The optimal consumption problem

A numerical simulation of the value function with the presence of a random income flow

Examensarbete för kandidatexamen i matematik vid Göteborgs universitet

Kandidatarbete inom civilingenjörsutbildningen vid Chalmers

Angelica Andersson
Johanna Svensson
Jakob Karlsson
Olof Elias

Institutionen för matematiska vetenskaper
Chalmers tekniska högskola
Göteborgs universitet
Göteborg 2012

The optimal consumption problem

A numerical simulation of the value function with the presence of a random income flow

Examensarbete för kandidatexamen i matematisk statistik inom matematikprogrammet vid Göteborgs universitet

Johanna Svensson

Kandidatarbete i matematik inom civilingenjörsprogrammet Teknisk fysik vid Chalmers

Angelica Andersson

Kandidatarbete i matematik inom civilingenjörsprogrammet Teknisk matematik vid Chalmers

Olof Elias Jakob Karlsson

Handledare: Dmitry Zhelezov
Examinator: Carl-Henrik Fant

Institutionen för matematiska vetenskaper
Chalmers tekniska högskola
Göteborgs universitet
Göteborg 2012

Sammanfattning

I denna uppsats används två metoder för att lösa problemet optimal konsumtion. Problemet är välkänt inom finansiell matematik och är i sin ursprungliga form löst av Robert Merton. Denna rapport betraktar en utvidgning med ett slumpmässigt inkomstflöde. Problemet löses approximativt med hjälp av två numeriska metoder, den ena använder Markovkedjor medan den andra ansätter en oändlig serieutveckling. Metoden med Markovkedjor är en generell metod utvecklad för stokastisk kontrollteori medan metoden som ansätter en oändlig serieutveckling är en metod som bara går att använda för att lösa vissa specifika problem. I uppsatsen implementeras och jämförs de två metoderna med hjälp av MATLAB. Metoderna tycks komplettera varandra väl men resultaten är något ofullständiga.

Abstract

In this thesis two methods are used to solve the optimal consumption problem. The optimal consumption problem is a well known problem in mathematical finance which in its original form was solved by Robert Merton. This report considers an extension with a presence of a random income flow. The problem is approximately solved using two numerical methods, the approximating Markov chain approach and the infinite series expansion. The Markov chain approach is a general method developed for stochastic control theory whereas the infinite series expansion method only can be applied to a specific set of problems. In the thesis the methods are implemented and compared using MATLAB. The methods seem to complement each other well however the results are somewhat inconclusive.

Preface

This thesis is divided into two subproblems and two groups have been working separately since the methods used differ. Angelica Andersson and Jakob Karlsson have been attempting an analytical solution using infinite series expansion and Johanna Svensson and Olle Elias have been using a Markov chain method. The thesis is written in English since the supervisor is not Swedish. The problem is solved numerically using MATLAB and all members of the group have contributed to the implementation. The group as a whole have put together the report in terms of solving the problem and analyzing the results.

The introductory chapter is mainly written by Johanna and the basic theory has been described by Angelica and Johanna. Johanna also described the vital assumptions in the economic setting whereas Olle has described the processes and the Hamilton-Jacobi-Bellman equation. The reduction of the problem and the optimal controls has Jakob provided. In the part where the infinite series expansion is described Angelica introduces the problem and described the algorithm while Jakob supplies the derivation of the solution. In the section about the Markov chain approach Johanna describes the Markov decision process and Olle has written the remaining parts. Angelica is responsible for the results about the infinite series expansion and Johanna has written the introduction and the part about the Markov chain method. In the final chapter have all four made equal contributions.

During the process a journal has been kept with details regarding the work. It also contains specific information about what has been done by whom throughout the project.

Thanks to

We would foremost like to thank our supervisor *Dmitry Zhelezov* who has made this thesis possible. We would also like to thank *Anna-Lena Fredriksson* whose advice regarding the language has been invaluable.

Contents

1 Introduction **1**
1.1 The optimal consumption problem 1

2 Stochastic processes **3**
2.1 Brownian motion 3
2.2 Markov chain 4

3 The Optimal Consumption Problem **5**
3.1 The economic setting 5
3.2 The Hamilton-Jacobi-Bellman equation 6
3.3 Reducing the problem 7

4 Numerical Methods for Solving the Problem **10**
4.1 The infinite series expansion method 10
4.1.1 Deriving an analytical solution 10
4.1.2 Algorithm 13
4.2 Markov chain approach 13
4.2.1 Markov decision process 13
4.2.2 Constructing the approximating Markov chain 14
4.2.3 The approximating Markov chain for the optimal consumption problem 16
4.2.4 The dynamic programming equation 17
4.2.5 Convergence scheme 18

5 Results **20**
5.1 Numerical results for the infinite series expansion 20
5.2 Numerical results for the Markov chain approach 25

6 Discussion **31**
6.1 Infinite Series Expansion 31
6.2 Markov chain approach 31
6.3 Comparing the methods 31

A MATLAB code **34**
A.1 The approximating Markov chain approach 34
A.2 Infinite series expansion 40

B Solving tridiagonal matrices **45**

C Additional plots for Markov chain approach **46**

Chapter 1

Introduction

1.1 The optimal consumption problem

The optimal consumption problem describes the optimal way an investor can use his money if he only has three choices; it is possible to save the money in a risk free bond, it is also possible to invest it on the risky stock market and spend the money on consumption. Robert Merton studied the case where the investor had no income flow and managed to solve it analytically [1].

To emulate the investor's decision, a way to measure the investor's preference and risk attitude will be needed. A basic concept in economics is utility theory in which there are some assumptions made about the investor's behavior. First, we assume the investor to be risk averse, which simply states that the investor will reject investments that are fair game or worse. We also assume non-satiation, that the investor always prefers more to less wealth. These assumption describe some characteristics of the utility function. Since the investor is risk averse, the utility function will be concave, implying that the marginal utility of wealth decreases as the wealth increases. The assumptions of non-satiation implies that the utility function will always be increasing. In this report we will use the utility function $U(c) = \log(c)$ where c denotes the consumption. Since the value of money is not constant over time, we need a discount factor to make the choice of time reasonable. This is set to $e^{-\beta t}$ where β represents the continuous discount factor.

The main concepts to formulate this problem mathematically has now been presented. Our purpose is to maximize the investor's expected utility during his lifetime, where an infinite time horizon is assumed. Hence we get the following objective function, which is referred as the value function,

$$\max\left\{E\left[\int_0^\infty e^{-\beta t}U(c_t)dt\right]\right\}. \tag{1.1}$$

The problem above is a version of the original Merton problem, which has a closed form solution. In this thesis the problem is generalized by assuming that the investor's income is unpredictable which makes it impossible for the investor to borrow against future income. Hence we get an incomplete market where the investor's wealth must stay positive at all times. When adding random income flow to the problem there is no closed form solution and therefore we will instead use two different numerical methods to find an approximate solution.

The first of the two methods used in this report is the infinite series expansion which was introduced by Claudio Tebaldi and Eduardo S. Schwartz in the 2000's [2]. This method is not very general, but can be used in our specific case. For the logarithmic utility function there is not much literature or work presented using infinite series expansion.

The second method is more general and was developed for stochastic control problem in the early 1990s by Harold J. Kushner and Paul Dupuis [3]. This method uses Markov chains to approximate the optimal policies and it is most commonly used when solving this type of problems. Analysis of this numerical method has been made by Munk [4] for a different utility function. In this thesis we intend to follow his work but with the logarithmic utility function.

The purpose of this thesis is to solve the generalized optimal consumption problem using the two numerical methods. We will also compare the methods and see how they can complement each other. We will investigate how the investor should behave under variation of economic parameters and deduce the optimal policy.

In this thesis we will only consider the case where we have a logarithmic utility function and an infinite time horizon. Unfortunately a lot of the underlying theory of the developed methods will be out of scope of this thesis and we will focus more on the derivation of the formulae and implementation rather than proving properties of the methods.

Chapter 2

Stochastic processes

This thesis relies heavily on the concept of stochastic processes. A stochastic process is the mathematical model of an empirical process whose development is governed by probability laws. A stochastic process according to [5] is defined as:

Definition 1 *Given an index set I, a stochastic process, indexed by I is a collection of random variables $\{X_\lambda : \lambda \in I\}$ on a probability space $(\Omega,\mathcal{F},P)$ taking values in a set S. The set S is called the state space of the process.*

Two important properties of random processes is mean square continuity and mean square differentiation which are defined below using Hwei's definition in [6].

Definition 2 *A random process is said to be mean square (m.s.) continuous if*

$$\lim_{\varepsilon\to 0} \mathbf{E}\left[(X(t+\varepsilon) - X(t))^2\right] = 0 \tag{2.1}$$

Then the m.s derivative $X'(t)$ can be defined as

Definition 3

$$\text{l.i.m.}_{\varepsilon\to 0} \frac{X(t+\varepsilon) - X(t)}{\varepsilon} = X'(t) \tag{2.2}$$

where l.i.m. *denotes limit in the mean (square), provided that*

$$\lim_{\varepsilon\to 0} \mathbf{E}\left[\left(\frac{X(t+\varepsilon) - X(t)}{\varepsilon} - X'(t)\right)^2\right] = 0 \tag{2.3}$$

2.1 Brownian motion

The most important random process for our work will be the Brownian motion, also called the Wiener process. The name Brownian motion is due to its origin as a model for the erratic movement of particles suspended in a fluid.

In order to clearly state what a Brownian motion is the concept of stationary independent increments are defined:

Definition 4 *A random process $X(t), t \geq 0$ is said to have independent increments if whenever $0 < t_1 < t_2 < ... < t_n$,*

$$X(0),X(t_1) - X(0),X(t_2) - X(t_1),...,X(t_n) - X(t_{n-1}) \tag{2.4}$$

are independent. If $X(t), t \geq 0$ has independent increments and $X(t) - X(s)$ has the same distribution as $X(t+h) - X(s+h)$ for all $s,t,h \geq 0$, $s < t$, then the process $X(t)$ is said to have stationary independent increments.

The Brownian process is characterized by the following properties [6]:

1. $X(t)$ has stationary independent increments
2. The increment $X(t) - X(s)$ $t > s$ is normally distributed
3. $\mathbf{E}[X(t)] = 0$
4. $X(0) = 0$

The Brownian motion is the most vital stochastic process since it is utilized to model the behavior of stock prices.

2.2 Markov chain

The discrete-time, discrete-space Markov process is referred to as the Markov chain. The property of Markov processes is that the probability of going one step forward in the process only depends on the last step taken, all other transitions made before are irrelevant. The formal definition is stated below.

Definition 5 *A stochastic process* $\{x_n, n = 0, 1, ...\}$ *with a discrete state space* I *is called a discrete time Markov chain if*

$$P\{x_{n+1} = i_{n+1} | x_0 = i_0, ..., x_n = i_n\} = P\{x_{n+1} = i_{n+1} | x_n = i_n\} \tag{2.5}$$

for $i_0, ..., i_{n+1} \in I$.

The transition probability of moving from state i to state j can be written as $P\{x_{n+1} = j | x_n = i\} = p_{ij}$ where $i, j \in I$. These probabilities must satisfy following conditions:

1. $p_{ij} \geq 0$ for $i, j \in I$
2. $\sum_{j \in I} p_{ij} = 1$ for $i \in I$.

To use Markov chains for modeling purposes, the first step is to choose state variables which make the Markov property in (2.5) hold. The second step is to determine the one-step transition probabilities.

A natural way of expanding the concept of Markov chains is to introduce Markov decision processes (MDP). The MDP extends the Markov chain in two ways. The process allows actions, also called controls, in each step and add rewards or costs for the chosen action. The actions are chosen from a set of allowed actions, or admissible controls.

The use of actions demand a way of controlling the actions in each step, and hence we need to introduce a policy, or rule, which describes what action to be taken. A fundamental question in Markov decision theory is whether there exists an optimal policy and how to find it. The policies are divided into classes, one of the most important classes is the stationary policies. These policies suggests the same action every time the Markov chain visits a specific state.

Chapter 3

The Optimal Consumption Problem

3.1 The economic setting

The most critical assumptions are stated already in Mertons article [1] from 1971. Two important assumption concern the behavior of asset prices and also the investor's attitude to risk. There are also some other assumptions made about the market, which should be perfect, with continuous trading possibilities and no transaction costs.

Definition 6 *If the log-price process* $\ln S(t), t \geq$ *is governed by a Brownian motion with a drift,* $S(t) = S(0)e^{\alpha t + \sigma W(t)}, t \geq 0$, *where* $\alpha > 0$ *and* $\sigma \geq 0$, *then the stock price process* $S = (S(t))_{t \geq 0}$ *is called a geometric Brownian motion.*

Assumption 1 *The behavior of asset prices in a perfect market can be described by a random walk of return, or in the continuous case, by a geometric Brownian motion.*

Assumption 1 might be the most important one and is often used in financial models. The accuracy of the assumption was questioned already when Merton presented his work [1] but it is still used in a lot of financial models. Several alternative assumptions that could be used instead have been presented throughout history of financial mathematics but none of them seem to improve assumption 1.

Definition 7 *The utility function* $U(c) : S \longrightarrow \mathbb{R}$, $S \subseteq \mathbb{R}$ *measures the investors risk attitude and preferences. The function has the following properties:* $U(c) \in C^2(\mathbb{R}_+)$ *with* $U'(c) > 0$ *(non-satiation) and* $U''(c) < 0$ *(risk aversion).*

In the model described in this report, it is assumed that the investor has a logarithmic utility function, $U(c) = \log(c)$, which fulfill the conditions in definition 7.

The model used in this thesis is an equilibrium model, in the sense that it assumes the market to be perfect. The perfect market has no transaction costs, is accessible with sufficient trading possibilities, has perfect competition and perfect information. This assumption makes the model more theoretical, since the real economy is not always in equilibrium in the short run.

The last important assumption made, is that the investor has an initial wealth endowment. This assumption is important and can be seen as a part of the problem formulation, since the problem to be solved is to allocate this wealth and the future unknown income between consumption, risky investment in the stock market and a low risk bond.

Before the problem can be formulated along with the associated Hamilton-Jacobi-Bellman equation the economic setting of this problem must be described. The setting describes a risk-free bond $B(t)$, the stock price $S(t)$, an illiquid asset $H(t)$ and a wealth process $L(t)$. The setting used is described in detail in [7].

- The risk-free bond with a constant positive interest rate r is described as:

$$dB(t) = rB(t)dt, \ t > 0.$$

- The stock price follows the geometrical Wiener process which in differential form is written as:

$$\frac{dS(t)}{S(t)} = \alpha dt + \sigma dW_1(t), \ t > 0.$$

where α ($> r$) is the continuously compounded rate of return and the standard deviation σ also referred as the volatility.

- The illiquid asset which is correlated with the stock price with correlation coefficient ρ:

$$\frac{dH(t)}{H(t)} = (\mu - \delta)dt + \eta\left(\rho dW_1(t) + \sqrt{1-\rho^2}dW_2(t)\right), \ H(0) = h, \ t > 0. \tag{3.1}$$

where μ is the expected rate of return on the risky illiquid asset, δ is the rate of dividend paid by the illiquid asset and η is the continuous standard deviation of the rate of return.

- The wealth process is fed by the holdings in bond, stock and dividends from the non-traded asset and is defined as:

$$dL(t) = (rL(t) + \delta H(t) + \pi(t)(\alpha - r) - c(t))\, dt + \pi(t)\sigma dW_1(t), \ L(0) = l, \ t > 0. \tag{3.2}$$

Note that the processes are written in differential form, this is needed since no exact solution can be found (for most cases) and also because the approximating Markov chain method relies on the processes being in this form.

3.2 The Hamilton-Jacobi-Bellman equation

In this section the original problem stated in section 1.1 will be reformulated and the Hamilton-Jacobi-Bellman equation, which from now on will be called the HJB equation, will be introduced.

By recalling (1.1) and writing it more carefully using the economic setting described in the previous section this yields the following function,

$$V(l,h) = \max_{c,\pi \in A(l,h)} \left\{ E\left[\int_0^\infty e^{-\beta t} U(c_t)dt | L(0) = l, H(0) = h\right]\right\}, \tag{3.3}$$

where $A(l,h)$ is the set of all admissible controls is described in detail in [7]. Unfortunately a rigorous definition is out of scope for this thesis.

Now to derive the HJB equation one utilizes the *Bellman's linear programming principle* which describes the infinitesimal change of the function $V(l,h)$. The actual derivation of the equation relies on Ito's formula from stochastic calculus which we will not be able to describe in this thesis but a formal derivation can be seen in [7].

$$\frac{1}{2}\eta^2 h^2 V_{hh} + (rl + \delta h)\, V_l + (\mu - \delta)\, hV_h + \max_{\pi} G(\pi) + \max_{c \geq 0} H(c) = \beta V, \tag{3.4}$$

where

$$G(\pi) = \frac{1}{2}V_{ll}\pi^2\sigma^2 + V_{lh}\eta\rho\pi\sigma h + \pi(\alpha - r)V_l(l,h), \tag{3.5}$$

$$H(c) = -cV_l + U(c) = -cV_l + \log(c). \tag{3.6}$$

Now to reduce the equation one needs to make some assumptions about the value function. To maximize $G(\pi)$ we study the behavior of the function through the second order derivative.

$$\frac{d^2}{d\pi^2}G(\pi) = V_{ll}\sigma^2$$

In order for the maximum to exist the function $G(\pi)$ must be concave thus we need to assume that $V_{ll} \leq 0$ and also that V_{ll} exists at all points which simply means that we assume that V is smooth.

3.3 Reducing the problem

Following the papers by Munk [4] and Tebaldi/Schwarz [2] a reduction of the problem is needed for the numerical methods. To reduce this problem from a PDE to an ODE the following transform will be used:

$$
\begin{aligned}
z &= \frac{l}{h}\,,\\
V(l,h) &= K + \frac{\log h}{\beta} + W(z)\,,
\end{aligned}
$$

where K is an arbitrary constant which will be set later to simplify the problem further. The derivatives of V will take on the following form,

$$
\begin{aligned}
V_h &= \frac{1}{h\beta} - \frac{l}{h^2}W' \Leftrightarrow hV_h = \frac{1}{\beta} - zW',\\
V_l &= \frac{1}{h}W' \Leftrightarrow \begin{cases} lV_l = zW' \\ hV_l = W' \end{cases},\\
V_{hh} &= -\frac{1}{h^2\beta} + \frac{2l}{h^3}W' + \frac{l^2}{h^4}W'',\\
&\Leftrightarrow h^2V_{hh} = -\frac{1}{\beta} + zW' + z^2W'',\\
V_{ll} &= \frac{1}{h^2}W'' \Leftrightarrow \begin{cases} h^2V_{ll} = W'' \\ lhV_{ll} = zW'' \\ l^2V_{ll} = z^2W'' \end{cases},\\
V_{lh} &= -\frac{1}{h^2}W' - \frac{l}{h^3}W'' \Leftrightarrow h^2V_{lh} = -W' - zW''.
\end{aligned}
$$

Looking at (3.4) the right hand side becomes:

$$
\beta V = \beta K + \log h + \beta W, \tag{3.7}
$$

for the left hand side the transformation is done in steps, starting with $\max_\pi G(\pi)$ and $\max_{c\geq 0} H(c)$

$$
\begin{aligned}
\max_{\pi} G(\pi) &= \max_{\pi}\left[\frac{1}{2}V_{ll}\pi^2\sigma^2 + V_{lh}\eta\rho\pi\sigma h + (\alpha - r)V_l\pi\right] \\
&= \max_{\pi}\left[\frac{1}{2}W''\sigma^2\frac{\pi^2}{h^2} - (W' + zW'')\eta\rho\sigma\frac{\pi}{h} + (\alpha - r)W'\frac{\pi}{h}\right] = \left\{h > 0, \quad \pi_1 = \tfrac{\pi}{h}\right\} \\
&= \max_{\pi_1}\left[\frac{1}{2}W''\sigma^2\pi_1^2 - (W' + zW'')\eta\rho\sigma\pi_1 + (\alpha - r)W'\pi_1\right] \\
&= \max_{\pi_1}\left[\frac{1}{2}W''\left(\sigma^2\pi_1^2 - 2\eta\rho z\pi\sigma\right) - W'\eta\rho\sigma\pi_1 + (\alpha - r)W'\pi_1\right] \\
&= \max_{\pi_1}\left[\frac{1}{2}W''\left(\sigma\pi_1 - \eta\rho z\right)^2 - W'\eta\rho\sigma\pi_1 + (\alpha - r)W'\pi_1\right] - \frac{\eta^2}{2}\rho^2 z^2 W'' \\
&= \left\{\varphi = \pi_1 - \tfrac{\eta\rho z}{\sigma}\right\} \\
&= \max_{\varphi}\left[\frac{1}{2}W''\sigma^2\varphi^2 + \underbrace{(-\eta\rho\sigma + \alpha - r)}_{k_1}\left(\varphi + \frac{\eta\rho z}{\sigma}\right)W'\right] - \frac{\eta^2}{2}\rho^2 z^2 W'' \\
&= \max_{\varphi}\left[\frac{1}{2}W''\sigma^2\varphi^2 + k_1\varphi W'\right] - \frac{\eta^2}{2}\rho^2 z^2 W'' + \frac{\eta\rho k_1}{\sigma}zW', \\
\max_{c\geq 0} H(c) &= \max_{c\geq 0}\left[-cV_l + \log c\right] = \max_{c\geq 0}\left[-\frac{c}{h}W' + \log c\right] = \left\{h > 0, \quad c_1 = \tfrac{c}{h}\right\} \\
&= \max_{c_1\geq 0}\left[-c_1W' + \log c_1\right] + \log h,
\end{aligned}
$$

and then the rest of the left hand side

$$
\begin{aligned}
&\frac{1}{2}\eta^2h^2V_{hh} + (rl + \delta h)V_l + (\mu - \delta)hV_h \\
&= \frac{\eta^2}{2}\left(-\frac{1}{\beta} + 2zW' + z^2W''\right) + (rz + \delta)W' + (\mu - \delta)\left(\frac{1}{\beta} - zW'\right) \\
&= \frac{\eta^2}{2}z^2W'' + \left(\eta^2 + r - (\mu - \delta)\right)zW' + \delta W' - \frac{\eta^2}{2\beta} + \frac{\mu - \delta}{\beta}.
\end{aligned}
$$

To simplify the equation even further we can first make the observation that both sides contain the term $\log h$ and hence these cancel out. Next we can remove the constants by remembering that K is just an arbitrary constant, and hence we can set it to

$$
K = \frac{\mu - \delta}{\beta^2} - \frac{\eta^2}{2\beta^2}\ , \tag{3.8}
$$

and the equation can now be written as

$$
\begin{aligned}
\frac{\eta^2}{2}\left(1 - \rho^2\right)z^2W'' + kzW' + \delta W' + \max_{\varphi}\left[\frac{1}{2}W''\sigma^2\varphi^2 + k_1\varphi W'\right] \\
+ \max_{c\geq 0}\left[-cW' + \log c\right] = \beta W,
\end{aligned}
$$

where $k = \eta^2 + r - (\mu - \delta) - \frac{\eta\rho k_1}{\sigma}$ and $k_1 = -\eta\rho\sigma + \alpha - r$. Here we make a small transformation $\zeta = c - \delta$ to get rid of the term $\delta W'$. And so we will finally end up with the reduced HJB equation:

$$
\frac{\eta^2}{2}\left(1 - \rho^2\right)z^2W'' + kzW' + \max_{\varphi} G_2(\varphi) + \max_{\zeta\geq -\delta} H_2(\zeta) = \beta W, \tag{3.9}
$$

with

$$
\begin{aligned}
G_2(\varphi) &= \frac{1}{2}W''\sigma^2\varphi^2 + k_1\varphi W', \\
H_2(\zeta) &= -\zeta W' + \log(\zeta + \delta), \\
k &= \eta^2 + r - (\mu - \delta) + \frac{\eta\rho k_1}{\sigma}, \\
k_1 &= -\eta\rho\sigma + \alpha - r.
\end{aligned}
$$

Recall that in Section 3.2 we made certain assumptions regarding the value function. Since h and l are both positive these assumptions now give us that W is smooth and that $W'' \leq 0$. This means that we can solve for the maximum of G_2 and H_2. We get that

$$\begin{aligned}
\varphi^* &= -\frac{k_1 W'}{\sigma^2 W''}, \\
\zeta^* &= \frac{1}{W'} - \delta, \\
G_2(\varphi^*) &= -\frac{k_1^2 (W')^2}{2\sigma^2 W''}, \\
H_2(\zeta^*) &= -1 + \delta W' - \log W',
\end{aligned}$$

and taking the values of φ^* and ζ^* and converting them back to the original optimal controls, π and c. By scaling with the initial wealth l they can be expressed as:

$$\frac{\pi^*}{l} = \frac{\eta\rho}{\sigma} - \frac{k_1 W'}{\sigma^2 z W''}, \tag{3.10}$$

$$\frac{c^*}{l} = \frac{1}{zW'}. \tag{3.11}$$

Chapter 4

Numerical Methods for Solving the Problem

4.1 The infinite series expansion method

In this section we will describe in detail how the optimal consumption problem can be solved using infinite series expansion. We will utilize the fact that it is possible to find an infinite series expansion that solves a transformed version of equation (3.9). Once this series expansion has been found, we will be able to return to W(z) and hence also find the optimal controls using MATLAB.

4.1.1 Deriving an analytical solution

We start off with the reduced HJB equation (3.9) which we can now write as

$$\frac{\eta^2}{2}\left(1-\rho^2\right)z^2W'' + kzW' - \frac{k_1^2}{2\sigma^2}\frac{\left(W'\right)^2}{W''} + \delta W' - \log W' - 1 = \beta W. \tag{4.1}$$

As the equation takes up a lot of space we introduce constants K_i and a function F and write our equation as

$$K_1 W + K_2 zW' + K_3 z^2 W'' + K_4 \frac{\left(W'\right)^2}{W''} + F\left(W'\right) = 0, \tag{4.2}$$

where

$$\begin{aligned}
K_1 &= -\beta, \\
K_2 &= k = \eta^2 + r - \mu + \delta + \frac{\eta\rho k_1}{\sigma} = \\
&= \eta^2 + r - \mu + \delta - \eta^2\rho^2 + \frac{\eta\rho}{\sigma}\left(\alpha - r\right), \\
K_3 &= \frac{\eta^2}{2}\left(1-\rho^2\right), \\
K_4 &= -\frac{k_1^2}{2\sigma^2} = -\frac{\left(-\eta\rho\sigma + \alpha - r\right)^2}{2\sigma^2}, \\
F\left(x\right) &= \delta x - \log x - 1.
\end{aligned}$$

If we now take a look at (4.2) we can start to see a pattern emerging where we have terms on the form $z^k W^{(k)}\left(z\right)$. This is what we will use to find the solution, so where is this pattern broken? By simple calculation we can see that

$$\frac{\left(W'\right)^2}{W''} = \frac{\left(zW'\right)^2}{z^2 W''}, \tag{4.3}$$

so the part of the equation which is making it difficult for us is the term $F(W')$. To simplify this, what we essentially want to do is a variable transformation where the new variable is W' or something similar. There is a transform known as the Legendre transform which does exactly this, the transformation acts on the function $f(x)$ as follows.

$$g(y) = \max_x \left(f(x) - xy\right),$$

which gives us the new variable $y = f'(x)$. However, the transformation does not work too well in our case. What we will instead use is a variation of this transformation which will introduce a new variable y and a new function $\widetilde{W}(y)$

$$\widetilde{W}(y) = \max_z \left(W(z) - \frac{z}{y}\right). \tag{4.4}$$

At optimality of the right hand side we see that $y = \frac{1}{W'(z)}$. The inverse of this transformation takes the form

$$W(z) = \min_y \left(\widetilde{W}(y) + \frac{z}{y}\right), \tag{4.5}$$

and we can see that at optimality we have that $z = y^2\widetilde{W}'(y)$. So now we have the following relationships

$$\begin{aligned}
y &= \frac{1}{W'(z)},\\
z &= y^2\widetilde{W}'(y),\\
W''(z) &= \frac{dW'}{dz} = \frac{d\frac{1}{y}}{dy^2\widetilde{W}'(y)} = \frac{d\widetilde{y}}{d\frac{1}{\widetilde{y}^2}\widetilde{W}'\left(\frac{1}{\widetilde{y}}\right)},\\
&= \frac{1}{-\frac{1}{\widetilde{y}^4}W''\left(\frac{1}{\widetilde{y}}\right) + 2\frac{1}{\widetilde{y}^3}\widetilde{W}'\left(\frac{1}{\widetilde{y}}\right)} = -\frac{1}{y^4\widetilde{W}'' - 2y^3\widetilde{W}'},
\end{aligned}$$

this means that the term $F(W')$ is now written as

$$F(W') = F\left(\frac{1}{y}\right) = \frac{\delta}{y} + \log y - 1,$$

which turns equation (4.2) into:

$$K_1\widetilde{W} + (K_1 + K_2 + 2K_4)\, y\widetilde{W}' - K_3\frac{(\widetilde{W}')^2}{\widetilde{W}'' - \frac{2}{y}\widetilde{W}'} - K_4 y^2\widetilde{W}'' + \frac{\delta}{y} + \log y = 1. \tag{4.6}$$

At this point we can try to find a solution to the ODE. The first step is to find a way to deal with the term $\log y$, and the easy way to do so is to set that $\widetilde{W}$ contains the term $-\frac{1}{K_1}\log y$. A reasonable guess is that the remaining terms would deal with the derivatives of $\log y$ namely y^{-k} with $k = 1,2,\ldots$ and have some constant in front of it. Let us call these constants B_k and see what happens when we assume such a solution.

$$\widetilde{W} = -\frac{1}{K_1}\log y + B_0 + \sum_{n=1}^{\infty} B_n y^{-n}, \tag{4.7}$$

$$\widetilde{W}' = -\frac{1}{K_1 y} - \sum_{n=1}^{\infty} nB_n y^{-n-1} = \sum_{n=0}^{\infty} C_n y^{-n-1}, \tag{4.8}$$

$$\widetilde{W}'' = \frac{1}{K_1 y^2} + \sum_{n=1}^{\infty} n(n+1) B_n y^{-n-2} = \sum_{n=0}^{\infty} D_n y^{-n-2}. \tag{4.9}$$

By comparing this to equation (4.6) we can now look at the individual terms y^{-k} for $k = 0,1,2,...$ to get an expression for B_k. But before we can do that we need to see what happens to the term $\frac{(\widetilde{W}')^2}{\widetilde{W}'' - \frac{2}{y}\widetilde{W}'}$.

$$\frac{\widetilde{W}'}{\widetilde{W}'' - \frac{2}{y}\widetilde{W}'} = \frac{\sum_{n=0}^{\infty} C_n y^{-n-1}}{\sum_{n=0}^{\infty} D_n y^{-n-2} - 2\sum_{n=0}^{\infty} C_n y^{-n-2}} = y\frac{\sum_{n=0}^{\infty} C_n y^{-n}}{\sum_{n=0}^{\infty} \left(D_n - 2C_n\right) y^{-n}}. \tag{4.10}$$

At this point we make the assumption that this can be written as an infinite sum $y\sum_{n=0}^{\infty} E_n y^{-n}$. This assumption is made for us to be able to use the method described above and find the terms B_k. Multiplying both sides by the denominator on the left hand side and dividing by y gives us that

$$\sum_{n=0}^{\infty} C_n y^{-n} = \left(\sum_{n=0}^{\infty} E_n y^{-n}\right)\left(\sum_{n=0}^{\infty} \left(D_n - 2C_n\right) y^{-n}\right). \tag{4.11}$$

Comparing the individual terms y^{-k} on both sides we get that on the left hand side C_k is the term multiplied by y^{-k} and on the right hand side we will get a sum of terms on the form $E_i\left(D_j - 2C_j\right)$ which have the property that $i + j = k$. This means that we can now write the relationships between our constants explicitly as follows.

$$C_n = \sum_{i=o}^{n} E_{n-i}\left(D_i - 2C_i\right), \tag{4.12}$$

$$C_0 = -\frac{1}{K_1}, \tag{4.13}$$

$$D_0 = \frac{1}{K_1}, \tag{4.14}$$

$$D_n = -\left(n+1\right)C_n = n\left(n+1\right)B_n, n \neq 0. \tag{4.15}$$

As we now have all the relations between the constants we can start to calculate what they are. The first step is to find E_0 using the fact that C_0 and D_0 are known. We can then insert this result in equation (4.6) and compare the constant term to retrieve B_0. Then we can use (4.12) to express E_1 as a function of B_1. By repeating this process we can get all the constants B_n and E_n. So let us see how this works. First we can use (4.12) to express E_n as a function of B_n. When doing so we can obviously assume that all B_i and E_i are known for $i < n$:

$$\begin{aligned}
C_0 &= E_0\left(D_0 - 2C_0\right)\\
&\Rightarrow E_0 = \left(\frac{1}{K_1} - 2\left(-\frac{1}{K_1}\right)\right)^{-1}\left(-\frac{1}{K_1}\right) = -\frac{K_1}{3}\frac{1}{K_1} = -\frac{1}{3},\\
C_n &= \sum_{i=o}^{n} E_{n-i}\left(D_i - 2C_i\right) =\\
&= -\frac{1}{3}\left(D_n - 2C_n\right) + \sum_{i=1}^{n-1} E_{n-i}\left(D_i - 2C_i\right) + E_n\frac{3}{K_1}\\
\Rightarrow E_n &= \underbrace{\frac{K_1}{9}n^2}_{\text{Known}} B_n - \underbrace{\frac{K_1}{3}\sum_{i=1}^{n-1} E_{n-i}\left(i\left(i+1\right) + 2i\right)B_i}_{\text{Known}}\\
&= F_n^1 B_n + F_n^2.
\end{aligned}$$

For B_0, we get that

$$B_0 = \frac{1}{K_1^2}\left(2K_1 + K_2 + 2K_4 - \frac{1+K_4}{3}K_3\right). \tag{4.16}$$

As we have now described E_n as a function of B_n we can insert our results into equation 4.6 and solve for B_n.

$$K_1 B_n y^{-n} + (K_1 + K_2 + 2K_4)\, C_n y^{-n} - K_3 y^{-n} \sum_{i=0}^{n} E_i C_{n-i} - K_4 D_n y^{-n} = \begin{cases} \delta y^{-1}, & n = 1 \\ 0, & n > 1 \end{cases}. \tag{4.17}$$

Using that C_n, D_n and E_n can be expressed as functions of B_n we can solve for B_n and get that (for $n > 1$)

$$\left(K_1 - (K_1 + K_2 + 2K_4)\, n + K_3 \left(\frac{1}{K_1} F_n^1 + \frac{n}{3}\right) - n\,(n+1)\, K_4\right) B_n = K_3 \left(\sum_{i=1}^{n-1} E_i C_{n-i} - \frac{1}{K_1} F_n^2\right). \tag{4.18}$$

$$B_n = \frac{K_3 \left(\sum_{i=1}^{n-1} E_i C_{n-i} - \frac{1}{K_1} F_n^2\right)}{\left(K_1 - (K_1 + K_2 + 2K_4)\, n + K_3 \left(\frac{1}{K_1} F_n^1 + \frac{n}{3}\right) - n\,(n+1)\, K_4\right)}. \tag{4.19}$$

Note that for $n = 1$ the formula is somewhat different. From equation (4.17) an extra δ occurs on the right hand side of the equation (4.18) and in the numerator in equation (4.19). Now that all required formulae have been derived it is possible to solve for the coefficients numerically.

4.1.2 Algorithm

Since the coefficients B_n are known $\widetilde{W}(y)$, $\widetilde{W}'(y)$ and $\widetilde{W}''(y)$ are also known, see equations (4.7)- (4.9). The coefficients B_n are calculated in MATLAB, the interested reader can study Appendix A2. This makes it possible to retrace our steps and obtain $W(z)$, $W'(z)$ and $W''(z)$ using the following relations:

$$\begin{aligned} z &= y^2 \widetilde{W}' \\ W(z) &= \widetilde{W} + y\widetilde{W}' \\ W'(z) &= \frac{1}{y} \\ W''(z) &= -\frac{1}{y^4 \widetilde{W}'' + 2y^3 \widetilde{W}'}. \end{aligned}$$

The optimal controls can be expressed as a function of z according to equations (3.10) Finally, the value function and optimal controls are plotted as a function of z with different values of correlation ρ, stock volatility σ and income volatility δ.

4.2 Markov chain approach

The approximating Markov chain approach was initially developed by Kushner and Dupuis in the early 1990's and is very well documented. In the following subsections we will in detail describe more specific theory that is used for this method and provide a way of constructing an approximate Markov chain for some processes. We will also derive the formulae for the optimal consumption problem and describe precisely how the approximate solution is obtained.

4.2.1 Markov decision process

In this section we will focus on the Markov decision process and the admissible controls attached to this process. We begin with a formal definition of the Markov decision process, also called the controlled Markov chain, following definition is found in [8].

Definition 8 *The Markov decision process is defined by* $(S,C,\{P(u)\}_{u\in C},\pi_0)$*, where* S *is a finite state space,* C *is a finite set of actions and for each* $u \in C$*,* $P(u) \in [0,1]^{n\times n}$ *is a probability transition matrix on* S*. Let* $x_k \in S$ *be a state and* π_0 *be the probability distribution of* x_0*.*

The Markov decision process is important because it has the property of controls (actions) attached to the process. Next, we want to express the controls more formal. The following definitions are found in a book written by D.P. Bertsekas [9]. Note that these definitions are quite general in the sense that they describe admissible controls for some discrete-time dynamic systems, hence the Markov property is not necessary for these definitions.

Definition 9 *Let* $x_k \in S_k$ *be a state,*$u_k \in C_k$ *a control and* $w_k \in D_k$ *the random disturbance. Then we can write a discrete-time dynamic system as* $x_{k+1} = f_x(x_k,u_k,w_k)$ *where* $k = 0,1,...,N-1$*. The control* u_k *is constrained to a nonempty set* U_k *in* C_k *where* $u_k \in U_k(x_k)$ *for all* $x_k \in S_k$ *and all* k*. The probability distribution* $P_k(\cdot|x_k,u_k)$ *describe the random disturbance* w_k*, where the distribution is not allowed to depend on prior disturbances* $w_1,...,w_{k-1}$*.*

Definition 10 *The policies that consist of a sequence of functions* $\pi = \mu_0,...,\mu_{N-1}$ *where* μ_k *maps states* x_k *onto controls* $u_k = \mu_k(x_k)$ *such that* $\mu_k(x_k) \in U_k(x_k)$ *for all* $x_k \in S_k$*, are called admissible controls.*

In our case, it is necessary that the admissible controls have the property that the wealth process is non-negative at all times. There are some more relevant notations of the control policies. A policy μ is called Markov if each μ_k only depends on x_k. If the policy is Markov, it is called stationary if μ_k does not depend on the time k. The stationarity tells us that there exists a function $\vartheta : S \to \pi_u$ such that $\mu_k \equiv \vartheta$ for all k. If the policy ϑ is stationary, then the state process x_k is Markov with transition probability matrix P_ϑ.

The goal when using Markov decision processes to numerically solve a problem, is to find an optimal policy. A policy is said to be optimal if it maximizes our value function. If there exist an optimal policy, that policy could be found by using dynamic programming with policy iteration.

4.2.2 Constructing the approximating Markov chain

When dealing with continuous stochastic processes it is often hard to attain explicit solutions, so discrete approximations are very useful. The idea is to approximate the controlled state variable process $(X(t))_{t\in\mathbb{R}_+}$ with a controlled discrete time Markov chain $\xi^h = \left(\xi^h_n\right)_{n\in\mathbb{Z}_+}$ on a discrete state space $\mathbf{R}_h$. In this part we will describe a general method for constructing an approximate Markov chain and state some necessary conditions that will be vital for the numerical convergence of the approximating Markov chain method.
First of all we need to consider a continuous state and space stochastic process. Due to practicality we will use the controlled diffusion process defined below:

Definition 11 *The stochastic diffusion process which is governed by a Brownian motion* $\{W(t)\}_{t\geq 0}$ *is as follows,*

$$dX(t) = b\left(X(t), u(X(t))\right)dt + \sigma\left(X(t)\right)dW(t), \tag{4.20}$$

where $b(\cdot)$ *and* $\sigma(\cdot)$ *are assumed to be bounded and Lipschitz continuous i.e* $|f(y) - f(x)| \leq C|x-y|$ *where* C *is some positive constant.*

Since we want the Markov chain to "follow" the processes we approximate we need some conditions on the Markov chain, these are the local consistency conditions [3] which are defined below

Definition 12 *The local consistency conditions for a controlled Markov chain* $\left\{\xi^h_n\right\}_{n\in\mathbf{Z}_+}$ *is*

$$\mathbf{E}\left[\xi^h_{n+1} - \xi^h_n|\xi^h_n = X, u^h_n = \alpha\right] \equiv b_h(x,\alpha)\Delta t^h(x,h) = b(x,\alpha)\Delta t^h(x,h) + o(\Delta t^h(x,\alpha)), \tag{4.21}$$

$$\mathbf{Var}\left(\xi^h_{n+1} - \xi^h_n | \xi^h_n = X, u^h_n = \alpha\right) \equiv a_h(x,\alpha)\Delta t^h(x,\alpha) = \sigma^2(x,\alpha)\Delta t^h(x,\alpha) + o(\Delta t^h(x,\alpha)), \tag{4.22}$$

which also defines the functions $a_h(x,\alpha)$ and $b_h(x,\alpha)$

This gives us a way to confirm that the transition probabilities we construct are sound.

Now consider a discrete state space $\mathbf{R}_h \subset \mathbb{R}$ and an interpolation function $\Delta t^h(x,h) = \frac{h^2}{\sigma^2(x)+h|b(x,\alpha)|}$ hence the transition probabilities are defined as:

Definition 13 *The transition probabilities for the approximating Markov chain is*

$$p^h(x,x \pm h|\alpha) = \frac{\sigma^2(x)/2 + hb^{\pm}(x,\alpha)}{\sigma^2(x) + h|b(x,\alpha)|}, \; \forall x \in \mathbf{R}_h, \tag{4.23}$$

where α is the applied control at time t and $b^{\pm} = \max\{0, \pm b\}$

which satisfy the consistency conditions.

From a coding point of view these may be cumbersome to implement since both the denominator of the interpolation function $\Delta t^h(z,\alpha)$ and the transition probabilities are control dependent. Now let us consider a different diffusion process, namely a diffusion process governed by a two dimensional Brownian motion. The reason we chose this will become apparent in the coming chapter.

Definition 14 *The stochastic diffusion process which is governed by a two-dimensional Brownian motion $(W_1(t), W_2(t))_{t\geq 0}$ is defined as:*

$$dZ(t) = \left(b\left(Z(t)\right) + u(Z(t))\right)dt + \sigma_1\left(Z(t)\right)dW_1(t) + \sigma_2\left(Z(t)\right)dW_2(t), \tag{4.24}$$

where $b(\cdot)$, $u(\cdot)$ and $\sigma_{1,2}(\cdot)$ are assumed to be bounded and Lipschitz continuous.

Now first of all we need to redefine the transition probabilities in a more manageable way and the best way to proceed is to define the denominator as a function itself, here denoted $Q^h(x,\alpha)$

Definition 15 *The denominator and function that determines the interpolation function $\Delta t^h(z)$ is defined as,*

$$Q^h(z) = \max_{\alpha}\left\{\sigma_1^2 + \sigma_2^2 + h\left(|\alpha| + |b|\right),\right\} \tag{4.25}$$

Where α is the control applied at time t and is bounded by some interval I.

With this out of our hands we define the interpolation function and the more practical transition probabilities implementation-wise as

Definition 16 *The transition probabilities and interpolation function for the approximating Markov chain of the process $Z(t)$ is defined as*

$$\begin{array}{ccc} p^h(z,z \pm h|\alpha) & = & \left((\sigma_1^2(z) + \sigma_2^2(z))/2 + h(b(z)^{\pm} + \alpha^{\pm})\right)/Q^h(z), \\ p^h(z,z|\alpha) & = & 1 - p^h(z,z+h|\alpha) - p^h(z,z-h|\alpha), \\ \Delta t^h & = & h^2/Q^h(z). \end{array} \tag{4.26}$$

The only thing left to do now is to check whether these fulfill the local consistency conditions,

$$\mathbf{E}\left[\xi^h_{n+1} - \xi^h_n | \xi^h_n = z, u^h_n = \alpha\right] = \sum_{z' \in \mathbb{R}_h} (z' - z)p^h(z,z'|\alpha) =$$

$$h\left(p^h(z,z+h|\alpha) - p^h(z,z-h|\alpha)\right) = \frac{h^2\left(b+\alpha\right)}{Q^h(z)} = \Delta t^h(z)(b+\alpha).$$

where we used the relation $a^+ - a^- = a$, $\forall a \in \mathbb{R}$, now all that is left is to check the variance:

$$\mathbf{Var}\left(\xi^h_{n+1} - \xi^h_n | \xi^h_n = z, u^h_n = \alpha\right) =$$

$$\mathbf{E}\left[(\xi^h_{n+1}-\xi^h_n)^2|\xi^h_n=z,u^h_n=\alpha\right]-\mathbf{E}\left[\xi^h_{n+1}-\xi^h_n|\xi^h_n=z,u^h_n=\alpha\right]^2=$$
$$\sum_{z'\in\mathbb{R}_h}(z'-z)^2p^h(z,z'|\alpha)-\left(\Delta t^h(z)(b+\alpha)\right)^2=$$
$$h^2\left(p^h(z,z+h|\alpha)+p^h(z,z-h|\alpha)\right)+o(\Delta t^h(z))=$$
$$\Delta t^h(z)(\sigma_1^2+\sigma_2^2)+o(\Delta t^h(z)).$$

and thus we can conclude that these transition probabilities are locally consistent with the state process in definition 14

4.2.3 The approximating Markov chain for the optimal consumption problem

To construct the approximating Markov chain we need to model it after our continuous space and state process $Z(t)$ defined below,

$$dZ(t)=(k_1\varphi-\zeta+kz)\,dt+\sigma\varphi dW_1(t)+\eta Z(t)\sqrt{1-\rho^2}dW_2(t). \tag{4.27}$$

Now it may not be obvious why this is the wealth process for the reduced case however it will become clearer in the section which describes the convergence of the method. The process $Z(t)$ is a positive valued process so the state space will simply be $\mathbf{R}_h=\{0,h,2h,...,Ih\}$ where I is some big number and h will be the mesh size. The control, $\left(\zeta^h,\varphi^h\right)=\left(\zeta^h_n,\varphi^h_n\right)_{\mathbb{Z}_+}$, for the Markov chain is a bounded sequence that will be dependent on the current state of the Markov chain i.e $\zeta^h_n=\zeta^h_n\left(\xi^h_n\right)$ and $\varphi^h_n=\varphi^h_n\left(\xi^h_n\right)$ with the bounds $-\delta\le\zeta^h(z)\le K_\zeta z$ and $|\varphi^h(z)|\le K_\varphi$.

The transition probabilities are given by the underlying stochastic process with the methods described in section 4.2.2

$$p^h\left(z,z-h|\zeta,\varphi\right)=\frac{\frac{1}{2}\left(\sigma^2\varphi^2+\eta^2(1-\rho^2)z^2\right)+h\left((k_1\varphi)^-+(\zeta)^++k^-z\right)}{Q^h(z)}, \tag{4.28}$$

$$p^h\left(z,z+h|\zeta,\varphi\right)=\frac{\frac{1}{2}\left(\sigma^2\varphi^2+\eta^2(1-\rho^2)z^2\right)+h\left((k_1\varphi)^++(\zeta)^-+k^+z\right)}{Q^h(z)}, \tag{4.29}$$

$$p^h\left(z,z|\zeta,\varphi\right)=1-p^h\left(z,z+h|\zeta,\varphi\right)-p^h\left(z,z-h|\zeta,\varphi\right), \tag{4.30}$$

for $z\in\mathbf{R}_h$. At the boundaries, $z=0$ and $z=\bar{z}$, these will be:

$$p^h\left(\bar{z},\bar{z}-h|\zeta,\varphi\right)=\frac{\frac{1}{2}\left(\sigma^2\varphi^2+\eta^2(1-\rho^2)\bar{z}\right)+h\left((k_1\varphi)^-+\zeta^++k^-\bar{z}\right)}{Q^h(z)}, \tag{4.31}$$

$$p^h\left(\bar{z},\bar{z}|\zeta,\varphi\right)=1-p^h\left(\bar{z},\bar{z}-h|\zeta,\varphi\right), \tag{4.32}$$

and at the lower boundary

$$p^h\left(0,h|\zeta,\varphi\right)=\zeta^-, \tag{4.33}$$

$$p^h\left(0,0|\zeta,\varphi\right)=1-\zeta^-, \tag{4.34}$$

where
$$Q^h(z)=\sigma^2K_\varphi^2z^2+\eta^2(1-\rho^2)z^2+h\left(|k|z+|k_1|K_\varphi z+\max\{\delta,K_\zeta z\}\right).$$

In this thesis we will not consider any other transition probabilities to be possible.
Before we can utilize these transitions we need to confirm that this Markov chain will fulfill the local consistency conditions.

$$\mathbf{E}\left[\xi^h_{n+1}-\xi^h_n|\xi^h_n=z,\zeta^h_n=\zeta,\varphi^h_n=\varphi\right]=\sum_{z'\in\mathbf{R}_h}(z'-z)p^h\left(z,z'|\zeta,\varphi\right)=$$

$$\frac{h^2(k_1\varphi - \zeta + kZ(t))}{Q^h(z)} = \Delta t^h(z)\,(kz + k_1\varphi - \zeta)$$

$$\mathbf{Var}\left(\xi^h_{n+1} - \xi^h_n | \xi^h_n = z, \zeta^h_n = \zeta, \varphi^h_n = \varphi\right) =$$

$$\mathbf{E}\left[(\xi^h_{n+1} - \xi^h_n)^2 | \xi^h_n = z, \zeta^h_n = \zeta, \varphi^h_n = \varphi\right] - \mathbf{E}\left[\xi^h_{n+1} - \xi^h_n | \xi^h_n = z, \zeta^h_n = \zeta, \varphi^h_n = \varphi\right]^2 =$$

$$\sum_{z' \in \mathbf{R}_h} (z' - z)^2 p^h\,(z,z'|\zeta,\varphi) - \left(\Delta t^h(z)\,(kz + k_1\varphi - \zeta)\right)^2 =$$

$$\Delta t^h(z)\left(\sigma^2\varphi^2 + \eta^2(1-\rho^2)z^2\right) + o(\Delta t^h(z)).$$

4.2.4 The dynamic programming equation

In this part of the thesis we will derive the dynamic programming equation, which will be the DPE from now on, for the Markov chain defined in previous chapter. As we recall the value function for the reduced problem is defined as

$$F(z) = \sup_{(\zeta,\phi)\in\hat{A}(z)} \mathbf{E}\left[\int_0^\infty e^{-\beta t}\log(\zeta(t) + \delta)dt\right].$$

Now by using the interpolation function defined as $\Delta t^h(z) = h^2/Q^h(z)$ and letting $\Delta t^h_n = \Delta t^h(\xi^h_n)$ and $\Delta t^h_n = \sum_{m=0}^{m=n-1} \Delta t^h_m$ we can define the approximate value function as

$$W^h(z) = \sup_{(\zeta^h,\phi^h)\in A^h(z)} \mathbf{E}\left[\sum_{n=0}^{\infty} e^{-\beta t^h_n}\log(\zeta^h_n + \delta)\Delta t^h_n | \xi^h_0 = z\right], \quad z \in \mathbf{R}_h. \tag{4.35}$$

For the discrete Markov chain according to Munk [4] this will become $I_\zeta = [-\delta, K_\zeta z]$, $I_\varphi = [-K_\varphi z, K_\varphi z]$

$$W^h(z) = \sup_{\zeta\in I_\zeta,\varphi\in I_\varphi}\left\{\Delta t^h(z)\log(\zeta+\delta) + e^{-\beta\Delta t^h(z)}\sum_{z'\in\mathbf{R}_h} p^h\,(z,z'|\zeta,\varphi)\,W^h(z')\right\}. \tag{4.36}$$

Policy iteration

To solve the dynamic programming equation we utilize the policy iteration algorithm. We start with an arbitrary selected control $\left(\zeta^h_0(z), \varphi^h_0(z)\right)$ and solve the system

$$W^h_0(z_i) = \Delta t^h(z_i)\log(\zeta^h_0 + \delta) + e^{-\beta\Delta t^h(z_i)}\sum_{z'_i\in\mathbf{R}_h} p^h\left(z_i,z'_i|\zeta^h_0,\varphi^h_0\right)W^h_0(z'_i), \quad \forall z_i \in \mathbf{R}_h,$$

then we compute a so called policy improvement which is simply that we update the values of ζ and φ which will be given below:

$$\zeta_j(z) = \arg\max_{\zeta\in I_\zeta}\left\{\log(\zeta+\delta) + e^{-\beta\Delta t^h(z)}\left(-\zeta^+ D^- W^h_{j-1}(z) + \zeta^- D^+ W^h_{j-1}(z)\right)\right\},$$

$$\varphi_j(z) = \arg\max_{\varphi\in I_\varphi}\left\{\frac{1}{2}\sigma^2\varphi^2 D^2 W^h_{j-1}(z) - (k_1\varphi)^- D^- F^h_{j-1}(z) + (k_1\varphi)^+ D^+ W^h_{j-1}(z)\right\},$$

where

$$D^2W(z) = \frac{W(z+h) - 2W(z) + W(z-h)}{h^2},\ D^+W(z) = \frac{W(z+h) - W(z)}{h},\ D^-W(z) = \frac{W(z) - W(z-h)}{h}$$

These values will then be used to compute $W^h_j(z)$ in the same manner as $W^h_0(z)$.

By noting that these equations is linear for all $W_j^h(z_i),\ z_i \in \mathbf{R}_h$ the DPE is simply a linear system of equations which we solve for $W_m(z)$, and then update the controls ζ_m and φ_m. The equation is as following:

$$\mathbb{P}_m \mathbb{W}_m = \mathbb{R}_m \tag{4.37}$$

and since the matrix $\mathbb{P}_m$ is tridiagonal this equation is solved very fast. Below the matrix is described in detail.

$$\mathbb{P}_m = \begin{pmatrix} a & b & & & & \\ c & \ddots & \ddots & & \varnothing & \\ & \ddots & a & b & & \\ & & c & a & \ddots & \\ & \varnothing & & \ddots & \ddots & b \\ & & & & c & a \end{pmatrix}, \mathbf{a}_m = \begin{pmatrix} p^h\left(0{,}0|\zeta_m{,}\varphi_m\right) - e^{\beta \Delta t^h(0)} \\ \vdots \\ p^h\left(z_i{,}z_i|\zeta_m{,}\varphi_m\right) - e^{\beta \Delta t^h(z_i)} \\ \\ \vdots \\ p^h\left(\bar{z}{,}\bar{z}|\zeta_m{,}\varphi_m\right) - e^{\beta \Delta t^h(\bar{z})} \end{pmatrix},$$

$$\mathbf{b}_m = \begin{pmatrix} p^h(0{,}h|\zeta_m{,}\varphi_m) \\ \vdots \\ p^h(z_i{,}z_i+h|\zeta_m{,}\varphi_m) \\ \\ \vdots \\ p^h(\bar{z}-h{,}\bar{z}|\zeta_m{,}\varphi_m) \end{pmatrix}, \mathbf{c}_m = \begin{pmatrix} p^h(h{,}0|\zeta_m{,}\varphi_m) \\ \vdots \\ p^h(z_i+h{,}z_i|\zeta_m{,}\varphi_m) \\ \\ \vdots \\ p^h(\bar{z}{,}\bar{z}-h|\zeta_m{,}\varphi_m) \end{pmatrix},$$

$$\mathbb{R}_m = \begin{pmatrix} -\Delta t^h(0)\log(\zeta_m+\delta)e^{\beta\Delta t^h(0)} \\ \vdots \\ -\Delta t^h(z_i)\log(\zeta_m+\delta)e^{\beta\Delta t^h(z_i)} \\ \vdots \\ -\Delta t^h(\bar{z})\log(\zeta_m+\delta)e^{\beta\Delta t^h(\bar{z})} \end{pmatrix}, \mathbb{F}_m^h = \begin{pmatrix} F_m^h(0) \\ \vdots \\ F_m^h(z_i) \\ \vdots \\ F_m^h(\bar{z}) \end{pmatrix},$$

$$\forall z_i \in \mathbb{R}_h.$$

The algorithm in short

Now the algorithm for solving this problem is quite simple, first of all we initialize the controls $(\zeta_0{,}\varphi_0)$, an arbitrary initial state z and of course all the relevant constants. Then we continue by solving the first iteration of the discrete dynamic programming equation $\mathbb{P}_0\mathbb{W}_0^h = \mathbb{R}_0$. Then we generate with our approximating Markov chain a transition to another state, that is $z \to z'$ and solve the controls for the current state. Then with our updated controls denoted $(\zeta_1{,}\varphi_1)$ we solve the matrix equation (4.37) again and check if

$$\sup_z |W_i(z) - W_{i+1}(z)| < \epsilon, \tag{4.38}$$

where ϵ is some user supplied tolerance. If this condition is not fulfilled we generate another transition step and solve everything again until the criteria (4.38) is satisfied.

4.2.5 Convergence scheme

Unfortunately any rigorous proofs regarding the convergence of this method is out of scope for this thesis, however we will argue that our method is sound with the help of already developed theory. First of all we can conclude that our approximating Markov chain will converge towards the state process $Z(t)$ as $h \to 0$ since it is locally consistent, this has been proven for numerous stochastic control problems by Kushner and Dupuis in [3].

So all that is left to prove according to [4] is the stability of the method and that the discrete equation converges to the continuous HJB equation. Furthermore the dynamic programming equation for the discrete case (4.36) will yield the sought result if we write the transition probabilities (4.28)-(4.30) explicitly:

$$\frac{e^{\beta \Delta t^h(z)} - 1}{\Delta t^h(z)} F(z) = \sup_{\zeta \in I_\zeta, \phi \in I_\phi} \left\{ \log(\zeta + \delta) e^{\beta \Delta t^h(z)} + \frac{\Sigma}{2} D^2 F(z) + \mu^+ D^+ F(z) - \mu^- D^- F(z) \right\}, \tag{4.39}$$

where $\Sigma = \sigma^2 \phi^2 + \eta^2 (1 - \rho^2) z^2$, $\mu^\pm = (k_1 \phi)^\pm + \zeta^\mp + k^\pm z$.

Now if we let $h \to 0$ and assume that the finite differences exists i.e

$D^2 F(z) \to F''(z)$, $D^\pm F(z) \to F'(z)$ then we see that this equation converges to the HJB equation of the reduced problem, since $e^{\Delta t^h(z)\beta} \to 1$, $\frac{e^{\beta \Delta t^h(z)} - 1}{\Delta t^h(z)} \to \beta$, $h \to 0$ and $\mu^+ - \mu^- = \mu$. By recalling (4.27) we see that the shape of the state process coincides with the associated HJB equation which actually follows from the derivation of the original equation.

The stability of the method has already been proven by Munk [4] and it utilizes some techniques from functional analysis which is not described in this thesis.

Chapter 5

Results

In this section several plots from both numerical models are presented. The plots show how the value function and the optimal controls respond when the most interesting constants are changed, ceteris paribus. The values of some initial constants are shown in table 5.1, we use the same values as Munk used in his paper [4].

Table 5.1:

Constants		
Stock volatility	σ	0.3
Income volatility	η	0.1
Correlation	ρ	0.0
Time preference rate	β	0.2

In the plots, the ratio z has been used, recall that $z = l/h$, where l denotes initial wealth and h denotes initial income. The initial wealth l has been used to normalize the optimal controls which is possible since $l > 0$ is assumed. Due to normalization, the controls together with the capital placed in the risk-free bond, should sum up to one. Hence the optimal investment strategy can be achieved from the figures in this section.

In the model, it is possible for the investors wealth process $L(t)$ to equal zero at some time t. If the wealth process gets down to zero, it is possible to achieve positive wealth again, due to the random income flow.

5.1 Numerical results for the infinite series expansion

In the following section, we will present the results given by the infinite series expansion. By recalling that Merton's original problem [1] has no random income i.e $h = 0$ it is possible to see that the asymptotic behavior of the controls tend to the Merton solution. Note that in the following plots $h > 0$.

The first interesting constant we choose to vary, is the correlation ρ, between changes in income and changes in the risky stock market. From figure 5.1 we can see that the value function is highest for the highest value of ρ and lowest for the lowest value. Figure 5.2 describes the optimal consumption for different values of ρ. It states that the consumption should be higher for a higher correlation. Finally figure 5.3 declares that if the correlation is high, then your optimal risky investment should be high. The result suggests that the investor should both spend more money and invest more in the risky stock market if the correlation is high. This may seem odd at first glance, but there is a chance that our investor would both make profit on the risky market and also obtain a high income flow (because of the high correlation). This chance is positive indeed, since we assumed in our model that the investor is risk averse, i.e. will not accept fair game or worse. Also note that the total variations of the fractions to be put into risky intestments, risk-free bonds and on comsumption are quite small. The distributions is roughly said about 50% risky investments, 30% risk-free bonds and 20% consumption for large values of z. For small z, i.e. when your initial wealth is not extremely large compared to your initial income, there is a lot more variance. This is also reasonable.

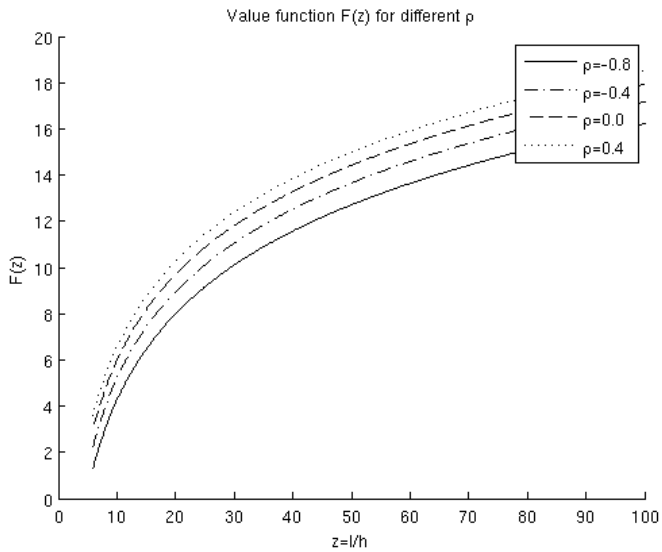

Figure 5.1: Value function F(z) for different values of the correlation ρ between changes in income and changes in the risky stock market.

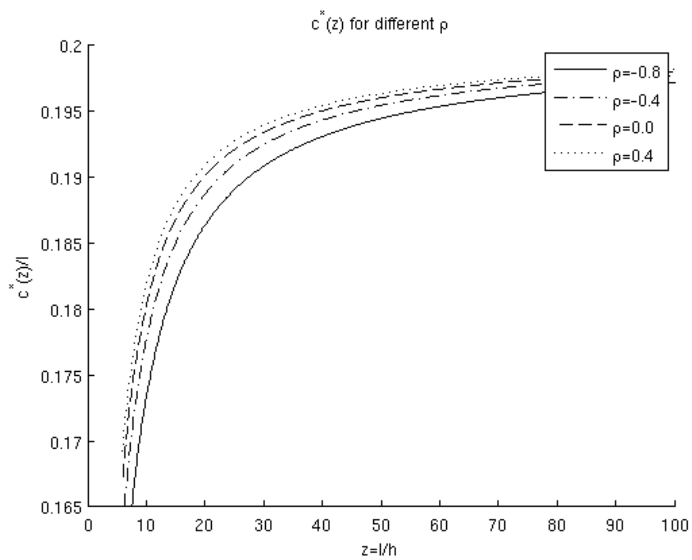

Figure 5.2: Optimal consumption c^*/l for different values of the correlation ρ between changes in income and changes in the risky stock market.

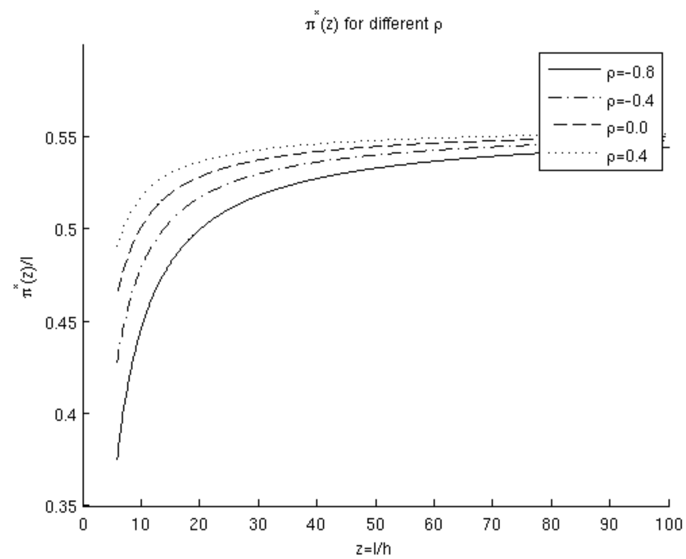

Figure 5.3: Optimal risky investment π^*/l for different values of the correlation ρ between changes in income and changes in the risky stock market.

The next interesting constant is the stock volatility σ. Figure 5.4 decribes the value function for different values of σ. We can see that the value function is high for high values of stock volatility and lower for lower values. From figures 5.5 and 5.6 it is evident that it is resonable to make large risky investments when the market is relatively stable (i.e. when σ is low) and also to consume a smaller fraction of your wealth. Note that the change in consumption is small, while the change in investments in risky stock markets is very high. This means that the distribution between risky and risk-free investments is highly affected by changes in stock volatility.

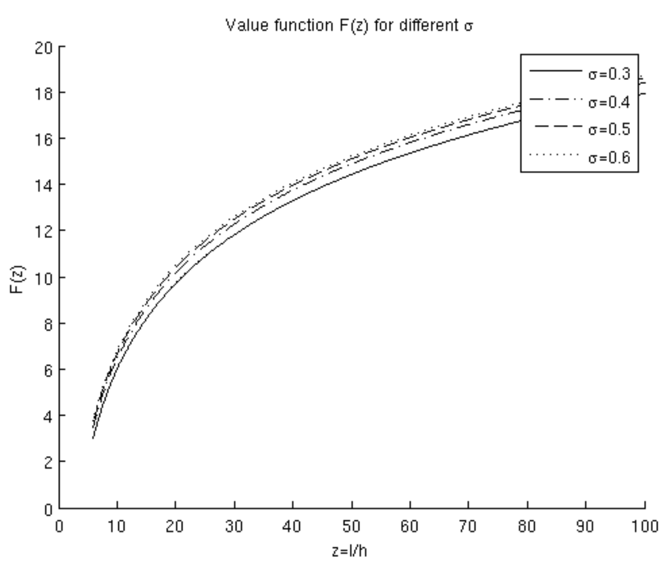

Figure 5.4: Value function F(z) for different values of stock volatility σ.

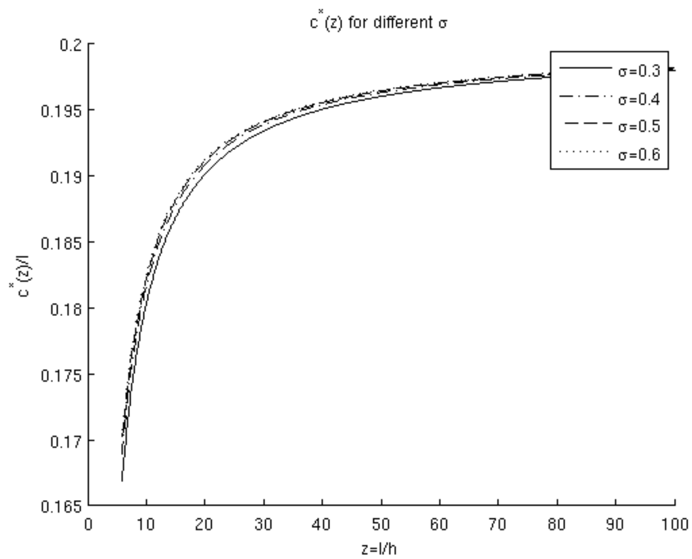

Figure 5.5: Optimal consumption c^*/l for different values of stock volatility σ.

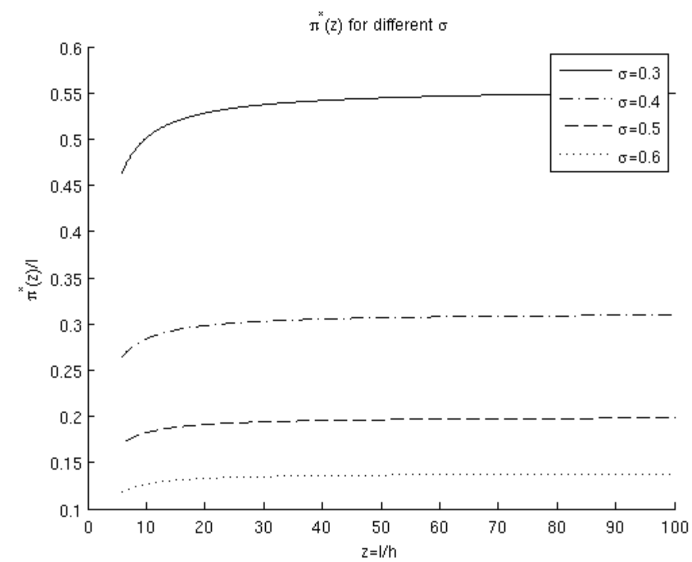

Figure 5.6: Optimal risky investment π^*/l for different values of stock volatility σ.

The last constant of interest is the income volatility η. Figure 5.7 declares that the value function is shifted upwards for increasing values of η. In figures 5.8 and 5.9 we can see that the optimal consumption and optimal risky investment also follow this pattern. In other words the investor should invest more money on the risky stock market and also consume more if the income volatility is high. This may seem a bit counterintuitive, but that does not mean that it is completely wrong. A motivation to the risky investments could be that since our investor is risk averse it might be a good idea to make risky investments in order to make profit to avoid bankruptcy. When studying the result for the optimal consumption, we must consider what it truly is we are optimizing. It is not strictly speaking to maximize wealth, but instead to maximize the utility function, and therefore it might be preferable to consume more with high income volatility.

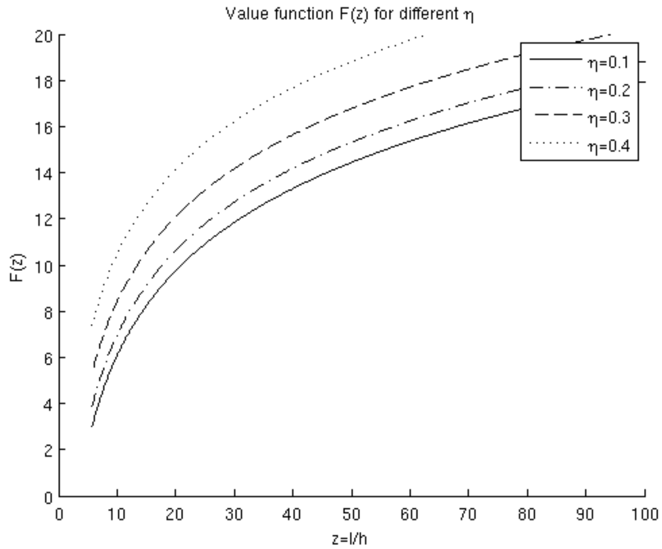

Figure 5.7: Value function F(z) for different values of income volatility η.

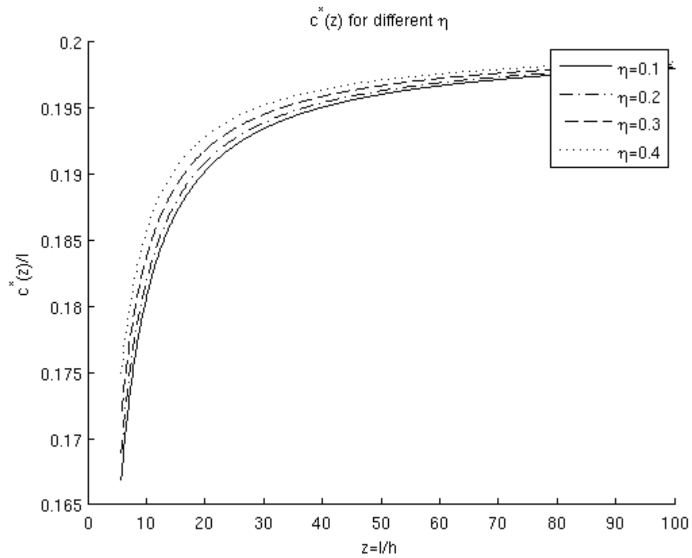

Figure 5.8: Optimal consumption c^*/l for different values of income volatility η.

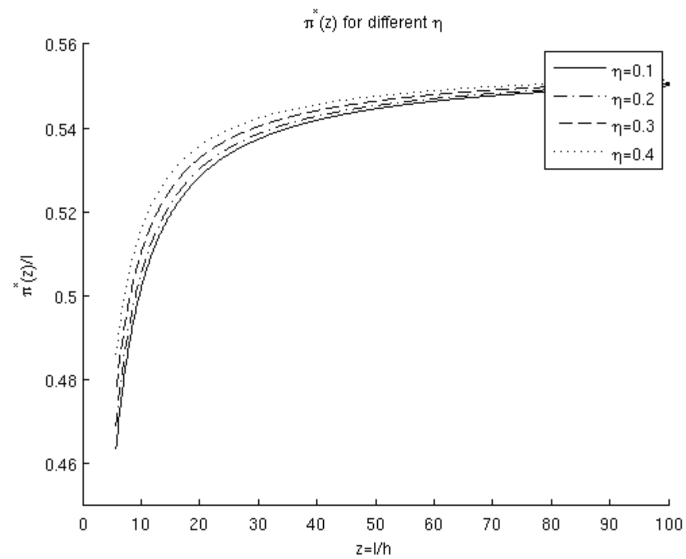

Figure 5.9: Optimal risky investment π^*/l for different values of income volatility η.

To conclude, the only parameter that has a truly large influence on our optimal investments is the market volatility. It affects the distribution between risky and risk-free investments. The other parameters have interesting effects but does not affect the total ratios as much.

5.2 Numerical results for the Markov chain approach

In this section we present the figures from the numerical Markov model. The code used to produce the figures in this section can be found in appendix A.1. In the code, we differentiate numerically when solving for the optimal controls, described in equation (3.10). The value function produced by our code is quite numerically unstable, which make our numerical differentiation very unstable. For some low values, the value function is unstable and the optimal controls tend to infinity. Since the problematic values are not of interest, these have been removed from the plots. The problematic region is included in the plots shown in appendix C. The area of focus should be where the ratio z is larger, therefore the behavior of our functions close to zero is less important.

The first constant of interest is the correlation parameter ρ, which describes correlation between changes in income and changes in the risky stock market. Figure 5.10 shows the value function for four different values of ρ. The value of the function $F(z)$ is lowest when $\rho = -0.8$ and highest when $\rho = 0.0$. The interpretation of this result should be, that if there is negative correlation, the stock market will do worse at the same time at which the investor achieve new income. Hence the investor will have low risk of having zero wealth, but have no possibility to save the assets and thereby increase their value. If the correlation instead is positive, the investor will gain income at the same time as the stock market does well. In this scenario the investor can keep the assets and achieve a higher wealth before the income vanishes and the stock market does worse again. Hence the investor will be better off when the correlation is positive compared to negative. When income and the changes in the stock market are uncorrelated, the investor will have the opportunity to save some of the assets most of the time. The growth of these assets will increase the investors wealth and thereby this scenario is optimal.

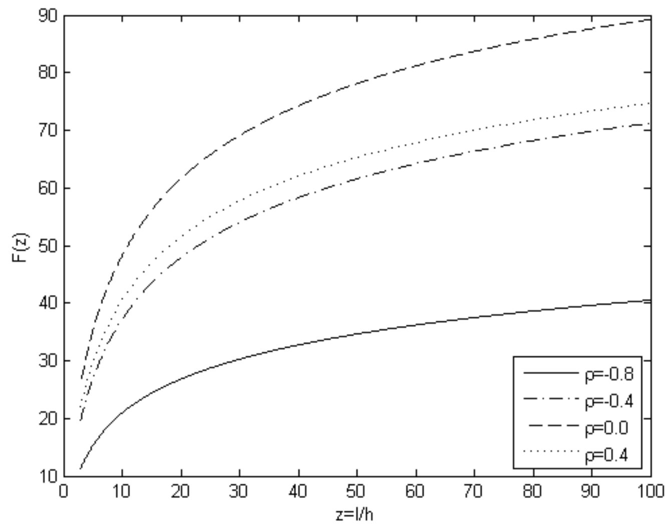

Figure 5.10: Correlation between changes in income and changes in the risky stock market affect the value function $F(z)$, note that $z = l/h$, where l denotes wealth and h denotes income of the investor.

The effect of correlation is also visible when the optimal consumption rate $c^*(z)/l$ is plotted. In figure 5.11, the model suggest that the investor should consume more when the correlation is non-zero. Consistently with what the plot in figure 5.10 showed, the consumption should be highest when $\rho = -0.8$. Note that the optimal consumption is higher for $\rho = 0.4$ than $\rho = -0.4$ when z is low, but when z is higher, the consumption should be greater for $\rho = -0.4$ than $\rho = 0.4$. There is no straightforward explanation for this behavior, it could be a numerical issue in the code.

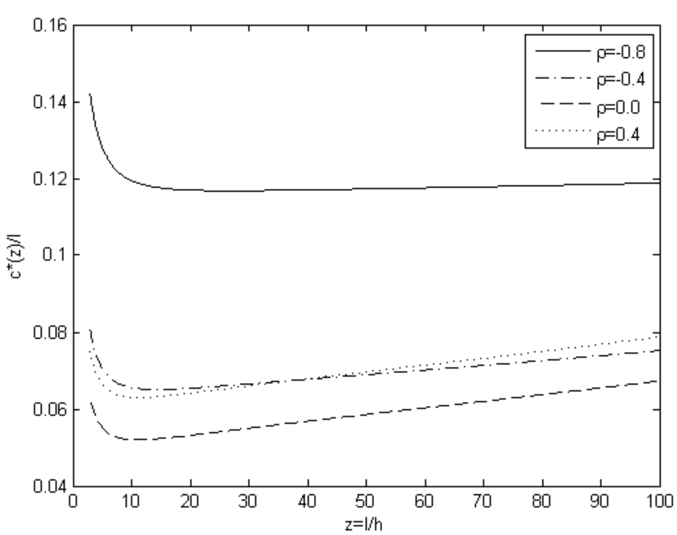

Figure 5.11: The effect of correlation on the optimal consumption rate $c^*(z)/l$.

To complete the picture given by figure 5.10 and 5.11, the optimal risky investment is shown in figure 5.12. The model suggest that the investment on the risky stock market should be highest when the correlation is negative and lowest when the correlation is zero. The result of the figures shown is that when the correlation is negative, most of the investors wealth should be invested on the risky stock market or in the bond. Hence less money will be spent on consumption. Both for positive and zero correlation, there will be a greater amount invested in the bond and spent on consumption, in expense of the risky investment. It is also possible to observe that the trade-off between the three choices seem to be lower when the correlation is highly negative. For the other values of correlation shown in the figures, the slope of the optimal consumption is increasing while the slope of the optimal risky investment is decreasing. An interpretation is that if the correlation is highly negative, the investor has limited choices and will behave in a similar way regardless of the fraction z.

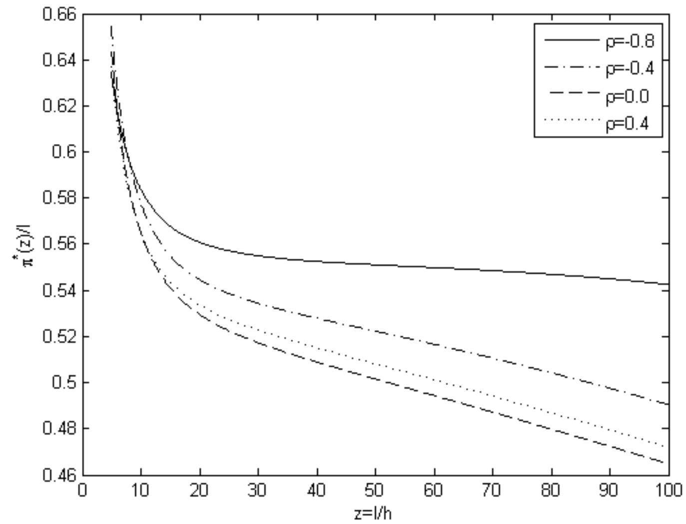

Figure 5.12: Correlation affects the optimal risky investment, $\pi^*(z)/l$, when the correlation is negative, the optimal risky investment is higher.

The volatility of market, denoted σ, is the next important constant. In figure 5.13 the value function with different σ is presented. In the figure it is clear that when volatility of market increases the value function gets lower.

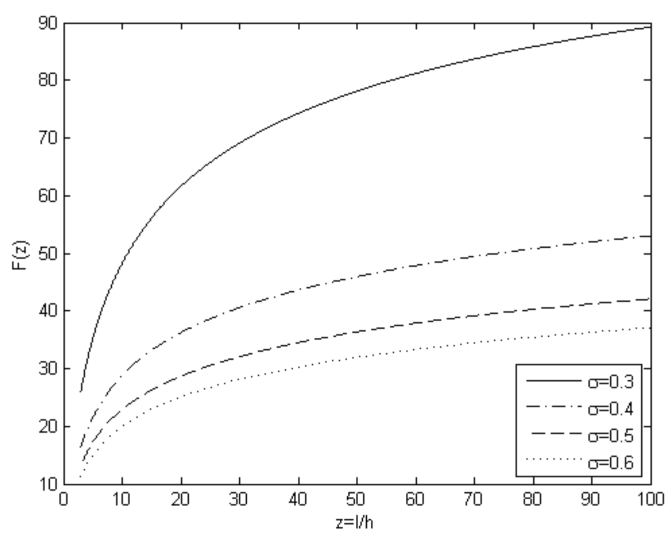

Figure 5.13: When the volatility is higher, the corresponding value function is lower.

When the volatility of market increase, the choice of consumption becomes more attractive, this relation is shown in figure 5.14. It is also possible to observe that the slope of the curves are increasing, suggesting that a higher amount should be consumed, if the initial ratio z is larger.

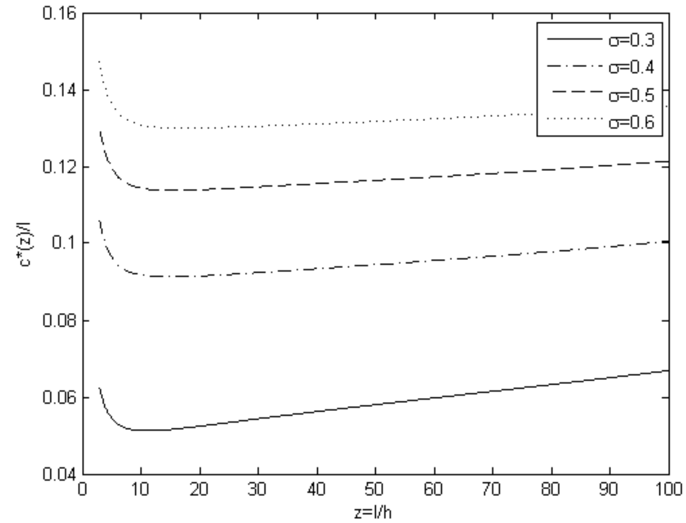

Figure 5.14: Higher volatility of the stock market will increase the optimal consumption ratio.

The third plot, shown in figure 5.15, describes how the volatility of market affect the optimal risky investment ratio. When the volatility is higher the investment is also higher. Here π^* have decreasing slope for all values of σ. Combined with the change in slope in figure 5.14, this result suggest that if z is high, a greater amount could be invested on the risky stock market in expense of consumption. It is also possible to observe that both consumption and the investment on the risky stock market are quite low, that implies that the investment in the risk free bond is high.

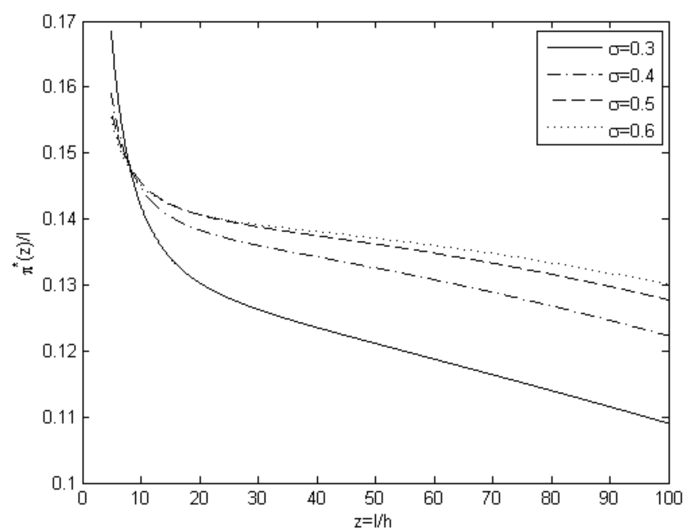

Figure 5.15: The volatility of the stock market affect the optimal risky investment ratio, $\pi^*(z)/l$.

The last constant of interest is the income volatility, denoted η. When the numerical simulations were performed, the model was very unstable when changing this constant. The optimal control ζ has a logarithmic term containing η which makes the model unstable for large values of this constant. Since the control is updated during the calculations of the value function, this numerical problem has major impact on the accuracy of the model. To make the model work properly, the changes in income volatility where chosen quite small. In figure 5.16, the value functions are shown. In the figure it is possible to note that when income volatility increases the value function also shift upwards.

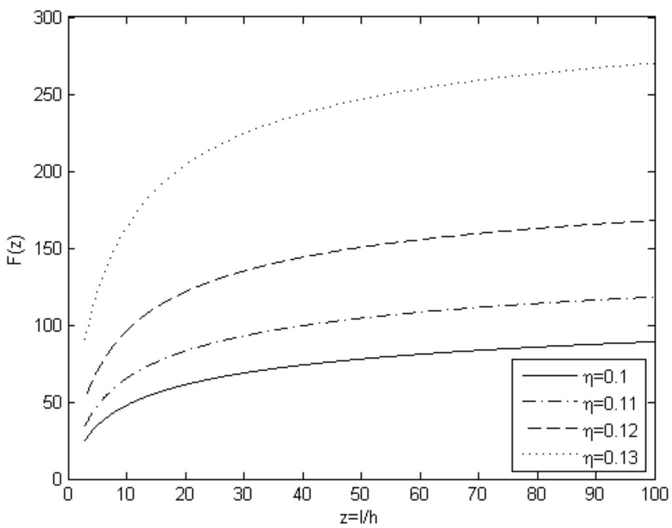

Figure 5.16: When volatility of income increases, the value function shift upwards.

When the income volatility increases the model suggest that the investor should consume less. In figure 5.17 it is clearly shown that the optimal consumption is higher when the income volatility is lower. This result is quite reasonable, if there is high income volatility it is better to save some of the income for the future instead of consuming. Note that the optimal consumption has an increasing slope for all values of η.

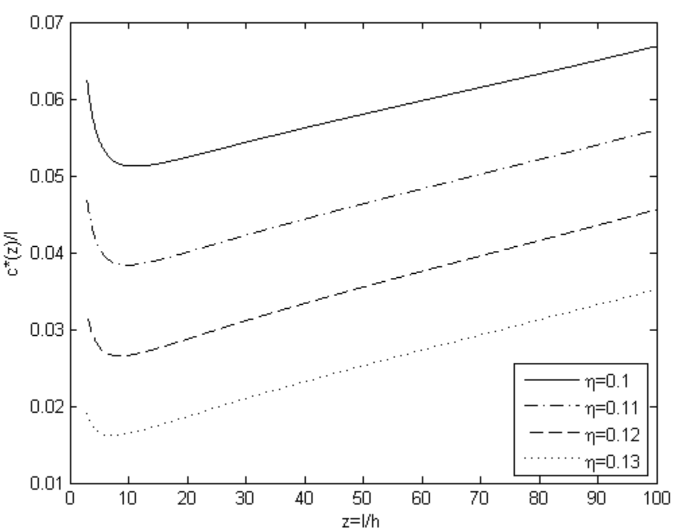

Figure 5.17: If the income volatility is higher, the investor should consume less.

The last picture in this section, figure 5.18, shows the optimal risky investment ratio. When income volatility is high, the optimal risky investment is low. Note that the optimal risky investment decreases when z increases for all values of η, this implies that the consumption is high in expense of the risky investment. The values of the fractions $\pi^*(z)/l$ and $c^*(z)/l$ states that a very small amount should be consumed, the remaining part should be invested in the risky stocks and the risk free bond. Since z denotes initial wealth divided by initial income, a high value of z implies that either the initial income is very low, or the initial wealth is very high. Hence the investor should be less affected by high volatility of income when z is high, because the initial wealth endowment is so dominating. When the investor is less dependent of the income, a greater amount could be spent on consumption, since it is less important when next income is achieved. The investor will make the decision between the investments and consumption based upon the wealth rather than the income.

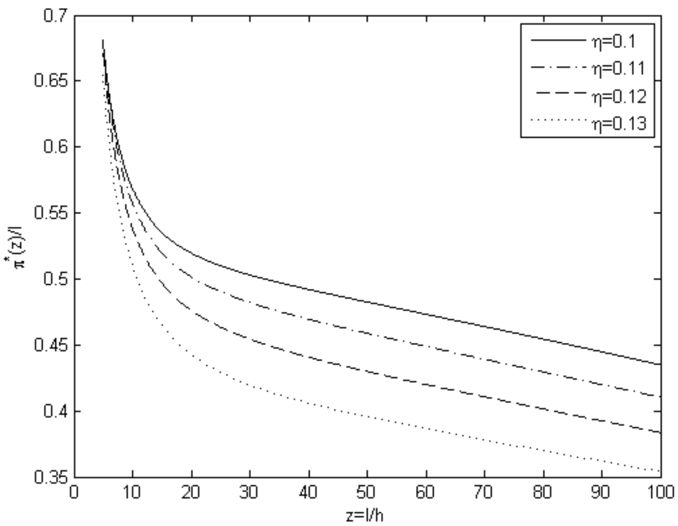

Figure 5.18: Higher volatility of income will decrease the optimal risky investment ratio $\pi^*(z)/l$.

The results shown in this section clearly show that all shown parameters have a major impact on the model and i.e. the investor's decision.

Chapter 6

Discussion

6.1 Infinite Series Expansion

The main reason for choosing the infinite series method is because it gives an exact solution to the problem. If you have a problem which can be solved with this method, you will end up with a near perfect solution. The problem is however, that it does not hold for all utility functions. Another problem is that very high tolerance is needed in the computations, otherwise there is a great risk of rounding errors as the coefficients B_n are very small. Since B_n depends on the parameters this might explain the low sensitivity to changes in parameter values. However the results turned out to be quite reasonable. The only reason for concern is that the results between the methods do not completely coincide.

The shape of the value function is as expected, concave and positive. The values of π^* and c^* seem to give reasonable advice to an investor and the asymptotic behavior supports this conclusion.

6.2 Markov chain approach

The Markov chain method has numerous advantages to the infinite series expansion, first of all it is possible to generalize the code for other utility functions although the reduction of the problem to an ODE is still required. According to Munk [4] it is actually of great importance that the problem needs to be reduced since the variance and covariance matrix of the original two-dimensional state variables $(L(t),H(t))$ has a non-trivial control-dependence and thus it is not possible to approximate it by a locally consistent Markov chain. Another benefit of this method is that is very well documented and there is a large strand of literature regarding this topic. There are some issues implementation-wise such as the size of the increments of the discrete state space $\mathbf{R}_h$ and the size of the bounds on the controls, if they are to large too much time will be spent on optimizing the controls. Another important remark is that the tolerance when computing solutions of equation (4.36) should be around 10^{-5} otherwise the improvement for each new solution is not worth the computations.

The numerical results of the Markov chain approach turned out quite well. The policies behaved in a reasonable way when important economic constants where changed. The method is very sensitive to changes in income volatility since this constant plays a major role in the computations.

6.3 Comparing the methods

The methods have different strengths and weaknesses. The infinite series expansion is good when one is only interested in exact values with one utility function. The Markov chain method on the other hand is fairly fast to implement for different utility functions but does not yield as exact results. There are some numerical issues with both methods but these are discussed in detail in the previous section. A general conclusion is that if one wants trustworthy results one needs to utilize both methods since the Markov approach can be implemented fast and confirm whether the series expansion used for a certain utility function is correct.

When important parameters where changed, the behavior of the optimal controls differed between the two methods. This result is quite surprising since the optimal controls mostly depend on derivatives of the value function, and the value functions of the two methods have similar shape. Another difference is the scale of the value function, the function achieved by infinite series is much lower than one achieved from the Markov chain approach. The scaling error might be due to some faulty implementation in the Markov code but it is not clear exactly where. An important observation when comparing the methods is that the Markov chain approach seem to depend more on the constants as changes in the optimal controls modeled by infinite series expansion are surprisingly small. This is probably because the values of the coefficients in the series expansion tends to zero very fast and yields rounding errors.

What might have been interesting but probably out of our reach is to formulate some condition on the utility function such that the reduction of the original HJB equation is possible. There is some connection between the choice of transform and utility function, see Munk [4] or Tebaldi/Schwarz work [2] with the HARA utility function. To continue the work done in this thesis one might improve the implementations numerically and draw some more theoretical conclusions regarding the two methods.

Bibliography

[1] Merton R.C. Optimum Consumption and Portfolio Rules in a Continous-Time Model. Journal of Economic Theory 3, 373-413; 1971

[2] Schwartz S. E, Tebaldi C. Illiquid assets and optimal portfolio choice. Working Paper, National Bureau of Economic Research, Cambridge.Available at http://www.nber.org/papers/w12633. 2006

[3] Kushner HJ, Dupuis P. Numerical Methods for Stochastic Control Problems in Continuous Time. 2nd ed. New York: Springer; 2001

[4] Munk C. Optimal consumption/investment policies with undiversifiable income risk and liquidity constraints. Journal of Economic Dynamics & Control 24, 1315-1343; 2000

[5] Bhattacharya R, Waymire, E. Stochastic Processes with Applications. 1st ed. New York: Siam; 1992

[6] Hwei H. Probability, Random Variables, and Random Processes. 2nd ed. New York: McGraw-Hill; 1997

[7] Yamschchikov I, Zhelezov D. Liquidity and optimal consumption with random income [Master thesis]. Halmstad, Halland: Halmstad University; 2011.

[8] A. Arapostathis (The University of Texas at Austin, Electrical and Computer Engineering), R. Kumar (corresponding author, Iowa State University, Electrical and Computer Engineering), S. Hsu (National Chi-Nan University, Electrical Engineering). Control of Markov Chains with Safety Bounds. Paper No.V2004-003/V2003-107.

[9] Bertsekas D.P. Dynamic Programming and Optimal Control. 3d ed. Nashua: Athena Scientific; 2005

[10] Billingsley P. Probability and Measure. 3rd ed. New York: John Wiley & Sons Ltd; 1995

Appendix A

MATLAB code

A.1 The approximating Markov chain approach

```
function x = TDMAsolver(a,b,c,d)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%This function solves tri-diagonal matrices
%using Thomas algorithm, returns the solution
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%a, b, c are the column vectors for the compressed tridiagonal
    matrix, d is the right vector
n = length(b); % n is the number of rows

% Modify the first-row coefficients
c(1) = c(1) / b(1);    % Division by zero risk.
d(1) = d(1) / b(1);    % Division by zero would imply a singular
    matrix.

for i = 2:n-1
    temp = b(i) - a(i) * c(i-1);
    c(i) = c(i) / temp;
    d(i) = (d(i) - a(i) * d(i-1))/temp;
end

d(n) = (d(n) - a(n-1) * d(n-1))/( b(n) - a(n-1) * c(n-1));

% Now back substitute.
x(n) = d(n);
for i = n-1:-1:1
    x(i) = d(i) - c(i) * x(i + 1);
end
end
```

```
function inc = delta_t(z)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% This is the interpolation function
% It calls the denominator function q(z)
% Is used when solving the DPE
% Returns the time increment.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
global h;
inc = h^2/q(z);
```

```
function Q = q(z)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% This is the denominator function
% for both the transition
% probabilities and also the
% interpolation function delta_t
% Returns the denominator for
% the transistions and time increm.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


global rho sigma delta K_phi  K_zeta h;
%ALL the coefficients!!
gamma = 0.5; r = 0.1; b=0.15; mu = 0.05;
k1 = b-r-(1-gamma)*sigma*delta*rho;
k2 = delta^2*(1-gamma)+r-mu;
k = k2+rho*k1*delta/sigma;

Q = sigma^2*K_phi^2*z^2+delta^2*(1-rho^2)*z^2+ ...
    +h*(abs(k)*z+abs(k1)*K_phi*z+max(1,K_zeta*z));
end
```

```
function f = phi_fun_log(x,z,V)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% This function is used when
% solving the DPE associated
% with the problem at hand.
% Returns the updated value of
% the control phi
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% x is phi for which we solve
% z is the state
% V is the current iteration of the value fct

global rho sigma delta; % constants
global h; % increment
global zmax; % Boundaries
gamma = 0.5; r = 0.1; b=0.15;
k1 = b-r-(1-gamma)*sigma*delta*rho;
i = round(z/h)+1;
% Certain cases are required when solving the controls
if z ~= zmax
    f = -(sigma^2*x^2*(V(i+1)-2*V(i)+V(i))*...
        0.5/(h^2)-max(-k1*x,0)*(V(i)-V(i-1))/h+...
        max(k1*x,0)*(V(i+1)-V(i))/h);
else
    f = -(V(end)-V(end-1))*(0.5*sigma*x^2+h*max(-k1*x,0))/h;
end
end
```

```
function f = zeta_fun_log(x,z,F)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% This function is used when
% solving the DPE associated
% with the problem at hand.
```

```
% Returns the updated value of
% the control zeta
%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% x is zeta for which we solve over
% z is the state
% V is the current iteration of the value fct

global h zmax delta I;
beta_hat = 0.17625;
i = round(z/h); Max_Ind = I;
% Certain cases are required when solving the controls
if z>h && z < zmax
    f = -(log(x+delta)*delta_t(z)+exp(beta_hat*delta_t(z))*...
        (-max(x,0)*(F(i)-F(i-1))/h+max(-x,0)*...
        (F(i+1)-F(i))/h));
elseif z == 0
    f = -(delta_t(z)*log(x+delta)+exp(delta_t(z)*beta_hat)*...
        (max(-x,0)*F(1)+(1-max(-x,0))*F(2)));
else
    f = -(log(x+delta)-exp(beta_hat*delta_t(z))*...
        max(x,0)*(F(Max_Ind)-F(Max_Ind-1))/h);
end
end
```

```
function [] = update_log(F,z)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Function that updates zeta_m and phi_m
% Takes in a Matrix P,vector R and a state z.
% Doesn't return, updates the values of
% zeta and phi here
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
global K_phi  K_zeta zeta phi delta h;
options = optimset('MaxFunEvals',10^6);
war1 = 1; war2 = 1;
% dummy are dummy variables
% war1,2 are warning variables that tells us that no good solution
    was
% found
[x1,dummy1,war1]= fminbnd( @(x)zeta_fun_log(x,z,F),-delta,K_zeta*z
    ,options );
if z>h
    [x2, dummy2,war2]= fminbnd( @(x)phi_fun_log(x,z,F),-K_phi*z,
        K_phi*z,options);
    phi = x2;
end
zeta = x1;
if war1 ~= 1
    'Bad solution for control zeta'
end
if war2 ~= 1
    'Bad solution for control phi'
end
end
```

```
function prob = ptrans_log(z1,z2)
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% This function computes the transition
% probabilities for the approximating markov
% chain.
% It checks the special cases first,
% i.e zmax, z = 0 and then the basic cases
% Returns the probabilities
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


global rho sigma delta eta; % constants
global zeta phi; % control
global h; % increment
global zmax;
%Coefficients
gamma = 0.5; r = 0.1; b=0.15; mu = 0.05;
k1 = b-r-sigma*delta*rho;
k2 = delta^2*(1-gamma)+r-mu;
k = k2+rho*k1*delta/sigma;
a = (sigma^2.*phi^2+delta^2*(1-rho^2)*z1^2)*0.5;
% Special cases z = zmax and z = 0
if z1 == zmax || z1 == 0
    if z1 == 0
        if z2>0
            % Transition from 0->h
            prob = max(-zeta,0);
            return
        else
            % Transition from 0->0
            prob = 1-max(-zeta,0);
            return
        end

    end
    % Transition from zmax -> zmax-h
    if z2< zmax
        prob = a+h*(max(-k1*phi,0)-max(zeta,0)+max(-k,0)*zmax);
        prob = prob/q(z1);
        return
    else
    % Transition from zmax -> zmax
        prob = a+h*(max(-k1*phi,0)-max(zeta,0));
        prob = 1-prob/q(z1);
        return
    end

end
% Basic cases
if z1>z2 % checking which transistion is made
    % Transition from z -> z-h
    prob = a+h*(max(-k1*phi,0)+max(zeta,0)+max(-k,0)*z1 );
    prob = prob/q(z1);
    return
elseif z1<z2
        % Transition from z -> z+h
        prob = a+h*(max(k1*phi,0)+max(-zeta,0)+max(k,0)*z1);
        prob = prob/q(z1);
```

```
            return
        else
            % Transition from z -> z
            p1 = a+h*(max(k1*phi,0)+max(-zeta,0)+max(k,0)*z1);
            p2 = a+h*(max(-k1*phi,0)+max(zeta,0)+max(-k,0)*z1);
            prob = 1-(p1+p2)/q(z1);
end
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% This is the main file
% This is where the calls and the computations Mare made
% We simulate the continuous state process with
% the approximating markov chain and solve it
% The file is composed of to sections,
% 1: first time initialization and solving of the value function
% 2: Solving the DPE until convergence criteria is met
%    and computing the controls through the value function
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% All relevant constants defined, K's are restrictions for
    controls
% clear all; clc
global rho sigma delta K_phi  K_zeta eta;
global zeta phi % control
global h I; % increment
global zmax;
%positive constants required for applied control
sigma = 0.5; delta = 0.1; rho = 0.2; eta = 1;
%State increment and maximum inc
h = 0.05; I = 20000;
%initial states
t = [];
t(1) = 0;
z = 3*h; % Z(0) = z
zmax = I*h;
Z =[];
Z(1) = z;
zeta = 1/2; % Initial control
phi = 1/2; % Initial control
% Bounded controls
K_phi = 0.4;
K_zeta = 0.5;
beta_hat = 0.2; % from Munks paper
beta = 0.2;

% First time initiation
% Since the transistion matrix is tridiagonal it will be composed
% of three vectors a,b,c and utilizing the TDMA algorithm
rho = rho + 0.2;
a = zeros(1,I); % subdiagonal
b = zeros(1,I+1); % diagonal
c = zeros(1,I); % superdiagonal
R = zeros(1,I+1); % resultvector
for i = 1:length(b) % initiating the vectors
    R(i) = -(delta_t((i-1)*h)*log((i-1)*h+delta))*exp(delta_t((i
        -1)*h)*beta_hat);
    b(i) = ptrans_log((i-1)*h,(i-1)*h)-exp(beta_hat*delta_t((i-1)*
```

```
        h));
end
for i = 1:length(a)
    a(i) = ptrans_log(i*h ,(i-1)*h);
    c(i) = ptrans_log((i-1)*h,i*h);
end
F_old = TDMAsolver(a,b,c,R);
F_FirstIteration = F_old;
%tolerance and a boolean
epsilon = 10^-9; FirstIteration = 1;
%%                              Calculating F(z)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% This while loop computes the final approximate solution of F,
% this is done by the policy iteration algorithm where we simply
    generate
% a state transition update the controls for our current iteration
     of F
% and then solve it until our convergence criterion is met
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
while   FirstIteration == 1 || max(abs(F_new-F_old))> epsilon

    if FirstIteration == 0 % Simple programming technique really
        F_old = F_new;
    end

    % Generating the markov chain and updates the controls for the
    % current state
    if binornd(1,ptrans_log(z,z+h)) == 1
        z=z+h;
        update_log(F_old,z);
    elseif binornd(1,ptrans_log(z,z-h)) == 1
        z=z-h;
        update_log(F_old,z);
    else
        update_log(F_old,z);
    end
    %Updating the matrix P and vector R and calculating F_m(z)
    for i = 1:length(b)
        R(i) = -(delta_t((i-1)*h)*log((i-1)*h+delta))*exp(delta_t
            ((i-1)*h)*beta_hat);
        b(i) = ptrans_log((i-1)*h,(i-1)*h)-exp(delta_t((i-1)*h)*
            beta_hat);
    end
    for i = 1:length(a)
        a(i) = ptrans_log(i*h ,(i-1)*h);
        c(i) = ptrans_log((i-1)*h,i*h);
    end
    F_new = TDMAsolver(a,b,c,R); % New solution of F
    % This is used during the first iteration and no more
    FirstIteration = 0;
end
%%      Calculating the controls
%   This is where we calculate the controls c and \pi
%   Equations for the controls can be seen in the thesis.
% Coeffiencents
r = 0.1; b=0.15; mu = 0.05;
k1 = b-r-sigma*delta*rho;
```

```
z = 0:h:zmax;
% Numerical derivatives
Fprim = diff(F_new)/h; Fbiss = diff(F_new,2)/h^2;
constant1 = ones(1,length(z))*eta*rho/sigma;
%The controls
c_star = 1./(z(1:end-1).*Fprim);
pi_star = constant1(1:end-2)-((k1*Fprim(1:end-1))./(sigma^2*z(1:
    end-2).*Fbiss));
```

A.2 Infinite series expansion

```
%Main file Infinite series expansion, final version
close all
clear all
clc
global r mu delta beta z_max alpha;


%Constants
z_max=500;
AmountOfPoints=5000;
r=0.1;
sigma=0.3;
mu=0.05;
eta=0.1;
delta=0.3;
rho=0;
beta=0.2;
alpha = 0.15;

%Calculation of value function and optimal controls, for different
%sigma/eta/rho after preferences

[z W c_star pi_star]=InfSeriesFunc(AmountOfPoints, rho, sigma, eta
    );
%sigma = 0.4;
eta = 0.2;
%rho = -0.4;
[z W2 c_star2 pi_star2]=InfSeriesFunc(AmountOfPoints, rho, sigma,
    eta);
%sigma = 0.5;
eta = 0.3;
%rho = 0;
[z W3 c_star3 pi_star3]=InfSeriesFunc(AmountOfPoints, rho, sigma,
    eta);
%sigma = 0.6;
eta = 0.4;
%rho = 0.4;
[z W4 c_star4 pi_star4]=InfSeriesFunc(AmountOfPoints, rho, sigma,
    eta);


% Plots

figure(1)
```

```
set(0,'DefaultAxesColorOrder',[0 0 0],'DefaultAxesLineStyleOrder',
    '-|-.|--|:')
hold on
plot(z,W,z,W2,z,W3,z,W4);
axis([0 100 0 20])
xlabel('z=l/h')
ylabel('F(z)')
title('Value function F(z) for different \eta')
%legend('\sigma=0.3 ', '\sigma=0.4', '\sigma=0.5 ', '\sigma=0.6')
legend('\eta=0.1 ', '\eta=0.2', '\eta=0.3', '\eta=0.4')
%legend('\rho=-0.8 ', '\rho=-0.4', '\rho=0.0 ', '\rho=0.4')
hold off

figure(2)
hold on
plot(z,pi_star,z,pi_star2,z,pi_star3,z,pi_star4);
axis([0 100 0.45 0.56])
title('\pi^*(z) for different \eta')
xlabel('z=l/h')
ylabel('\pi^*(z)/l')
%legend('\sigma=0.3 ', '\sigma=0.4', '\sigma=0.5 ', '\sigma=0.6')
legend('\eta=0.1 ', '\eta=0.2', '\eta=0.3', '\eta=0.4')
%legend('\rho=-0.8 ', '\rho=-0.4', '\rho=0.0 ', '\rho=0.4')
hold off

figure(3)
hold on
plot(z,c_star,z,c_star2,z,c_star3,z,c_star4);
axis([0 100 0.165 0.2])
title('c^*(z) for different \eta')
xlabel('z=l/h')
ylabel('c^*(z)/l')
%legend('\sigma=0.3 ', '\sigma=0.4', '\sigma=0.5 ', '\sigma=0.6')
legend('\eta=0.1 ', '\eta=0.2', '\eta=0.3', '\eta=0.4')
%legend('\rho=-0.8', '\rho=-0.4', '\rho=0.0 ', '\rho=0.4')
hold off
```

```
function [z W c_star pi_star]=InfSeriesFunc(AmountOfPoints, rho,
    sigma, eta)
%%%%%%%%%%%%%%%%%%%%%%%%%%%
% The point with this function
% is to collect values of W,
% c_star and pi_star for
% different values of rho,
% sigma and eta.
% Returns values of z, W,
% c_star, pi_star for current
% values of rho, sigma and
% eta
%%%%%%%%%%%%%%%%%%%%%%%%%%%

% AmountOfPoints is the amount of pionts used in the calculations
% rho is correlation coefficient
% sigma is the stock voaltility
% eta is the income voality


```

```
    global r mu delta beta B K1 K2 K3 K4 N z_max alpha;

    %Amount of iterations when calculating B_n
    N=400;
    B=zeros(1,N);
    E=zeros(1,N);
    F1=zeros(1,N);
    F2=zeros(1,N);

    %Constants
    K1=-beta;
    K2=eta^2+r-mu+delta-(eta*rho)^2 + (alpha-r)*eta*rho/sigma;
    K3=(eta^2+(eta*rho)^2)/2;
    K4=-(-eta*rho*sigma+alpha-r)^2/(2*sigma^2);

    %Starting points of iteration
    E(1)=-1/3;
    B(1)=-(1/K1)*((2*K1+K2+2*K4)*(-1/K1)-K3*E(1)*(-1/K1)-K4/K1);

    %Obatining values of the coefficients B_n
    for n=1:N-1
        F1(n+1)=K1/9*n^2;
        a=0;
            for j=1:n-1
            a=a+E(n-j+1)*j*(j+3)*B(j+1);
            end
        F2(n+1)=-K1/3*a;
        b=0;
        for j=1:n-1
            b=b+j^2*E(n-j+1)*B(j+1);
        end
        if n==1
            B(n+1)=(K3/3*b+delta)/(K1-(K1+K2+2*K4)*n+K3*(1/K1*F1(n
                +1)+n/3)-n*(n+1)*K4);
        else
            B(n+1)=K3/3*b/(K1-(K1+K2+2*K4)*n+K3*(1/K1*F1(n+1)+n/3)
                -n*(n+1)*K4);
        end
        E(n+1)=F1(n+1)*B(n+1)+F2(n+1);
    end



    y_max=fzero(@yMaxRoot,z_max);
    y_step=y_max/AmountOfPoints;
    y=1:y_step:y_max;

    % Getting WTilde, WTildePrim and WTildeBiss
    [ WTilde WTildePrim WTildeBiss ] = WTildeFunc( y );

    % Getting z, W, WPrim and WBiss
    [ W, WPrim, WBiss, z] = Wfunc( WTilde, WTildePrim, WTildeBiss,
        y);

    % Calculating the optimal controls.
    k1 = alpha-r-sigma*eta*rho;
    c_star=zeros(1,length(z));
```

```
    pi_star=zeros(1,length(z));

    c_star = 1./(z.*WPrim);
    const1= ones(1,length(z))*eta*rho/sigma;
    pi_star = const1-((k1*WPrim)./(sigma^2*z.*WBiss));

end
```

```
function [ W, WPrim, WBiss z] = Wfunc( WTilde, WTildePrim,
    WTildeBiss, y)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% This function is used for
% transforming WTilde to W.
% Returns W, WPrim and WBiss
%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% WTilde is the transformed version of the valuefunction
% WTildePrim is the first derivative of WTilde (expressed as a
    series)
% WTildeBiss is the second derivative of WTilde (also expressed as
    a series)

z=y.^2.*WTildePrim;
W=WTilde+y.*WTildePrim;
WPrim=1./y;
WBiss=-1./(y.^4.*WTildeBiss+2*y.^3.*WTildePrim);
end
```

```
function [ WTilde, WTildePrim, WTildeBiss ] = WTildeFunc( y )
%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% This function is used for
% expressing WTilde,
% WTildePrim and WTildeBiss.
% Returns WTilde, WTildePrim
% and WTildeBiss
%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% y is the variable on which WTilde depends

global K1 B N;

sum1=zeros(1,length(y));
sum2=zeros(1,length(y));
sum3=zeros(1,length(y));
for j=1:length(y);
    for i=1:N-1;
        sum1(j) = sum1(j) + B(i+1)*y(j)^(-i);
        sum2(j) = sum2(j) + i*B(i+1)*y(j)^(-i-1);
        sum3(j) = sum3(j) + i*(i+1)*B(i+1)*y(j)^(-i-2);
    end
end

WTilde=-1/K1*log(y)+B(1)+sum1;
WTildePrim=-1./K1./y-sum2;
WTildeBiss=1/K1./y.^2+sum3;

```

```
end
```

```
function zeroPoint = yMaxRoot(y_max)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% This function returns a value
% that is 0 only if the variable
% y_max corresponds to the value
% of y which will later return
% z_max.
% Returns zeroPoint
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% y_max is the maximum value of the variable y

    global z_max B N K1;
    sum=0;
    for i=1:N-1
        sum = sum + i*B(i+1)*y_max^(1-i);
    end
    zeroPoint=z_max+y_max/K1+sum;
end
```

Appendix B

Solving tridiagonal matrices

In this section we describe the algorithm that is used for solving the system of linear equations when the matrix is tridiagonal.

$$\begin{bmatrix} a_1 & b_1 & & & 0 \\ c_1 & a_2 & b_2 & & \\ & c_2 & a_3 & \ddots & \\ & & \ddots & \ddots & b_{n-1} \\ 0 & & & c_{n-1} & a_n \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} d_1 \\ d_2 \\ d_3 \\ \vdots \\ d_n \end{bmatrix} \tag{B.1}$$

The concept of the algorithm is to modify the coefficients

$$c'_1 = \frac{c_1}{b_1}, d'_1 = \frac{d_1}{b_1}$$

$$c'_i = \frac{c_i}{b_i - c'_{i-1} a_i}, i = 2,3,...,n-1$$

$$d'_i = \frac{d_i - d'_{i-1} a_i}{b_i - c'_{i-1} a_i}$$

this is the so-called forward sweep. Then we obtain the solution by doing a backwards substitution:

$$x_n = d'_n, \;\; x_i = d'_i - c'_i x_{i+1}, \; i = n-1, n-2, ..., 1$$

The advantage of this method is that it only takes $O(n)$ operations to solve (B.1) rather than $O(n^3)$ operations when using ordinary Gaussian elimination.

Appendix C

Additional plots for Markov chain approach

In this appendix some additional plots from the Markov chain approach are presented. The plots show how the value function and optimal controls behave for small z. Since this region is not of interest, this part has been removed from the plots presented in the report. In the beginning of the value function shown in figure C.1, a small change of slope can be seen. This deviant part has major impact on the optimal controls, shown in figure C.2 and C.3, which heavily depend on derivatives of the value function.

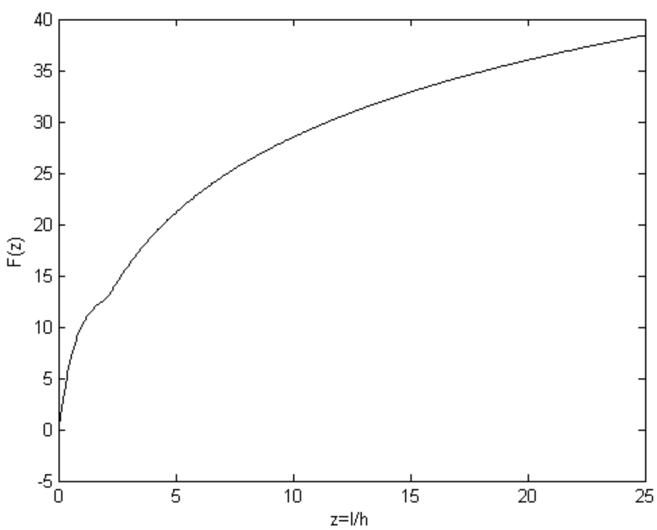

Figure C.1: The value function F(z) when the problematic region is included. Note the instability in the beginning.

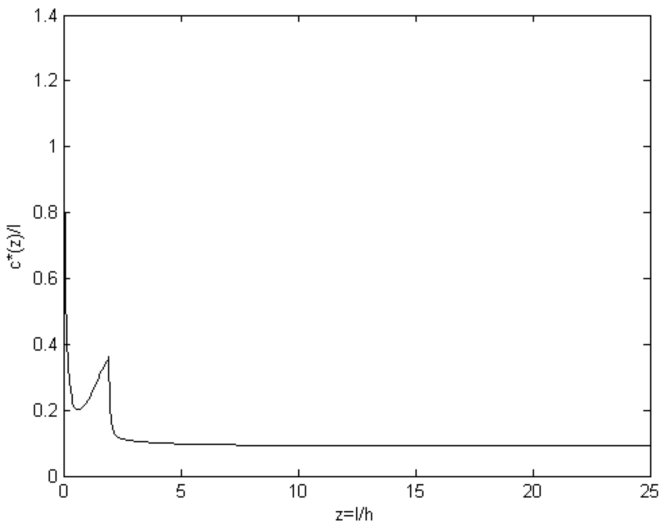

Figure C.2: The optimal consumption becomes unstable.

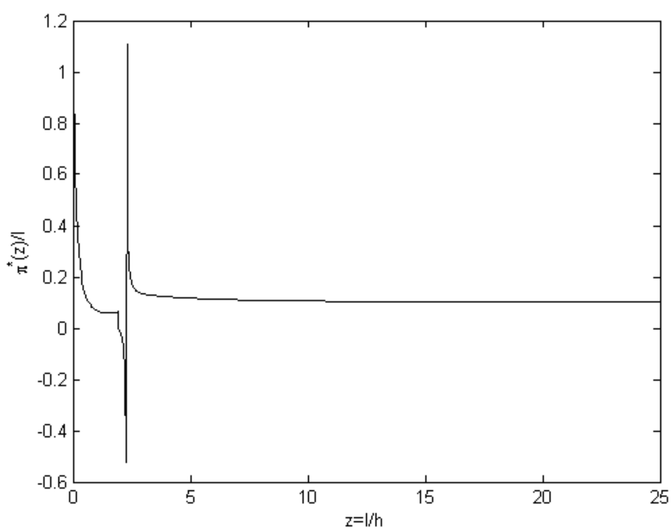

Figure C.3: The optimal risky investment becomes very unstable.