. This is done by splitting the matrix per direction.

First the Heston model with a European pay-off is extensively tested. Important delicate details of the finite difference method are treated such as upwinding, boundary conditions and the use of non-uniform grids.

Also the Greeks are computed, this can be done easily because the FDM produces option values for a whole grid, which actually is a whole set of spot prices and volatility values of the underlying.

As a reference solution the COS method is applied. The error with respect to the solution obtained with the COS method agrees with the theory. Also in case the Feller condition is not satisfied.

After that up- and out- call options are treated. The extra discontinuity of the initial pay-off function at the barrier, introduces a boundary layer and this is why upwinding is applied in s direction. Next to that, the described non-uniform grid is altered to fit this new discontinuity.

In this case the Monte Carlo method serves as a reference solution. Because this method calculates only one option price for a predefined spot price and volatility, also only one option price is needed from the FDM solution. To extract one option price belonging to a specific combination of strike price and volatility, two methods are used:

- Bilinear interpolation
- and grid shifting.

In addition a damping procedure is used, by performing the first time step fully implicit.

At last all the obtained results are applied to the SABR model. Again European and up-and-out call options are studied. In the European case the result is tested against the analytical solution introduced by Hagan et al. For short maturities this solution is a good approximation. For the up-and-out call options again the Monte Carlo method serves as a benchmark. It can be concluded that modelling the SABR model with the help of finite differences can well be done. A suggestion for further research would be to apply grid shifting in s-direction.

A Appendix

A.1 Heston's analytic solution

For some values an analytic solution of the Heston PDE can be derived. This solution will be used to do error analysis for the ADI methods. First derivation in the original paper of Heston [1] is studied. The PD derived in (2.14) states:

$$\begin{aligned}&\frac{1}{2}\sigma^2 v\frac{\partial^2 u}{\partial v^2}+\sigma s v\rho\frac{\partial^2 u}{\partial s\partial v}+\frac{1}{2}vs^2\frac{\partial^2 u}{\partial s^2}\\&+\kappa(\eta-v)\frac{\partial u}{\partial v}+rs\frac{\partial u}{\partial s}+\frac{\partial u}{\partial\tau}-ru=0.\end{aligned}\tag{A.1}$$

Similar as in the, Black-Scholes formula, a solution of the form is guessed

$$C(s,v,\tau)=sP_1(s,v,\tau)-KP(\tau,T)P_2(s,v,\tau),\tag{A.2}$$

where $P(\tau,T)$ denotes the discount factor. The first term is correctly interpreted as the present value, using the risk-free interest rate, of the expected stock price at expiration, given that the stock price at expiration is above the exercise price. The second term reflects the probability of the call option to be exercised, provided the stock drift is the risk-free rate, multiplied with the discounted strike price. For convenience the logarithm of the stock price is evaluated, so $x=\ln(s)$. Using standard change of variable techniques, one has $\frac{\partial}{\partial s}=\frac{\partial}{\partial x}\frac{\partial x}{\partial s}=\frac{1}{s}\frac{\partial}{\partial x}$ such that (A.1) becomes:

$$\begin{aligned}&\frac{1}{2}\sigma^2 v\frac{\partial^2 C}{\partial v^2}+\sigma v\rho\frac{\partial^2 C}{\partial x\partial v}+\frac{1}{2}vs^2\frac{\partial}{\partial s}\left(\frac{1}{s}\frac{\partial C}{\partial x}\right)+\kappa(\eta-v)\frac{\partial C}{\partial v}+r\frac{\partial C}{\partial x}+\frac{\partial C}{\partial\tau}-rC\\&=\frac{1}{2}\sigma^2 v\frac{\partial^2 C}{\partial v^2}+\sigma v\rho\frac{\partial^2 C}{\partial x\partial v}+\frac{1}{2}vs^2\left(\frac{1}{s^2}\frac{\partial^2 C}{\partial x^2}-\frac{1}{s^2}\frac{\partial C}{\partial x}\right)+\kappa(\eta-v)\frac{\partial C}{\partial v}+r\frac{\partial C}{\partial x}+\frac{\partial C}{\partial\tau}-rC\\&=\frac{1}{2}\sigma^2 v\frac{\partial^2 C}{\partial v^2}+\sigma v\rho\frac{\partial^2 C}{\partial x\partial v}+\frac{1}{2}v\frac{\partial^2 C}{\partial x^2}+(r-\frac{1}{2}v)\frac{\partial C}{\partial x}+\kappa(\eta-v)\frac{\partial C}{\partial v}+\frac{\partial C}{\partial\tau}-rC=0.\end{aligned}\tag{A.3}$$

Because (A.2) consists of two sums, two equations for P_1 and P_2 can be derived. Substituting sP_1 in (A.3) and multiplying with e^{-x} results in:

$$\frac{1}{2}\sigma^2 v\frac{\partial^2 P_1}{\partial v^2}+\frac{1}{2}v\frac{\partial^2 P_1}{\partial x^2}+\sigma v\rho\frac{\partial^2 P_1}{\partial x\partial v}+(r+\frac{1}{2}v)\frac{\partial P_1}{\partial x}+\frac{\partial P_1}{\partial v}\left(\sigma v\rho+\kappa(\eta-v)\right)+\frac{\partial P_1}{\partial\tau}=0.\tag{A.4}$$

Using the fact that $P(\tau,T)=e^{-r(T-\tau)}$ and dividing by $Ke^{-r(T-\tau)}$ yields the following equation for P_2:

$$\frac{1}{2}\sigma^2 v\frac{\partial^2 P_2}{\partial v^2}+\frac{1}{2}v\frac{\partial^2 P_2}{\partial x^2}+\sigma v\rho\frac{\partial^2 P_2}{\partial x\partial v}+(r-\frac{1}{2}v)\frac{\partial P_2}{\partial x}+\kappa(\eta-v)\frac{\partial P_2}{\partial v}+\frac{\partial P_2}{\partial\tau}=0.\tag{A.5}$$

At expiry the option value is known, namely it must be equal to 0 when the stock value is lower than the strike and equal to the positive difference between the strike and the stock value when the stock value is higher than the strike. In conclusion the following terminal condition can be deduced:

$$P_j(x,v,T)=\begin{cases}1 & \text{if } x\geq\ln K,\\ 0 & \text{else,}\end{cases}\qquad\text{for } j=1,2.\tag{A.6}$$

The appendix in the Heston [1] article shows that the characteristic function $f_j(x, v, \tau, \xi)$ for $j = 1, 2$ also satisfy equation (A.4) and (A.5) respectively subject to the terminal condition:

$$f_j(x, v, T, \xi) = e^{i\xi x}. \tag{A.7}$$

The solution to this characteristic function is

$$f_j(x, v, \tau, \xi) = e^{C(t,\xi)+D(t,\xi)v+i\xi x}, \tag{A.8}$$

where $t = T - \tau$ and:

$$\begin{aligned} C_j(t, \xi) &= r\xi it + \frac{\kappa\eta}{\sigma^2}\left((b_j - \rho\sigma\xi i + d)t - 2\ln\left[\frac{1 - ge^{dt}}{1 - g}\right]\right), \\ D(t, \xi) &= \frac{b_j - \rho\sigma\xi i + d}{\sigma^2}\left[\frac{1 - ge^{dt}}{1 - g}\right], \end{aligned} \tag{A.9}$$

and

$$\begin{aligned} g &= \frac{b_j - \rho\sigma\xi i + d}{b_j - \rho\sigma\xi i - d}, \\ d &= \sqrt{(\rho\sigma\xi i - b_j)^2 - \sigma^2(2u_j\xi i - \xi^2)}, \\ b_j &= \begin{cases} \kappa - \rho\sigma & \text{if } j = 1, \\ \kappa & \text{if } j = 2, \end{cases} \\ u_j &= \begin{cases} \frac{1}{2} & \text{if } j = 1, \\ \frac{-1}{2} & \text{if } j = 2. \end{cases} \end{aligned}$$

Using the Gil-Palaezs inversion [7], the inverse of the characteristic function can be calculated and with that, the probabilities from (A.2) are known:

$$P_j(x, v, T) = \frac{1}{2} + \frac{1}{\pi}\int_0^\infty \mathrm{Re}\left[\frac{e^{-i\xi\ln K} f_j(x, v, T, \xi)}{i\xi}\right] d\xi. \tag{A.10}$$

One problem of this method is that the existence of the Fourier-transform is not guaranteed to exist.

A.2 Carr-Madan inversion

To assure the Fourier-transform to exist, Carr-Madan [6] introduces a damped option value. First, let $k := \ln K$ and define the damped option value as follows:

$$c(x, v, \tau) = e^{\alpha k} C(x, v, \tau), \qquad \text{where } \alpha > 0. \tag{A.11}$$

The condition $\alpha > 0$ ensures the Fourier transform of the damped option value to exist. Deriving the damped option price follows slightly different steps as above and relates to the paper from Fang [2]:

$$\begin{aligned} C(x, v, \tau) &= e^{-r(T-\tau)} \mathbb{E}^{\mathbb{Q}} \left[(e^x - e^k)^+ \right] \\ &= e^{-r(T-\tau)} \int_{-\infty}^{\infty} (e^x - e^k)^+ f(x) dx, \end{aligned} \tag{A.12}$$

where $\mathbb{Q}$ denotes the risk neutral measure. Again the density function $f(x)$ is obtained via the characteristic function, which in his turn is obtained via the Fourier transform:

$$\phi(\xi) = \int_{-\infty}^{\infty} f(x) e^{i\xi x} dx. \tag{A.13}$$

Let $\tilde{c}$ denote the Fourier transform of the damped price, substituting (A.11) in (A.13) results in:

$$\begin{aligned} \tilde{c}(\xi) &= \int_{-\infty}^{\infty} e^{i\xi k} c(x, v, \tau) dk = \int_{-\infty}^{\infty} e^{i\xi k} e^{\alpha k} C(x, v, \tau) dk \\ &= \int_{-\infty}^{\infty} e^{k(\alpha + i\xi)} e^{-r(T-\tau)} \int_{-\infty}^{\infty} (e^x - e^k)^+ f(x) dx dk \\ &= e^{-r(T-\tau)} \int_{-\infty}^{\infty} f(x) \int_{-\infty}^{\infty} e^{k(\alpha+i\xi)} (e^x - e^k)^+ dk dx \\ &= e^{-r(T-\tau)} \int_{-\infty}^{\infty} f(x) \int_{-\infty}^{x} e^{k(\alpha+i\xi)} (e^x - e^k) dk dx \\ &= e^{-r(T-\tau)} \int_{-\infty}^{\infty} f(x) \left[\frac{1}{\alpha + \xi i} e^{k(\alpha+\xi i)+x} - \frac{1}{\alpha + \xi i + 1} e^{k(\alpha+\xi i+1)} \right]_{k=-\infty}^{x} dx \\ &= e^{-r(T-\tau)} \int_{-\infty}^{\infty} f(x) \frac{e^{x(\alpha+\xi i+1)}}{(\alpha + \xi i)(\alpha + \xi i + 1)} dx \\ &= \frac{e^{-r(T-\tau)}}{(\alpha + \xi i)(\alpha + \xi i + 1)} \int_{-\infty}^{\infty} f(x) e^{ix(-i\alpha+\xi-i)} dx \\ &= \frac{e^{-r(T-\tau)}}{(\alpha + \xi i)(\alpha + \xi i + 1)} \phi(\xi - i(\alpha + 1)), \end{aligned}$$

where $\phi\left(\xi - i(\alpha + 1)\right)$ is the characteristic function of the density function $f(x)$. So the Fourier transform of the damped option price is easily obtained via the characteristic function of the density. Using the inverse of the Fourier transform (and damping factor) the option price is known.

A.3 COS-method

The recently developed COS-method makes use of the Fourier-cosine series expansion. For a detailed description of this method it is wise to consult the paper by Fang [3] itself, because what follows below is a far from complete summary. The risk neutral valuation formula plays an important role:

$$u(x,t_0) = e^{-r(T-t_0)}\mathbb{E}^{\mathbb{Q}}\left[u(y,T|x)\right] = e^{-r(T-t_0)}\int_{\mathbb{R}} u(y,T)f(y|x)dy, \tag{A.14}$$

where u denotes the option value, T maturity date, t_0 initial date and $\mathbb{E}^{\mathbb{Q}}$ is the expectation p[erator under the risk neutral measure $\mathbb{Q}$. Furthermore x and y are state variables at t_0 and T respectively, $f(y|x)$ is the probability density of y given x, and r is the risk neutral interest rate. Now, the density function is replaced by its cosine expansion:

$$f(y|x) = \sum_{k=0}^{\infty'} A_k(x)\cos\left(k\pi\frac{y-a}{b-a}\right), \tag{A.15}$$

where $\sum'$ denotes that the first term in the summation is weighted by a half. The coefficients $A_k(x)$ are given as:

$$A_k(x) := \frac{2}{b-a}\int_a^b f(y|x)\cos\left(k\pi\frac{y-a}{b-a}\right)dy. \tag{A.16}$$

such that

$$u_1(x,t_0) = e^{-r(T-t_0)}\int_a^b u(y,T)\sum_{k=0}^{'\infty} A_k(x)\cos\left(k\pi\frac{y-a}{b-a}\right)dy, \tag{A.17}$$

interchanging summation and integration and defining V_k as follows:

$$V_k := \frac{2}{b-a}\int_a^b u(y,T)\sum_{k=0}^{'\infty} A_k(x)\cos\left(k\pi\frac{y-a}{b-a}\right)dy, \tag{A.18}$$

yields:

$$u_2(x,t_0) = \frac{1}{2}(b-a)e^{-r(T-t_0)}\sum_{k=0}^{'\infty} A_k(x)V_k. \tag{A.19}$$

Now, the coefficients can be approximated using the truncated Fourier transform of the characteristic function:

$$\phi_1(\omega) := \int_a^b e^{i\omega x}f(x)dx \approx \int_{\mathbb{R}} e^{i\omega x}f(x)dx, \tag{A.20}$$

such that:

$$A_k \equiv \frac{2}{b-a}Re\{\phi_1\left(\frac{k\pi}{b-a}\right)\exp\left(-i\frac{ka\pi}{b-a}\right)\}, \tag{A.21}$$

where $Re\cdot$ denotes the real part of the argument. By substituting (A.21) in (A.19) the cos formula is derived:

$$u(x,t_0) \approx u_3(x,t_0) = e^{-r(T-t_0)} \sum_{k=0}^{N-1'} Re\{\phi\left(\frac{k\pi}{b-a;x} e^{-ik\pi\frac{a}{b-a}}\right)\}V_k. \tag{A.22}$$

In the case of a European call, V_k can be calculated analytically as follows:

$$V_k = \frac{2}{b-a}\int_0^b K(e^y - 1)\cos\left(k\pi\frac{y-a}{b-a}\right). \tag{A.23}$$

The coefficients a and b are so called cumulants and for more information one should study the paper by Fang [3].

A.4 Monte Carlo-method

The Monte Carlo method makes use of paths. These paths can be created with the help of the stochastic process. Let S_t be a stochastic process (e.g. stock price) driven by the following stochastic differential equation:

$$dS_t = \mu(S_t, t)dt + \sigma(S_t, t)dW_t, \tag{A.24}$$

where W_t is a Wiener process. To create a path over a given time horizon $[0, T]$ this horizon is uniformly discretized as $0 = t_0 < t_1 < \cdots < t_n = T$ with step size dt. Integrating (A.24) from t to $t + dt$ results in the following stochastic integral equation:

$$S_{t+dt} = S_t + \int_t^{t+dt} \mu(S_t, t)dt + \int_t^{t+dt} \sigma(S_t, t)dW_t. \tag{A.25}$$

To evaluate the two integrals a discretization is needed. A popular method is the Euler scheme, where the following approximation is used:

$$\begin{aligned}\int_t^{t+dt} \mu(S_t, t)dt &\approx \mu(S_t, t) \int_t^{t+dt} dt \\ &= \mu(S_t, t)dt.\end{aligned}$$

In the case of the second integral the Wiener process implies that the increments are normally distributed: $W_{t+dt} - W_t\ N(0, dt)$. Using this one can derive:

$$\begin{aligned}\int_t^{t+dt} \sigma(S_t, t)dW_t &\approx \sigma(S_t, t) \int_t^{t+dt} dW_t \\ &= \sigma(S_t, t)(W_{t+dt} - W_t) \\ &= \sigma(S_t, t)\sqrt{dt}Z,\end{aligned}$$

where Z is a standard normal variable. Hence the Euler discretization states:

$$S_{t+dt} = S_t + \mu(S_t, t)dt + \sigma(S_t, t)\sqrt{dt}Z. \tag{A.26}$$

In addition to a single stochastic process the Heston model contains two stochastic processes. The Euler method can be applied on both the stochastic processes. Because the Wiener processes are correlated with correlation ρ, the correlated normally distributed variables need to be computed with the help of the Cholesky matrix. Let Z_1 and Z_2 be two standard normally random variables, then:

$$\begin{aligned}\xi^{(1)} &= Z_1, \\ \xi^{(2)} &= \rho Z_1 + \sqrt{1 - \rho^2} Z_2.\end{aligned}$$

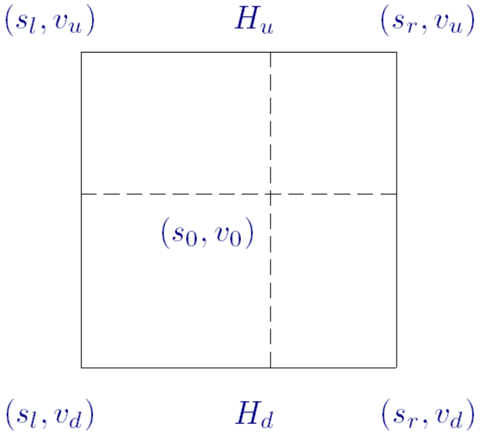
The following pseudo code now generates paths for a up-and-out barrier option for the Heston model:

```
• Start with $S_0$ and $V_0$.
• for i=1:Number of paths
    • for j=1:Number of time steps
```

```
        • create $\xi^{(1)}$ and $\xi^{(2)}$ with the help of the Cholesky decomposition.
        • create
              $S_{t+dt} = S_t + rS_t dt + \sqrt{V_t} S_t \sqrt{dt} \xi^{(2)}$ and
              $V_{t+dt} = V_t + \kappa * (\eta - V_t) dt + \sigma \sqrt{V_t} \sqrt{dt} \xi^{(1)}$.
        • if $V_{t+dt} < 0$; set $V_{t+dt} = 0$.
        • if $S_{t+dt} >= B$; set $S_{t+dt} = 0$ and exit for loop
    • end time stepping
    • Pay-off = $\max(0, S_N - K)$; sum=sum+Pay-off;
• end Path loop
• Price=$\frac{1}{N} e^{-rT}$sum
```

A.5 Bilinear Interpolation

Comparing a price for an option obtained by FDM for a specific spot price s_0 and volatility v_0 is not directly clear. If the stock value s_0 and volatility v_0 are determined beforehand the chance is small that the (s_0, v_0) is a grid point in the solution obtained by FDM. An idea is to alter the non-uniform grid in a way that it contains the specified grid point. This is left for further research. Another way to handle this problem is to apply bilinear interpolation. The desired grid point always lies in a certain square $s_l \leq s_0 \leq s_r$ and $v_d \leq v_0 \leq v_u$:



A.5a Square containing (s_0, v_0)

Bilinear interpolation comes down to subsequently applying linear interpolation twice first in horizontal way and then in vertical way as follows:

$$\begin{aligned} u(H_d) &\approx \tfrac{s_r - s_0}{s_r - s_l} u(s_l, v_u) + \tfrac{s_0 - s_l}{s_r - s_l} u(s_r, v_u), \\ u(H_u) &\approx \tfrac{s_r - s_0}{s_r - s_l} u(s_l, v_u) + \tfrac{s_0 - s_l}{s_r - s_l} u(s_r, v_u). \end{aligned} \tag{A.27}$$

Now the vertical interpolation with the help of H_u and H_d gives the desired approximation:

$$u(s_0, v_0) \approx \frac{v_u - v_0}{v_u - v_d} u(H_d) + \frac{v_0 - v_d}{s_r - s_l} u(H_u), \tag{A.28}$$

Substituting (A.27) in (A.28) one obtains the direct formula:

$$\begin{aligned} u(s_0, v_0) \approx & \frac{u(s_r, v_u)}{(s_r - s_l)(v_u - v_d)} (s_r - s_0)(v_u - v_0) \\ & + \frac{u(s_0 - s_l)}{(s_r - s_l)(v_u - v_d)} (s_0 - s_l)(v_u - v_0) \\ & + \frac{u(s_l, v_u)}{(s_r - s_l)(v_u - v_d)} (s_r - s_0)(v_0 - v_d) \\ & + \frac{u(s_r, v_u)}{(s_r - s_l)(v_u - v_d)} (s_0 - s_l)(v_0 - v_d) \end{aligned}$$

References

[1] S.L. Heston, *A closed-form solution for options with stochastic volatility with applications to bond and currency options*, The Review of Financial Studies, 6, 327-343, 1993.

[2] F. Fang, *Pricing Bermudan and American Options Using the FFT Method*, Diploma thesis, Universität Erlangen-Nuernberg, 2006.

[3] F. Fang, *A Novel Pricing Method For European options Based on Fourier-Cosine Series Expansions*, SIAM Journal on Scientific Computing 31, 826-848, 2008.

[4] T. Haentjens & K.J. in 't Hout, *ADI finite difference schemes for the Heston-Hull-White PDE*, Journal of Computational Finance, 16, 83-110, 2012.

[5] T. Kluge, *Pricing derivatives in stochastic volatility models using the finite difference method*, Diploma thesis, Technical University Chemnitz

[6] P. Carr & D.B. Madan, *Option Valuation Using the Fast Fourier Transform*, Journal of Computational Finance, 2, 4, 61-73,1999.

[7] J. Gil-Palaez, *Note on the Inversion Theorem*, Biometrika, 38, 481-482, 1951.

[8] P.S. Hagan & D. Kumar & A.S. Lesniewski & D.E. Woodward, *Managing Smile Risk*, 84-108, Willmot Magazine, 2002

[9] J.W. Thomas, *Numerical Partial Differential Equations: Finite Difference Methods*, Springer, New York, 1999.

[10] D.J. Duffy, *Finite Difference Methods in Financial Engineering*, John Wiley & Sons, England, 2006.

[11] P. Wilmott, *Derivatives. The theory and practice of financial engineering*, John Wiley & Sons, England, 1998.

[12] J. Hull & A. White, *The pricing of options on assets white stochastic volatilities.* Journal of Finance, 42:281-300, 1987

[13] T. Björk, *Arbitrage Theory in Continuous Time*, Oxford University Press, 2009

[14] L.B.G. Andersen & V.V. Piterbarg, *Interest Rate Modeling, Volume I: Foundations and Vanilla Models*, first edition, Atlantic Financial Press, 2010.

[15] J.C. Strikwerda, *Finite Difference Schemes and Partial Differential Equations.* first edition, Chapman & Hall, 1989.

[16] A. Janek, T. Kluge, R. Weron, U. Wystup, *FX Smile in the Heston Model.* working paper, 2010.