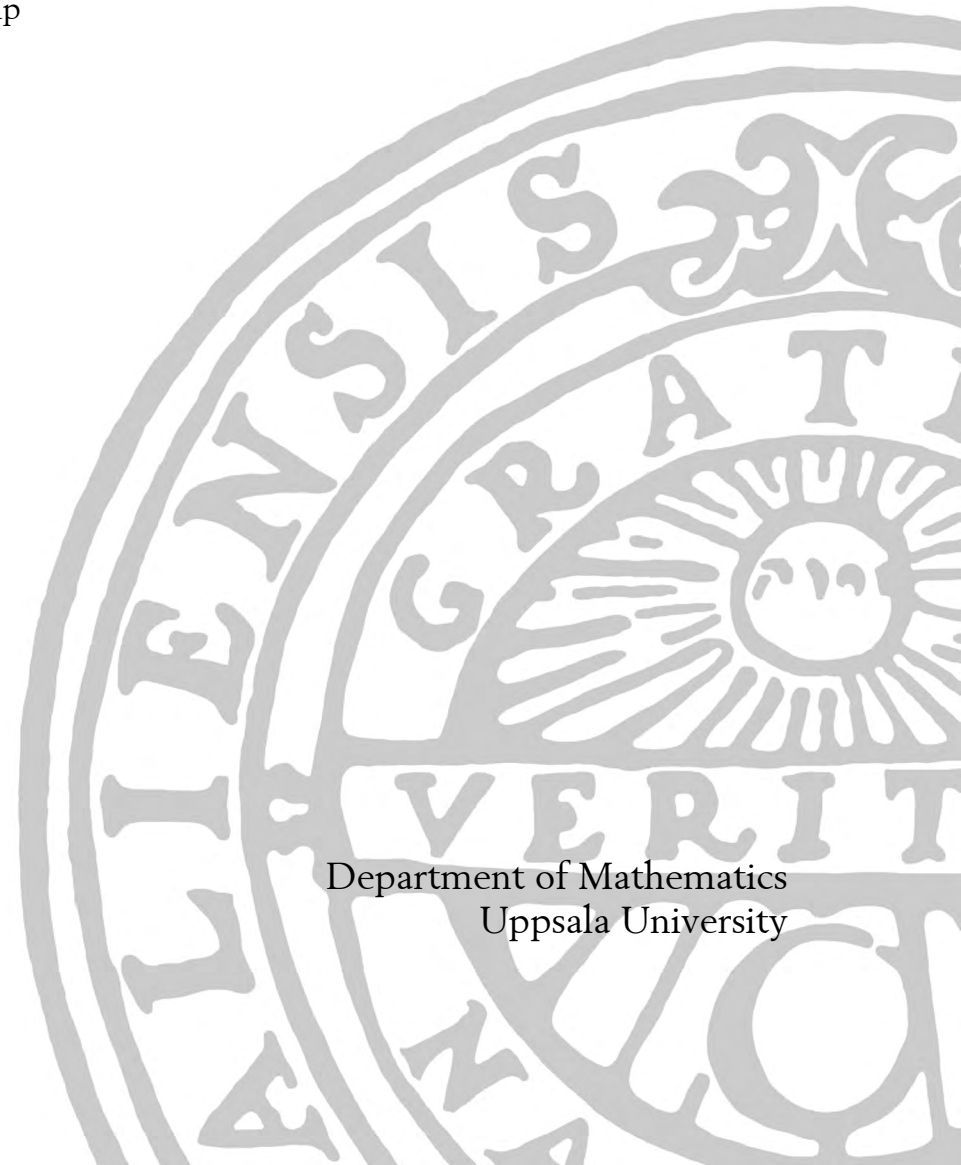
UPPSALA
UNIVERSITET

U.U.D.M. Project Report 2017:36

Fokker Planck for the Cox-Ingersoll-Ross Model

Teodor Fredriksson

Examensarbete i matematik, 15 hp
Handledare: Elisabeth Larsson
Ämnesgranskare: Erik Ekström
Examinator: Jörgen Östensson
Oktober 2017



Department of Mathematics
Uppsala University

Fokker Planck for the Cox-Ingersoll-Ross Model

Teodor Fredriksson

October 6, 2017

Abstract

In finance, it is of most importance to study and model interest rates. In the industry it is essential to estimate the parameters of a given model to get an good forecast of future interest rates and pricing. In this text we consider the Cox-Ingersoll-Ross model and we use the Radial Basis Function (RBF) approximation method to approximate the probability density function which is found by solving a Fokker-Planck Equation. Then the Maximum Likelihood method is used to estimate the parameters. A numerical error analysis procedure is then implemented in which we vary the number of points in the discretisation, the variance in the RBFs and the shape of the initial condition.

Contents

1 Introduction

In Finance, one of the most important variables is the interest rate. I some cases it might be constant at all times and in some other it might be defined as a function depending on time, in other words it's deterministic. The third possibility would be a stochastic interest rate, which means that it is random.

When the interest rate is random, it is modelled through what we call a Stochastic Differential Equation. It differs from an ODE in the sense that it contains a random component as well as the time parameter.

The Cox Ingersoll Ross model is an interest rate model that constitutes an Affine Term Structure, which means that the bond price at some time before maturity can be calculated as exponential functions.

An maximum likelihood estimator is then used to find the parameter for which the probability density attains it's maximum. One can use the law of total probability, Bayes theorem and the Markovian Property to get two complicated integrals to compute. Therefore we choose another approach, that is solving the Fokker-Planck Equation.

The Fokker-Planck Equation is an equation that describes the probability density function of a Stochastic Differential Equation, in this case the CIR model. This PDE can then be solved by first approximating the solution with a Radial Basis Function, this gives us an ODE depending on time t, divide the time interval into steps of different length and then use the BDF2 method for time-stepping.

The goal is to investigate how the error depends on N, the number of discretisation points, G, variance in the initial condition. and *shape*, which determines the variance of the RBFs.

2 Stochastic Processes

In this section we provide the definition of a diffusion process and the proceed with a detailed description of the specific models used in this work.

2.1 Diffusion Processes

Consider a vector process $X = (X_1, X_2, ..., X_n)^T$ where each X_i, $i = 1, 2, ..., n$ is governed by a stochastic differential equation (*SDE*) with dynamics

$$dX_i(t) = \mu_i\left(\theta, t, X_i(t)\right)\, dt + \sigma_i\left(\theta, t, X_i(t)\right)\, dW_i(t) \tag{1}$$

where $W_1, W_2, ..., W_d$ are independent Wiener Processes and $\mu = \mu(t) \in \mathbb{R}^n$ is the **drift vector** and $\sigma = \sigma(t) \in \mathbb{R}^{n\times d}$ is the **diffusion matrix**

$$\mu = \begin{bmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \mu_n \end{bmatrix}, \quad \sigma = \begin{bmatrix} \sigma_{11} & \sigma_{12} & \dots & \sigma_{1d} \\ \sigma_{21} & \sigma_{22} & \dots & \sigma_{2d} \\ \vdots & \vdots & & \vdots \\ \sigma_{1n} & \sigma_{2n} & \dots & \sigma_{nd} \end{bmatrix},$$

and the multidimensional Wiener Process

$$W(t) = \begin{bmatrix} W_1(t) \\ \vdots \\ W_d(t) \end{bmatrix},$$

Using these definitions and notations, we can rewrite (1) as

$$dX(t) = \mu(t)\, dt + \sigma(t)\, dW(t). \tag{2}$$

2.2 Interest rate models

The interest market data that we observe is zero coupon bond prices $\{p(\cdot, T); T \geq 0\}$.These prices depend on an underlying interest rate processes $r(t)$.

Suppose that $r(t)$ denotes the rate of interest,with dynamics

$$dr(t) = \mu(t, r(t))\, dt + \sigma(t, r(t))\, dW(t).$$

and money account $B(t)$ with dynamics

$$dB(t) = r(t)B(t)\, dt.$$

Assume now that there exists zero-coupon bonds for every maturity T and that the market is arbitrage-free and that the price process of a T-bond is on the form

$$p(t, T) = F(t, r(t); T)$$

where F is a smooth function of three variables. Furthermore F satisfies the boundary value problem

$$\begin{aligned} F_t + (\mu - \lambda\sigma)\, F_r + \frac{1}{2}\sigma^2 F_{rr} - rF = 0, \\ F(T, r) = 1, \end{aligned}$$

and F has stochastic representation

$$F(t, r; T) = \mathbb{E}^Q_{t,r}\left(\exp\left(-\int_t^T r(s)\, ds\right)\right),$$

where the Q-dynamics are given by

$$\begin{aligned} dr(s) &= (\mu - \lambda\sigma)\ ds + \sigma dW(s), \qquad s \geq t, \\ r(t) &= r_t. \end{aligned}$$

Here

$$\begin{aligned} \lambda &= \frac{\alpha_T - r}{\sigma_T}, \\ \alpha_T &= \frac{F_t + \mu F_r + \frac{1}{2}\sigma^2 F_{rr}}{F}, \\ \sigma_T &= \frac{\sigma F_r}{F}. \end{aligned}$$

2.2.1 Affine Term Structures

A process is said to posses an **affine term structure** if

$$p(t, T) = F(t, r(t); T),$$

where F has the form

$$F(t, r(t), T) = \exp\left(A(t, T) - B(t, T)r(t)\right).$$

Both the Vasiek and the Cox-Ingersoll-Ross models constitute an affine term structure.

2.2.2 The Cox-Ingersoll-Ross Model

The Cox-Ingersoll-Ross model (CIR model) is given by the one-dimensional dynamics

$$dr(t) = a\left(b - r(t)\right)\, dt + \sigma\sqrt{r(t)}\, dW(t), \quad a, b, \sigma \in \mathbb{R} \tag{3}$$

The process r_t is guaranteed not to reach zero if it satisfies the so called *Feller condition*[4]

$$\sigma^2 < 2ab.$$

The model constitutes an Affine Term Structure and the price of a T - bond is therefore given by

$$P(t,T) = \exp\left(A(t,T) - B(t,T)r_t\right),$$

where

$$\begin{aligned} A(t,T) &= \frac{2ab}{\sigma}\log\left(\frac{2\gamma\exp\left(\left(a+\gamma\right)\left(T-t\right)/2\right)}{2\gamma + \left(a+\gamma\right)\left(\exp\left(\left(T-t\right)\gamma\right)-1\right)}\right), \\ B(t,T) &= \frac{2\left(\exp\left(\left(T-t\right)\gamma\right)-1\right)}{2\gamma + \left(a+\gamma\right)\left(\exp\left(\left(T-t\right)\gamma\right)-1\right)}, \\ \gamma &= \sqrt{a^2+2\sigma^2}. \end{aligned}$$

A reason why this model has been popular is because it won't generate a negative interest rate. However, for the contemporary market situation, the possibility of negative interest rates should be included and other models might be more relevant.

2.2.3 The distribution of CIR

The distribution of a CIR process is given by

$$r_{t+T} = \frac{Y}{2c},$$

where Y is a non-central χ^2 distribution with $k = 4ab/\sigma^2$ degrees of freedom and non-centrality parameter $\lambda = 2cr_te^{-aT}$ and the pdf is given by

$$f(r_{t+T}; r_t, a, b, \sigma^2) = ce^{-u-v}\left(\frac{v}{u}\right)^{\frac{q}{2}} I_q\left(2\sqrt{uv}\right),$$

where

$$\begin{aligned} c &= \frac{2a}{\left(1-e^{-aT}\right)\sigma^2}, \\ q &= \frac{2ab}{\sigma^2} - 1, \\ u &= cr_te^{-aT}, \\ v &= cr_{t+T}, \end{aligned}$$

and $I_q(r)$ is the Bessel function of first kind, given by

$$I_q(r) = \left(\frac{r}{2}\right)^q \sum_{j=0}^{\infty} \frac{\left(r^2/4\right)^j}{j!\Gamma\left(q+j+1\right)}.$$

The expectaion and variance is given by

$$\mathbb{E}\left(r(t) \mid r(s)\right) = r(s)e^{-a(t-s)} + b\left(1 - e^{-a(t-s)}\right), \tag{4}$$

$$\mathrm{Var}\left(r(t) \mid r(s)\right) = r(s)\frac{\sigma^2}{a}\left(e^{-a(t-s)} - e^{-2a(t-s)}\right) + \frac{b\sigma^2}{2a}\left(1 - e^{-a(t-s)}\right)^2, \tag{5}$$

3 Fokker-Planck Equation

The Fokker-Planck equation for the CIR model reads

$$\frac{\partial p}{\partial t} + \left(a\left(b-r\right) - \sigma^2\right)\frac{\partial p}{\partial r} - \frac{\sigma^2}{2}\frac{\partial^2 p}{\partial r^2} - ap = 0, \quad r \in \Omega, \quad t_{n-1} \leq t \leq t_n$$
$$p\left(r, t_{n-1}\right) = p_\theta\left(r_{n-1} \mid y_{1:n-1}\right) \quad r \in \Omega$$

with boundary condition

$$\mathcal{L}p = \left(\frac{\sigma^2}{2} - a\left(b-r\right)\right)p + \frac{\sigma^2 r}{2}\frac{\partial p}{\partial r} = 0, \quad r \in \partial\Omega$$

where $\Omega \subset \mathbb{R}$ is our domain of interest in which we want to find a solution. For the purposes of this text, we have 100 data points to consider, that is a period over eight years, divided into 100 points $t_1, t_2, ..., t_{100}$, where $r_1, r_2, ..., r_{100}$ are the interest rate at each point and $y_1, y_2, ..., y_{100}$ are calculated from the measurement equation

$$y_n = Hr_n + J + \epsilon_n, \quad \epsilon_n \sim N(0, \Gamma)$$

where H is the observation matrix and J is an offset. The FP equation is solved for $n = 1, 2, ..., 100$, and for the the first interval $(n = 1)$ we have initial condition

$$p_\theta\left(r_1 \mid y_1\right) = \varphi\left(r_1, y_1, \Gamma\right)$$

where $G = G(t_n, t_{n-1}, a, b, \sigma)$ is the variance of the Radial Basis Function.

3.1 Discretization of the Fokker-Planck equation in time

We'll use the BDF2 method for time stepping whe the FP equation is solved numerically.. Divide the time interval $[t_{n-1}, t_n]$ into M time-steps of length $\tau^k = t^k - t^{k-1}$, $k = 1, ..., M$ and approximate the solution at discrete times t^k by

$$p^k(r) \approx p(r, t^k)$$

Keeping r fixed, we can think of the FP equation as an ODE with respect to the variable t

$$\frac{\partial p}{\partial t} = \mathcal{L}p$$

and then discretise it using the adaptive BDF2. For that we need to compute $p^1(r)$ from $p^0(r)$ as

$$\frac{p^1(r) - p^0(r)}{k^1} = \mathcal{L}p^1(r)$$
$$\Rightarrow p^1(r) - k^1\mathcal{L}p^1(r) = p^0(r)$$

And the for $m = 2, 3, ..., M$ we need to compute the rest of the $p^m(r)$'s from

$$p^m(r) - \beta_1^m p^{m-1}(r) + \beta_2^m p^{m-2}(r) = \beta_0^m \mathcal{L}p^m(r)$$
$$\Rightarrow p^m(r) - \beta_0^m \mathcal{L}p^m(r) = \beta_1^m p^{m-1}(r) - \beta_2^m p^{m-2}(r)$$

So in conclusion the discretised problem is as follows

$$p^1(r) - k^1\mathcal{L}p^1(r) = p^0(r), \qquad r \in \Omega \tag{6}$$
$$p^m(r) - \beta_0^m \mathcal{L}p^m(r) = \beta_1^m p^{m-1}(r) - \beta_2^m p^{m-2}(r), \quad r \in \Omega, \quad m = 2, 3, ..., M \tag{7}$$
$$\mathcal{L}^b p^m(r) = 0, \qquad r \in \partial\Omega, \quad m = 1, 2, ..., M \tag{8}$$
$$p^0(r) = p_\theta(r_{n-1} \mid y_{1:n-1}), \qquad r \in \Omega \tag{9}$$

where the coefficients $\beta_0^m, \beta_1^m, \beta_2^m$, are chosen to achieve the best computational procedure as possible.

$$\omega_m = \frac{k^m}{k^{m-1}} \tag{10}$$

$$\beta_0^m = k^m \frac{1+\omega_m}{1+2\omega_m} \tag{11}$$

$$\beta_1^m = \frac{(1+\omega_m)^2}{1+2\omega_m} \tag{12}$$

$$\beta_2^m = \frac{\omega_m^2}{1+2\omega_m} \tag{13}$$

4 Radial Basis Function approximation in space

We wish to approximate p^k as a sum of radial basis functions

$$p^k(r) = \sum_{j=1}^{N} \lambda_{j,k} \varphi\left(r, \alpha_j, G\right)$$

where

$$\varphi(r, y, G) = \frac{1}{\sqrt{|2\pi G|}} \exp\left(-\frac{1}{2}\left(y-r\right)^T G^{-1}\left(y-r\right)\right)$$

5 Parameter Estimation

We wish to compute the parameters $\theta = (a, b, \sigma)$ From observations $Y_j = y_j$. We consider the problem when the process $\{r_t\}$ cannot be observed directly. Instead we consider the process at time points $t_1, t_2, ..., t_N$, that is at points (t, r_t). From now on we denote $r_{t_j} = r_j$. The measurement equation is given by

$$Y_n = Hr_n + J + \epsilon_n,\ \epsilon_n \sim N\left(0, \Gamma\right)$$

We will use $Y = \log P$, that is we observe the logarithm of the bond prices. From this we get $H = -B(t,T),\ J = A(t,T)$.. We use the maximum likelihood method since it gives the smallest variance possible in the sense of asymptotically unbiased estimators. The ML-estimator is defined as

$$\widehat{\theta} = \arg\max_{\theta \in \Theta} \ell\left(\theta\right) = \arg\max_{\theta \in \Theta} \log p_\theta\left(y_1, ..., y_N\right).$$

According to Bayes theorem we can write

$$\ell\left(\theta\right) = \sum_{j=1}^{N} \log p_\theta\left(y_n \mid y_{1:n-1}\right)$$

Thus we need to compute $p_\theta\left(y_n \mid y_{1:n-1}\right)$ for $j = 1, 2, ..., N$. According to the theory

$$p_\theta\left(y_n \mid y_{1:n-1}\right) = \int p_\theta\left(y_n \mid r_n\right) p_\theta\left(r_n \mid y_{1:n-1}\right)\, dr_n$$

where $p_\theta\left(y_n \mid r_n\right)$ is the local likelihood for the observation y_n and $p\left(r_n \mid y_{1:n-1}\right)$ is the prediction density for the diffusion process. Now, for $j = 2, ..., N$ we do the following

1. Compute $p_\theta\left(r_n \mid y_{1:n-1}\right)$ from $p_\theta\left(r_{n-1} \mid y_{1:n-1}\right)$ by solving the FP equation using numerical integration. For the prediction density we have that

$$\begin{aligned} p_\theta\left(r_n \mid y_{1:n-1}\right) &= \int p_\theta\left(r_n, r_{n-1} \mid y_{1:\,n-1}\right)\, dr_{n-1} \\ &= \int p_\theta\left(r_n \mid r_{n-1}\right) p_\theta\left(r_{n-1} \mid y_{1:n-1}\right)\, dr_{n-1} \\ &= \int p_\theta\left(r_n \mid r_{n-1}, y_{1:n-1}\right) p_\theta\left(r_{n-1} \mid y_{1:n-1}\right)\, dr_{n-1} \end{aligned}$$

This integral is quite complicated to calculate analytically so we solve the Fokker Planck equation numerically instead.

2. Compute $p_\theta(y_n \mid r_n)$ and $p_\theta(y_n \mid y_{1:\,n-1})$ from $p_\theta(r_n \mid y_{1:n-1})$.

   We need to do this step obtain $p(y_n \mid y_{n-1})$ which we need to update the likelihood function. $p(r_n \mid y_{1:\,n})$ is used needed for the next prediction step. Using Bayes formula we find the equation

$$p_\theta(r_n \mid y_{1:n}) = \frac{p_\theta(y_n \mid r_n)p_\theta(r_n \mid y_{1:n-1})}{p_\theta(y_n \mid y_{1:n-1})} \tag{14}$$

   and the law of total probability allows us to find

$$p_\theta(y_n \mid y_{1:n}) = \int p_\theta(y_n \mid r_n)p_\theta(r_n \mid y_{1:n-1})\, dr_n \tag{15}$$

   Plugging (15) into (14) gives

$$p_\theta(r_n \mid y_{1:n}) = \frac{p_\theta(y_n \mid r_n)p_\theta(r_n \mid y_{1:n-1})}{\int p_\theta(y_n \mid r_n)p_\theta(r_n \mid y_{1:n-1})\, dr_n} \tag{16}$$

3. Update $\ell(\theta)$

6 Experiments

6.1 Layout of Experiment

We want to solve the Fokker-Planck equation on one interval in order to study the properties of the method. We are using the initial condition

$$p(r, t_{n-1}) = \frac{1}{\sqrt{2\pi G}} e^{\frac{(r-r_0)^2}{2G}}.$$

but the Fokker-Planck equation has no analytic solution for this condition. Therefore we try to approximate it with the analytic probability density function of the CIR process instead we find the corresponding T_1, which is implemented by the Matlab function `cirpdf` that is

$$p(r, t_{n-1}) \approx \text{cirpdf}(r, T_1, r_0, 0, a, b, \sigma).$$

This works, but still it gives a large error in the solution! Therefore we use `cirpdf` in order to be able to compare and analyse the error, both in the analytical and numeric solutions! Here we denote the probability density function of a Cox-Ingersoll-Ross distribution with parameters a, b, σ on the time interval $[t_0, t_1]$ by

$$\text{cirpdf}(r, t_1, r_0, t_0, a, b, \sigma).$$

where r_0 is the starting value of the process. The algorithm we use is as follows:

1. Choose initial guess $\theta_0 = (a, b, \sigma)$,variance of the RBFs G of the initial condition, number of nodes N and time interval (t_0, t_1) in which we solve the Fokker-Planck equation.

2. Solve for T_1 in the equation

$$G = \text{Var}(r(T_1) \mid r(0))$$

   and calculate the initial condition

$$u_0 = \text{cirpdf}(x_e, T_1, r_0, 0, a, b, \sigma)$$

3. Calculate $T_2 = T_1 + t_1 - t_0$ and compare

$$u_1 = \text{cirpdf}(x_e, T_2, r_0, 0, a, b, \sigma)$$

   with the solution from the Fokker-Planck equation.

6.2 Problem Parameters:

The problem parameters are a, b, σ, $G = \mathrm{Var}(\epsilon_n)$, variance in initial condition. The critical point here is to realise how the parameters N and *shape* should be choosen to minimise the error.

6.3 Error Analysis

Now lets calculate u_0 and u_1 computed above with the ones computed using RBFs, the error measures are defined below. We define the errors as

$$\varepsilon = \|E\|_2 = \frac{\sqrt{\frac{1}{N}|u_0 - \widetilde{u}_0|^2}}{\sqrt{\frac{1}{N}|u_0|^2}}$$

$$\varepsilon = \|E\|_\infty = \frac{\max_x |u_0 - \widetilde{u}_0|}{\max_x |u_0|}$$

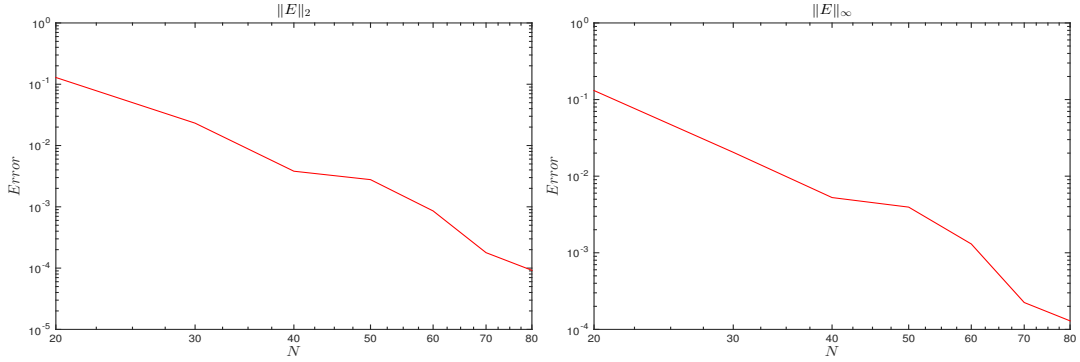
Here we put

$$x_e = \mathrm{linspace}(0, D, 4000)'$$

that is our domain of interest is $x_e \in [0, 0.2]$ and we use 4000 equidistant points, we get the results below.

6.3.1 Analysis of N

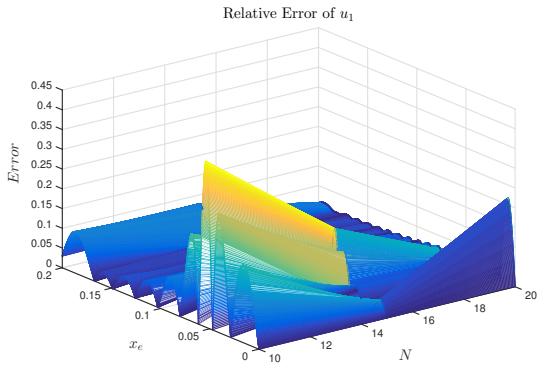
Figure 1: Errors of u_1



(a) loglog plot of error depending on N, with $G = 2.5 \cdot 10^{-8}$, $shape = 0.8$. The left picture shows the $\|E\|_2$ error and the picture to the right depicts the $\|E\|_\infty$ error.

Figure 1 illustrates that the error tends to zero as N increases.

Figure 2: Relative Errors



(a) Relative error depending on N and x_e, with $G = 2.5 \cdot 10^{-8}$, $shape = 0.8$.

From Figure 2, we notice that the error is exceptionally large around $x_e = 0.06$. This stems from the fact that the point $x_e = 0.06$ is close to the initial value $r_0 = 0.0552$.The closer x_e gets to the initial condition the larger the maximum error becomes.

Table 1: Errors of u_1

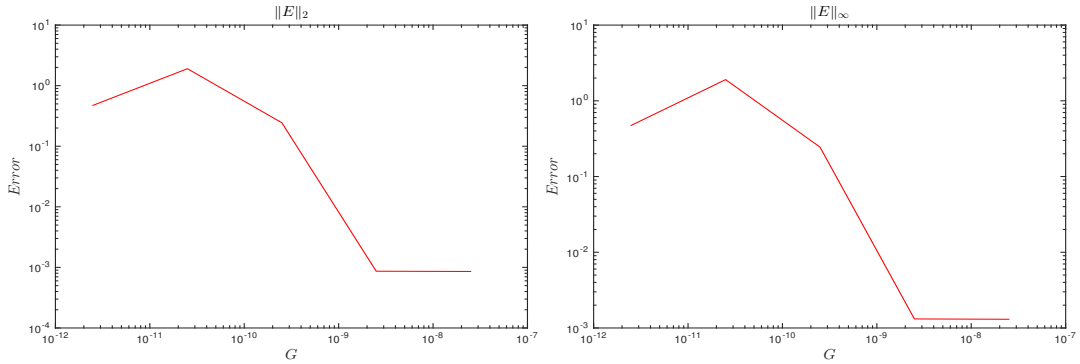
N	$\|E\|_2$	$\|E\|_\infty$
20	1.7683	6.6100
30	0.3737	1.3795
40	0.1322	0.5359
50	0.0352	0.2280
60	0.0112	0.0731
70	0.0043	0.0225
80	0.0023	0.0144

(a) Error values depending on N, with $G = 2.5 \cdot 10^{-8}$, *shape* $= 0.8$ The left column shows the $\|E\|_2$ error and the column to the right depicts the $\|E\|_\infty$ error.

Table 1 shows the actual error values. The smalles values are obtained at $N = 80$ where they are 0.0023 and 0.0144 respectively.

6.3.2 Analysis of G

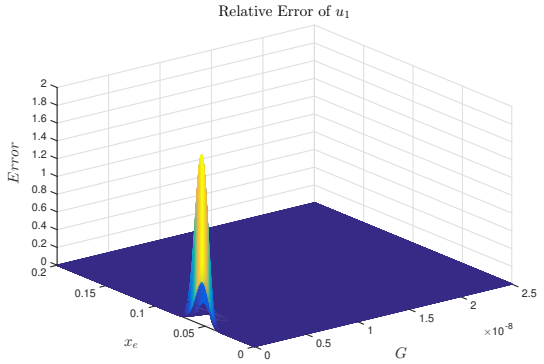
Figure 3: Errors of u_1, $G = 2.5 \cdot 10^{-d}$.



(a) loglog plot of error depending on G, with $N = 60$, *shape* $= 0.8$ The left picture shows the $\|E\|_2$ error and the picture to the right depicts the $\|E\|_\infty$ error.

Figure 3 depicts the behaviour of the error, clearly G should be small, but not too small.

Figure 4: Relative Errors, $G = 2.5 \cdot 10^{-d}$.



(a) Relative error values depending on G, with $N = 60$, $shape = 0.8$

Figure 4 confirms the same thing as Figure 2. The error is exceptionally large around $x_e = 0.06$.

Table 2: Errors of u_1

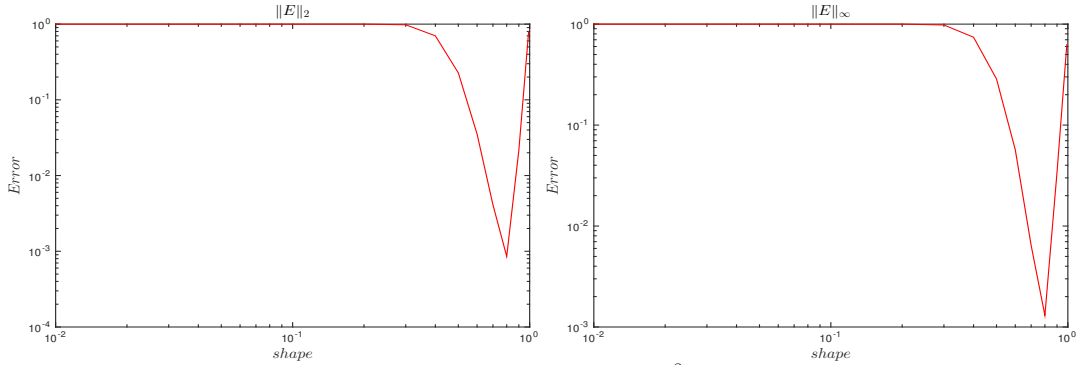
d	$\|E\|_2$	$\|E\|_\infty$
8	0.0009	0.0013
9	0.0009	0.0013
10	0.2445	0.2447
11	1.9049	1.9052
12	0.4732	0.4731

(a) Error values depending on G, with $N = 60$, $shape = 0.8$. The left column shows the $\|E\|_2$ error and the column to the right depicts the $\|E\|_\infty$ error.

Table 2 shows the actual error values. The smallest ones are for $d = 8, 9$ with 0.0009 and 0.0013 respectively.

6.3.3 Analysis of *shape*

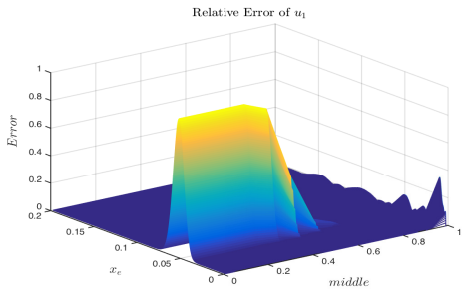
Figure 5: Errors of u_1



(a) Error values of error depending on *shape*, $N = 60$, $G = 2.5 \cdot 10^{-8}$ The left picture shows the $\|E\|_2$ error and the picture to the right depicts the $\|E\|_\infty$ error.

Figure 5 illustrates the behaviour of error, it is very clear what the value of *shape* should 0.8 be since the error is very large for most values. This means that the basis functions overlap eachother.

Figure 6: Relative Errors



(a) loglog plot of error depending on *shape*, $N = 60$, $G = 2.5 \cdot 10^{-8}$

Figure 6 shows the relative error, again just like figure 2 and 4, the error is large for $x_e \approx 0.06$

Table 3: Errors of u_1

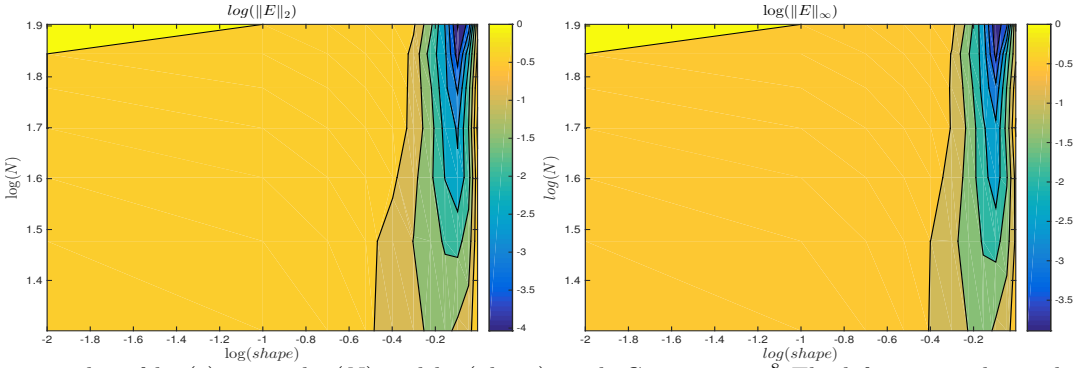
shape	$\|E\|_2$	$\|E\|_\infty$
0.01	1.0000	1.0000
0.1	1.0000	1.0000
0.2	0.9999	0.9999
0.3	0.9797	0.9799
0.4	0.7004	0.7420
0.5	0.2271	0.2861
0.6	0.0353	0.0570
0.7	0.0041	0.0064
0.8	0.0009	0.0013
0.9	0.0214	0.0340
0.99	0.7933	0.6262

(a) Error values depending on *shape*, with $N = 60$, and $G = 2.5 \cdot 10^{-8}$. The left column shows the $\|E\|_2$ error and the column to the right depicts the $\|E\|_\infty$ error.

Table 3 displays the actual error values, the smallest are when *shape* $= 0.8$ with values 0.0009 and 0.0013

6.3.4 Analysis of *shape* and N

Figure 7: Errors of u_1



(a) Contour plot of $\log(\varepsilon)$ versus $\log(N)$ and $\log(shape)$, with $G = 2.5 \cdot 10^{-8}$ The left picture shows the $\|E\|_2$ error and the picture to the right depicts the $\|E\|_\infty$ error.

In Figure 7, we notice that the errors are very large everywhere except in the interval $\log(0.7) < \log(shape) < \log(0.9)$ which can also be seen in table 5 and 6. Furthermore we can see that the order of accuracy is almost entirely independent of N i this interval. Clearly the choice of *shape* and N is quite small based on the dark blue are of the plot. it suggests that $N \geq 70$ and $shape \approx 0.8$

Table 5: Errors of $\|E\|_2$

$N \backslash shape$	0.0100	0.1000	0.2000	0.3000	0.4000	0.5000	0.6000	0.7000	0.8000	0.9000	0.9900
60	1.0000	1.0000	0.9999	0.9797	0.7004	0.2271	0.0353	0.0041	0.0009	0.0214	0.7933
61	1.0000	1.0000	1.0000	0.9893	0.7672	0.2894	0.0533	0.0055	0.0006	0.0155	0.5103
62	1.0000	1.0000	1.0000	0.9837	0.7021	0.2044	0.0259	0.0037	0.0002	0.0005	0.1953
63	1.0000	1.0000	1.0000	0.9837	0.6930	0.1857	0.0203	0.0039	0.0005	0.0140	0.3783
64	1.0000	1.0000	1.0000	0.9924	0.7872	0.2933	0.0496	0.0048	0.0005	0.0169	0.4934
65	1.0000	1.0000	1.0000	0.9940	0.7976	0.2945	0.0481	0.0046	0.0003	0.0082	0.6963
66	1.0000	1.0000	1.0000	0.9909	0.7387	0.2035	0.0195	0.0038	0.0002	0.0051	0.7202
67	1.0000	1.0000	1.0000	0.9934	0.7769	0.2500	0.0299	0.0034	0.0003	0.0134	0.3320
68	1.0000	1.0000	1.0000	0.9973	0.8469	0.3441	0.0597	0.0053	0.0003	0.0118	0.6417
69	1.0000	1.0000	1.0000	0.9966	0.8190	0.2863	0.0372	0.0035	0.0001	0.0025	0.3121
70	1.0000	1.0000	1.0000	0.9958	0.7913	0.2302	0.0188	0.0039	0.0002	0.0075	0.3900
71	1.0000	1.0000	1.0000	0.9979	0.8505	0.3256	0.0469	0.0038	0.0002	0.0113	0.3953
72	1.0000	1.0000	1.0000	0.9989	0.8819	0.3709	0.0612	0.0051	0.0002	0.0071	0.2949
73	1.0000	1.0000	1.0000	0.9982	0.8438	0.2817	0.0264	0.0033	0.0001	0.0014	0.3227
74	1.0000	1.0000	1.0000	0.9985	0.8503	0.2865	0.0259	0.0033	0.0002	0.0080	0.4083
75	1.0000	1.0000	1.0000	0.9994	0.9035	0.3946	0.0636	0.0050	0.0002	0.0085	0.3427
76	1.0000	1.0000	1.0000	0.9995	0.9026	0.3771	0.0537	0.0040	0.0001	0.0032	0.1510
77	1.0000	1.0000	1.0000	0.9993	0.8762	0.2972	0.0212	0.0037	0.0001	0.0035	0.3615
78	1.0000	1.0000	1.0000	0.9996	0.9036	0.3629	0.0432	0.0031	0.0001	0.0072	0.5697
79	1.0000	1.0000	1.0000	0.9998	0.9357	0.4433	0.0731	0.0055	0.0001	0.0056	0.6972
80	1.0000	1.0000	1.0000	0.9998	0.9193	0.3757	0.0415	0.0030	0.0001	0.0005	0.2540

(a) $\|E\|_2$ values depending on *shape* and N, with $G = 2.5 \cdot 10^{-8}$.

Table 6: Errors of $\|E\|_\infty$

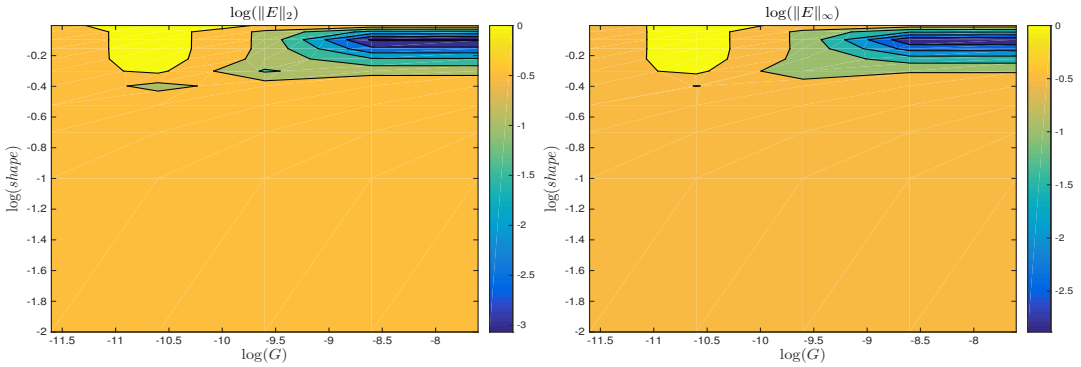
$N \backslash shape$	0.0100	0.1000	0.2000	0.3000	0.4000	0.5000	0.6000	0.7000	0.8000	0.9000	0.9900
60	1.0000	1.0000	0.9999	0.9799	0.7420	0.2861	0.0570	0.0064	0.0013	0.0340	0.6262
61	1.0000	1.0000	1.0000	0.9890	0.8044	0.3531	0.0787	0.0093	0.0009	0.0248	0.3109
62	1.0000	1.0000	1.0000	0.9888	0.7631	0.2781	0.0445	0.0054	0.0002	0.0007	0.1996
63	1.0000	1.0000	1.0000	0.9879	0.7506	0.2548	0.0327	0.0067	0.0007	0.0226	0.2630
64	1.0000	1.0000	1.0000	0.9917	0.8082	0.3513	0.0740	0.0082	0.0008	0.0275	0.5028
65	1.0000	1.0000	1.0000	0.9944	0.8335	0.3592	0.0718	0.0081	0.0004	0.0134	0.6126
66	1.0000	1.0000	1.0000	0.9940	0.7926	0.2769	0.0341	0.0062	0.0002	0.0083	0.9246
67	1.0000	1.0000	1.0000	0.9945	0.8147	0.3118	0.0490	0.0052	0.0005	0.0222	0.2212
68	1.0000	1.0000	1.0000	0.9970	0.8626	0.4002	0.0843	0.0089	0.0004	0.0196	0.7073
69	1.0000	1.0000	1.0000	0.9971	0.8538	0.3529	0.0589	0.0058	0.0002	0.0042	0.2484
70	1.0000	1.0000	1.0000	0.9973	0.8337	0.3002	0.0332	0.0063	0.0002	0.0125	0.2539
71	1.0000	1.0000	1.0000	0.9980	0.8711	0.3796	0.0698	0.0065	0.0003	0.0189	0.4093
72	1.0000	1.0000	1.0000	0.9988	0.8963	0.4267	0.0849	0.0086	0.0002	0.0120	0.1969
73	1.0000	1.0000	1.0000	0.9987	0.8756	0.3496	0.0459	0.0050	0.0002	0.0023	0.2806
74	1.0000	1.0000	1.0000	0.9990	0.8794	0.3502	0.0437	0.0055	0.0002	0.0135	0.3898
75	1.0000	1.0000	1.0000	0.9994	0.9116	0.4436	0.0873	0.0082	0.0002	0.0144	0.2924
76	1.0000	1.0000	1.0000	0.9995	0.9171	0.4342	0.0766	0.0070	0.0002	0.0056	0.1582
77	1.0000	1.0000	1.0000	0.9995	0.9022	0.3640	0.0395	0.0059	0.0002	0.0061	0.4351
78	1.0000	1.0000	1.0000	0.9997	0.9204	0.4185	0.0647	0.0045	0.0002	0.0123	0.4800
79	1.0000	1.0000	1.0000	0.9998	0.9364	0.4902	0.0965	0.0088	0.0002	0.0096	0.4539
80	1.0000	1.0000	1.0000	0.9998	0.9333	0.4349	0.0633	0.0043	0.0001	0.0008	0.3372

(a) $\|E\|_\infty$ values depending on *shape* and N, with $G = 2.5 \cdot 10^{-8}$.

Table 5 and 6 show the actual error values, the smallest errors are 0..0001 and occurs when $N = 80$ and $shape = 0.8$.

6.3.5 Analysis of G and *shape*

Figure 8: Errors of u_1



(a) Contour plot of $\log(\varepsilon)$ depending on $\log(G)$ and $\log(shape)$, with $N = 60$ The left picture shows the $\|E\|_2$ error and the picture to the right depicts the $\|E\|_\infty$ error.

With regard to what we know about the parameters in figures 1-7, we neglect examining small values of G, due to the fact that the errors are large for these small G. In figure 8 we see that the error are very big especially for small G and large *shape*, especially form $\log(shape) < -0.4$. Therefore we do not examine these values. We conclude that $\log(shape) = -0.2$ is the best choice for a minimal error, as can be seen

in figure 5 and table 3. For $\log(shape) = 0.2$ we can also see that the order of accuracy is independent of G, when $G > -9$

Table 7: Errors of $\|E\|_2, \quad G = 2.5 \cdot 10^{-d}$.

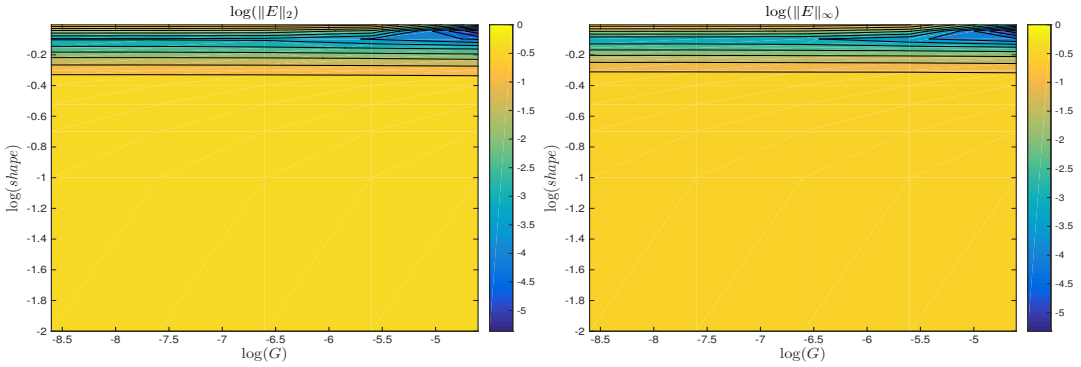
$shape \backslash d$	8	9	10	11	12
0.01	1.0000	1.0000	1.0000	1.0000	1.0000
0.1	1.0000	1.0000	1.0000	1.0000	1.0000
0.2	0.9999	0.9999	0.9999	0.9998	1.0000
0.3	0.9797	0.9798	0.9748	0.9413	0.9893
0.4	0.7004	0.7007	0.6283	0.2125	0.8413
0.5	0.2271	0.2275	0.0872	1.2895	0.5885
0.6	0.0353	0.0355	0.2145	1.8312	0.4870
0.7	0.0041	0.0042	0.2434	1.9022	0.4737
0.8	0.0009	0.0009	0.2445	1.9049	0.4732
0.9	0.0214	0.0216	0.2460	1.9060	0.4733
0.99	0.7933	0.8046	0.8702	2.6121	0.6321

Table 8: Errors of $\|E\|_\infty$, $G = 2.5 \cdot 10^{-d}$.

$shape \backslash d$	8	9	10	11	12
0.01	1.0000	1.0000	1.0000	1.0000	1.0000
0.1	1.0000	1.0000	1.0000	1.0000	1.0000
0.2	0.9999	0.9999	0.9999	0.9998	1.0000
0.3	0.9799	0.9799	0.9753	0.9476	0.9891
0.4	0.7420	0.7422	0.6825	0.3075	0.8581
0.5	0.2861	0.2866	0.1259	1.3156	0.6150
0.6	0.0570	0.0573	0.2237	1.8539	0.4936
0.7	0.0064	0.0065	0.2464	1.9085	0.4730
0.8	0.0013	0.0013	0.2447	1.9052	0.4731
0.9	0.0340	0.0343	0.2445	1.9049	0.4732
0.99	0.6262	0.6304	0.7035	1.8059	0.5077

Figure 7 and 8 shows the actual error values, they are 0.009 and 0.0013 respectively and occurs at $shape = 0.8$ and $d = 8, 9$

Figure 9: Errors of u_1 The left picture shows the $\|E\|_2$ error and the picture to the right depicts the $\|E\|_\infty$ error.

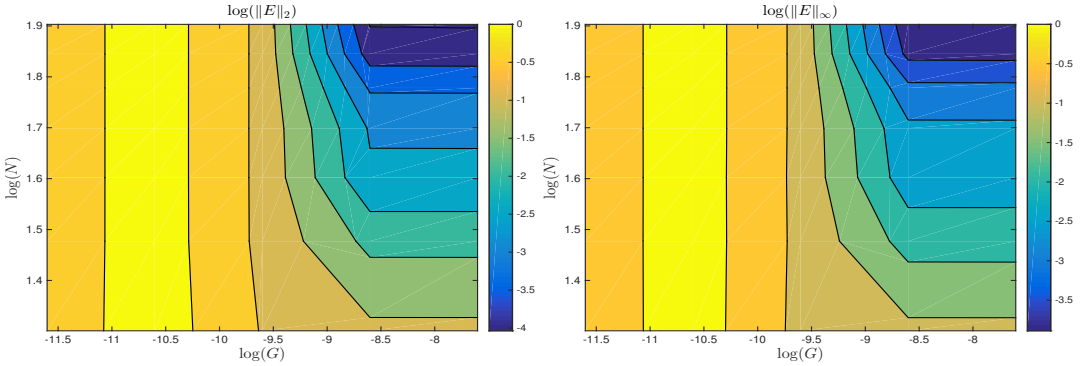


(a) Contour plot of $\log(\varepsilon)$ depending on $\log(shape)$ and $\log(G)$, with $N = 60$ The left picture shows the $\|E\|_2$ error and the picture to the right depicts the $\|E\|_\infty$ error.

In Figure 9 we se that the order of accuracy is constant , that is when G increases, we have a constant slope on the contour, except at large values of G when $\log(shape)$ is somewhere around -0.2, which means that one should choose G and $shape$ to be somewhere in this area.

6.3.6 Analysis of G and N

Figure 10: Errors of u_1



(a) Contour plot of $\log(\varepsilon)$ depending on $\log(G)$ and $\log(N)$, with *shape* $= 0.8$ The left picture shows the $\|E\|_2$ error and the picture to the right depicts the $\|E\|_\infty$ error.

From Figure 10 we see that the order of accuracy is independent of N for small G, however, for intermediate, and larger G , the order of accuracy becomes independent of G and decreases as N decreases.

Table 9: Errors of $\|E\|_2$, $\quad G = 2.5 \cdot 10^{-d}$.

$N\backslash d$	8	9	10	11	12
60	0.0009	0.0009	0.2445	1.9049	0.4732
61	0.0006	0.0006	0.2444	1.9047	0.4732
62	0.0002	0.0002	0.2446	1.9050	0.4732
63	0.0005	0.0005	0.2446	1.9052	0.4732
64	0.0005	0.0005	0.2445	1.9048	0.4732
65	0.0003	0.0003	0.2445	1.9048	0.4732
66	0.0002	0.0002	0.2446	1.9052	0.4732
67	0.0003	0.0003	0.2446	1.9050	0.4732
68	0.0003	0.0003	0.2444	1.9047	0.4732
69	0.0001	0.0001	0.2445	1.9049	0.4732
70	0.0002	0.0002	0.2446	1.9052	0.4732
71	0.0002	0.0002	0.2445	1.9049	0.4732
72	0.0002	0.0002	0.2444	1.9047	0.4732
73	0.0001	0.0001	0.2446	1.9051	0.4732
74	0.0002	0.0002	0.2446	1.9051	0.4732
75	0.0002	0.0002	0.2445	1.9048	0.4732
76	0.0001	0.0001	0.2445	1.9048	0.4732
77	0.0001	0.0001	0.2446	1.9052	0.4732
78	0.0001	0.0001	0.2446	1.9050	0.4732
79	0.0001	0.0001	0.2444	1.9047	0.4732
80	0.0001	0.0001	0.2446	1.9050	0.4732

Table 10: Errors of $\|E\|_\infty$, $G = 2.5 \cdot 10^{-d}$.

$N\backslash d$	8	9	10	11	12
60	0.0013	0.0013	0.2447	1.9052	0.4731
61	0.0009	0.0009	0.2445	1.9048	0.4732
62	0.0002	0.0002	0.2444	1.9046	0.4733
63	0.0007	0.0007	0.2446	1.9050	0.4732
64	0.0008	0.0008	0.2446	1.9051	0.4732
65	0.0004	0.0004	0.2445	1.9048	0.4732
66	0.0002	0.0002	0.2445	1.9048	0.4732
67	0.0005	0.0005	0.2446	1.9051	0.4732
68	0.0004	0.0004	0.2445	1.9049	0.4732
69	0.0002	0.0002	0.2445	1.9048	0.4732
70	0.0002	0.0002	0.2445	1.9050	0.4732
71	0.0003	0.0003	0.2446	1.9050	0.4732
72	0.0002	0.0002	0.2445	1.9048	0.4732
73	0.0002	0.0002	0.2445	1.9049	0.4732
74	0.0002	0.0002	0.2446	1.9050	0.4732
75	0.0002	0.0002	0.2445	1.9049	0.4732
76	0.0002	0.0002	0.2445	1.9049	0.4732
77	0.0002	0.0002	0.2446	1.9050	0.4732
78	0.0002	0.0002	0.2445	1.9049	0.4732
79	0.0002	0.0002	0.2445	1.9048	0.4732
80	0.0001	0.0001	0.2446	1.9050	0.4732

Table 9 and 10 shows the actual error values depending on N and G, The smallest errors are 0.0001 and occur for $N = 80$ and $d = 8, 9$.

7 Discussion

We must choose N with respect to G. Here, G is the variance of ask − bid. In terms of computational efficiency, small values of G are troublesome since we then must choose a large value of N.

In the solution step, for the the LU-decomposition, the number of operations required to solve the matrix equations can be computed as $\mathcal{O}(n^3)$, whereas in the time step of the BDF2 we have $\mathcal{O}(n^2)$.This means that we should n sufficiently small in order to minimise the computation time while still getting a small error. Furthermore, the Radial Basis Function Method approach requires less discretisation points that the Finite Difference Method. One can observe that we can get a decent estimation as long as we keep N large.

8 Appendix

8.1 Non-central χ^2 distribution

Let $X_1, X_2, ..., X_k$ be independent random variables with $X_k \sim N(\mu_k, 1)$ Then the random variable

$$Y = \sum_{i=1}^{k} X_i^2 \sim \chi^2(k, \lambda)$$

where k is the degrees of freedom and λ is the non-central parameter define by

$$\lambda := \sum_{i=1}^{k} \mu_i^2$$

is said to have a **non-central χ^2 - distribution**, denoted $\chi'^2(k,\lambda)$. This distribution has expectation and variance defined by

$$\mathbb{E}(Y) = k + \lambda$$
$$\mathrm{Var}(Y) = 2k + 4\lambda$$

The pdf is given by

$$f(x;k,\lambda) = \frac{1}{2}e^{\frac{x+\lambda}{2}}\left(\frac{x}{\lambda}\right)^{\frac{k}{4}-\frac{1}{2}} I_{\frac{k}{2}-1}\left(\sqrt{\lambda x}\right)$$

where $I_{\frac{k}{2}-1}(r)$ is the Bessel function of first kind, given by

$$I_q(x) = \left(\frac{x}{2}\right)^q \sum_{j=0}^{\infty} \frac{\left(x^2/4\right)^j}{j!\Gamma(q+j+1)}$$

References

[1] Thomas Björk, *Arbitrage Theory in Continous Time*, Oxford University Press , 3rd edition, 2009.

[2] Damiano Brigo,Fabio Mercurio, *Interest Models - Theory and Practice*, Springer Finance , 2nd edition, 2006.

[3] Lancaster, *The Chi-squared distribution*, Wiley Publications i Statistics , 1st edition, 1913.

[4] William Feller, *Two Singular Diffusion Problems*, Annals of Mathematics , 1st edition, 1951.

[5] Josef Höök, Elisabeth Larsson, Erik Lindström, Lina von Sydow *Filtering and parameter estimation of partially observed diffusion processes using Gaussian RBFs*, Not yet published , 2015

[6] E Hairer, S P Nørsett, G Wanner *Solving Ordinary Differential Equations I: Nonstiff Problems* Springer Verlag , 1980