



Technische Universiteit Delft
Faculteit Elektrotechniek, Wiskunde en Informatica
Delft Institute of Applied Mathematics

**A numerical method for backward stochastic differential
equations with applications in finance.**

Verslag ten behoeve van het
Delft Institute of Applied Mathematics
als onderdeel ter verkrijging

van de graad van

**BACHELOR OF SCIENCE
in
TECHNISCHE WISKUNDE**

door

THOMAS HUIJSKENS

**Delft, Nederland
Juni 2013**

Copyright © 2013 door Thomas Huijskens. Alle rechten voorbehouden.



BSc verslag TECHNISCHE WISKUNDE

**“A numerical method for backward stochastic differential equations
with applications in finance.”**

THOMAS HUIJSKENS

Technische Universiteit Delft

Begeleiders

Prof.dr.ir C.W. Oosterlee

Ir. M.J. Ruijter

Overige commissieleden

Dr. J.G. Spandaw

Dr. ir. F.H. van der Meulen

Juni, 2013

Delft

CONTENTS

1. General introduction	6
1.1. Financial mathematics	6
1.2. Advantages and disadvantages of BSDEs	6
1.3. Structure of the thesis	7
2. The Black-Scholes model	9
3. Introduction to BSDEs	14
3.1. Definitions and theorems	14
3.2. Linear BSDEs	16
4. Numerical method for BSDEs	18
4.1. Discretization of forward SDE	18
4.2. Discretization of the BSDE	19
4.3. Binomial tree approximation	22
5. Numerical results and convergence analysis	26
5.1. Stochastic FitzHugh-Nagumo equation	26
5.2. European call option in Black-Scholes market	26
5.3. European call spread in a market with an bid-ask spread for interest rates	28
5.4. Binary option in Black-Scholes model	29
6. Reflected BSDEs	32
6.1. Theorems and examples	32
6.2. Numerical method for RBSDEs	33
7. Convergence analysis of RBSDE scheme	36
7.1. American put option in Black-Scholes market	36
7.2. American call option in a market with a bid-ask spread for interest rates	37
8. FBSDEs with a $\mathbb{R}^2$ -valued forward SDE	39
8.1. Extension of the numerical method	39
9. Convergence analysis of the two-dimensional scheme	42
9.1. European spread option in a Black-Scholes market	42
10. Conclusions and further research	44
10.1. Conclusions	44
10.2. Further research	44
References	46
Appendix	47

1. GENERAL INTRODUCTION

In this thesis we will consider backward stochastic differential equations (BSDEs) and present a method to numerically solve these equations. BSDEs have found important applications in the areas of mathematical finance and stochastic control problems.

A BSDE is stochastic differential equation, but different from the more classical forward stochastic differential equations (SDEs) in the sense that now a terminal condition is specified instead of an initial condition. Furthermore, the solution to a BSDE consists of two different processes, instead of one process. To illustrate, a BSDE is an equation of the following form

$$\begin{aligned} Y_t &= Y_T + \int_t^T f(s, Y_s, Z_s) ds - \int_t^T Z_s dW_s, \\ Y_T &= \xi, \end{aligned}$$

where Y_t and Z_t are stochastic processes, W_t is a Brownian motion, ξ is a random variable called the terminal condition and f is a given function satisfying some properties.

1.1. Financial mathematics. This thesis is mainly considered with applications of BSDEs in the field of financial mathematics. One of the classical problems in the field is the problem of pricing financial options, sometimes called contingent claims. In such problems, we are working in a specific financial market model that consists of several assets in which an investor can invest his money.

One of those assets is a *riskless* asset, as such an asset has a deterministic return rate. An investor knows how much money he is going to end up with when investing in the riskless asset. Furthermore, the market consists of several other financial assets called *stocks*. The return on such investments is not known in advance and therefore these assets are referred to as *risky* assets.

To model the behaviour of these risky assets, one models the price of the stock as a solution to a forward SDE. Commonly, it is assumed that the stock price follows a geometric Brownian motion. This process is specified by two characteristics of the stock, the *drift* and the *volatility* of the stock. The drift captures the change in the expected stock price over time and the volatility is a measure of the variation of the stock level over time.

An option on a stock (called the *underlying* of the option) is a contract that pays out a predefined function of the underlying stock price level. For example, a *European call option* gives the holder of the option the right, but not the obligation, to buy the underlying asset for a predetermined price, called the *strike*, on a predetermined future date, called the *maturity* or *exercise date* of the option.

It is clear that the holder will only exercise this right if the price of the underlying is higher than the strike. In this case the holder can buy the underlying asset for the strike and then sell the stock in the market at the current spot price, which is higher than the strike, to obtain a profit. In more mathematical terms, the *payoff* of the option is equal to

$$\max(\tilde{S}_T - \tilde{K}, 0),$$

where $\tilde{S}_T$ is the stock price at the maturity T and $\tilde{K}$ is the strike of the call option.

A call option is just one of many examples of option contracts. Beside European options, where the option payoff only depends on the stock price level at maturity, this thesis will also look at applications with American options. Such options allow the owners of the option to choose the exercise date, until time T , themselves. The main goal in these kinds of option pricing problems in financial mathematics is to assign a price to such an option.

1.2. Advantages and disadvantages of BSDEs. In the field of financial mathematics, the approach with BSDEs has a couple of advantages compared to the more usual approach with forward SDEs. One advantage of BSDEs is that very many market models can be represented in a system of a forward SDE and a BSDE.

The standard Black-Scholes model can be formulated in terms of BSDEs pretty easily and more advanced models like local volatility models (where the volatility function is a deterministic function of the spot price and time, see [LL11]), stochastic volatility models (where the volatility is assumed to be stochastic itself, see [FTW11]) and jump-diffusion models (where the stock price is allowed to 'jump' up or down, see [Eyr05]) can also be incorporated in the BSDE framework.

Certain market imperfections can also be incorporated using BSDEs. Examples like a different interest rate for borrowing and lending money on the market [Gob10], the presence of transaction costs see [SS08])

and constraints on the amount of stocks an investor can short sell (see [EKPQ97]) can all be fitted in the BSDE framework. Thus, BSDEs provide a very flexible framework in which we can work.

Another advantage of BSDEs is that they can also be used in *incomplete markets* (see [EKPQ97]). When pricing a financial option, we try to find a portfolio of stocks that has the same value as the option and set the price of the option equal to the price of this portfolio, called the *replicating portfolio*. When dealing with incomplete markets however, not every financial option can be replicated by such a portfolio of stocks. One approach is to work with so-called super-strategies, which are portfolios with a value always greater or equal than the option value. Another approach is to assign a utility function to an investor, which represents the attitude toward risk of the investor, and maximize the utility of the replication error, the difference between the value of the super-replicating portfolio and the option value. The maximization problem then encountered can be solved by using BSDEs.

Another advantage of BSDEs is that we do not have to switch to the so-called *risk-neutral measure* when pricing financial options. When pricing financial assets, their expected values have to be adjusted for an investor's risk preferences. However, these discount rates will vary between investors because not all investors have the same risk preference. In complete markets there is an alternative way to do these calculations. One can adjust the probabilities of future outcome such that they incorporate all investors' risk preferences (which is characterized by a number called the *Sharpe ratio*) and then compute expected values in the new probability measure, which is the risk-neutral measure. A lot of mathematics is involved in this technique, and the advantage of BSDEs is that we do not need to switch to this equivalent world anymore and that we can work in one world. This makes the option pricing problem more intuitive and easier to grasp, another advantage of using BSDEs.

Finally, solutions of BSDEs are also linked to solutions of a certain class of partial differential equations (PDEs). This result is provided by an analogue of the well-known Feynman-Kac theorem for forward SDEs. If we are unable to find or approximate a solution to a PDE, we can still obtain some BSDE characterization of the solution without too many problems, and vice versa.

In some ways using BSDEs makes problems in financial mathematics easier to solve, but there are also more difficulties in using BSDEs compared to forward SDEs. A difficulty is that we are now working backwards in time. For a solution of a BSDE to make sense in any real-world application, we do not want the solution to provide us any information of the future, since we do not know the future in any real-world application. In technical terms, we want the solution of the BSDE to be adapted to the underlying filtration we are working on. The solution of a BSDE thus consists of two parts: the solution Y_t and the *control process* Z_t such that Z_t 'steers' Y_t towards the terminal condition and such that adaptedness is ensured.

Another disadvantage of BSDEs is that there are not so many existence and uniqueness results for BSDEs as for forward SDEs. The most regular assumption is that the function f in the BSDE should be Lipschitz continuous, in which case existence and uniqueness is guaranteed. Many researchers tried to relax these assumptions, to obtain results for a more general class of BSDEs, but these results are highly technical in nature. The backward nature of BSDEs also requires more involved numerical procedures, which is the reason why BSDEs are not used by practitioners yet.

1.3. Structure of the thesis. At the moment efficient, practical, numerical methods for BSDEs yet have to be developed. This thesis is an attempt towards that direction, combining the higher order theta-discretization of the BSDE (used, for example, in [ZWP09] and [RO13]) with a binomial tree approximation of the conditional expectations in that discretization (also treated in [MPSMT02] for a different discretization).

This thesis will start with a general introduction of the most fundamental model in mathematical finance: the Black-Scholes model. We develop the basic financial market model and derive the Black-Scholes partial differential equation for general European options and the Black-Scholes option pricing formula for call options.

In section 3 we provide a general introduction to BSDEs and discuss uniqueness and existence results and a Feynman-Kac theorem for BSDEs, which provides a link between PDEs and BSDEs. Furthermore, we show examples of problems in option pricing where a BSDE naturally arises. We then arrive at so-called forward-backward stochastic differential equations (FBSDEs). Such problems also consist of an extra stochastic process whose behaviour is governed by a forward stochastic differential equation and the driver and terminal condition of the BSDE are now allowed to be dependent on the value of this forward process.

To obtain a numerical solution to a FBSDE we first discretize the forward and backward equations. In section 4 we introduce the Euler-Maruyama method to approximate the forward SDE and derive a discretization scheme for the BSDE. This scheme is different from the schemes usually encountered in the literature since we will use a theta-method to approximate certain integrals. It can be proven that this leads to higher order convergence of the solution.

In this backward scheme we arrive at the problem of computing conditional expectations. We solve this problem by assuming a binomial tree structure for the Brownian motion underlying the BSDE. This assumption allows us to derive a numerical scheme for the solution to a FBSDE. In section 5 the behaviour of the numerical method is discussed by considering different problems, mostly problems involving the pricing of a financial option.

As an extra restriction, we can enforce that the value of the solution Y_t of a BSDE should always be greater than some pre-defined barrier. The type of equation that we then obtain is called a reflected backward stochastic differential equation (RBSDE) and is encountered when pricing American options, for example. In section 6 we discuss some properties of these equations and extend the numerical method to deal with RBSDEs. We discuss some examples of RBSDEs in section 7 and analyze the behaviour of our new numerical scheme.

As a final extension, we will look at a combination of two forward SDEs and one BSDE in section 8. The terminal condition is now allowed to be dependent on the values of both solutions of the forward SDEs. These type of equations arise when pricing financial options on two, correlated, assets. We discuss the pricing of a spread option on two correlated assets in chapter 9.

Finally, we present conclusions of the research conducted on this numerical method and provide some topics for further research.

2. THE BLACK-SCHOLES MODEL

In this section we will discuss some of the basic aspects of mathematical finance, risk-neutral option pricing and the Black-Scholes formula. We will work on a filtered probability space $(\Omega, \mathcal{F}, (\mathcal{F}_t)_{0 \leq t \leq T}, \mathbb{P})$ where $\mathcal{F}_t$ denotes the right-continuous version of the filtration generated by a Brownian motion and augmented with the $\mathbb{P}$ -nullsets so that the probability space is complete. Most of the theory described here is adapted from [Sch12].

We assume the existence of two basic assets: a bank account with constant interest rate $r \in \mathbb{R}$ which is continuously compounded, and a risky asset with two parameters: a return $\mu \in \mathbb{R}$ and a volatility $\sigma \in \mathbb{R}_{>0}$. Let $T \in \mathbb{R}$ be fixed and representing a time horizon. For simplicity, we will only work with one risky asset but the results can be generalized to more risky assets. The (undiscounted) behaviour of the bank account is modelled by the following ordinary differential equation:

$$(2.1) \quad d\tilde{S}_t^0 = \tilde{S}_t^0 r dt.$$

The behaviour of the risky asset in *undiscounted* terms is modelled by the forward SDE

$$(2.2) \quad \frac{d\tilde{S}_t^1}{\tilde{S}_t^1} = \mu dt + \sigma dW_t.$$

Note that the solutions of these equations are

$$(2.3) \quad \tilde{S}_t^0 = e^{rt},$$

$$(2.4) \quad \tilde{S}_t^1 = \tilde{S}_0^1 \exp\left(\left(\mu - \frac{1}{2}\sigma^2\right)t + \sigma W_t\right).$$

We will now switch to discounted quantities and use the bank account as a numeraire in our market model. We thus define $S^0 := \frac{\tilde{S}_t^0}{\tilde{S}_0^0} = 1$ and $S^1 := \frac{\tilde{S}_t^1}{\tilde{S}_0^1}$. From equations (2.3) and (2.4) we then obtain:

$$(2.5) \quad S_t^0 = 1,$$

$$(2.6) \quad S_t^1 = S_0^1 \exp\left(\left(\mu - r - \frac{1}{2}\sigma^2\right)t + \sigma W_t\right).$$

In the rest of this thesis $\tilde{X}_t$ will denote undiscounted units and X_t will denote discounted units for any arbitrary variable X_t at time t . One can switch between units by using the relation

$$\tilde{X}_t e^{-rt} = X_t.$$

By Itô's formula, the behaviour of the risky asset S^1 is now modelled by the SDE

$$(2.7) \quad \frac{dS_t^1}{S_t^1} = (\mu - r) dt + \sigma dW_t.$$

A certain class of stochastic processes plays a fundamental role in mathematical finance. These processes are called *martingales* and we have the following definition.

Definition 2.1 (Martingale). A stochastic process X_t on a filtered probability space $(\Omega, \mathcal{F}, (\mathcal{F}_t)_{0 \leq t \leq T}, \mathbb{P})$ is called a $\mathbb{P}$ -martingale (or simply a martingale, if the probability space is clear from the context) if X_t satisfies the following properties:

- X_t is $\mathcal{F}_t$ -measurable for all $t \in [0, T]$, i.e. X_t is *adapted* to the filtration $\mathcal{F}$.
- X_t is integrable:

$$\mathbb{E}_{\mathbb{P}}[|X_t|] < \infty \quad \forall t \in [0, T].$$

- X_t satisfies

$$\mathbb{E}_{\mathbb{P}}[X_t | \mathcal{F}_s] = X_s \quad \forall t \geq s.$$

If a process X_t is a martingale, the expected value of the future given the past is thus equal to the present observed value of the martingale. A martingale is thus a mathematical model for the concept of a 'fair game'. Knowledge of the past events, represented by $\mathcal{F}_s$, does not help to predict future profits or winnings.

A *trading strategy* $\phi = (V_0, \eta_t, \pi_t)$ is specified by an initial capital V_0 , an adapted process η_t representing the holdings in the bank account and a predictable process π_t representing the holdings in the risky asset at time t . We will only consider *self-financing strategies*, i.e. strategies that satisfy

$$(2.8) \quad V_t(\phi) := \pi_t S_t^1 + \eta_t S_t^0 = V_0 + \int_0^t \pi_u dS_u^1,$$

where $V_t(\phi)$ represents the value of the trading strategy ϕ . We drop the dependency on ϕ in $V_t(\phi)$ when it is clear with which strategy we are working.

An equivalent definition of self-financing strategies is that the value of the trading strategy $V_t(\phi)$ should satisfy

$$(2.9) \quad dV_t = \eta_t dS_t^0 + \pi_t dS_t^1.$$

Self-financing strategies are uniquely determined by their initial wealth V_0 and the process π_t , since it can be seen from (2.9) that $\eta_t = V_t - \pi_t S_t^1$. The more rigorous proof of this statement and the equivalence between equations (2.8) and (2.9) can be found in [Sch12].

In any real-world application, trading strategies are bound by some credit line. This credit line can be interpreted as the lower bound on the value process $V_t(\phi)$ of the corresponding strategy ϕ . One can be a certain amount of money in debt, but only within predefined limits. Therefore, we introduce the following definition:

Definition 2.2 (Admissible strategies). Let $a \in \mathbb{R}$. A trading strategy ϕ is called *a-admissible* if its value process $V_t(\phi)$ is uniformly bounded from below by $-a$, i.e. the following should hold:

$$\mathbb{P}[V_t(\phi) \geq -a] = 1 \quad \forall t \in [0, T].$$

A trading strategy is called *admissible* if it is a-admissible for some $a \geq 0$.

In any practical financial market model, trading strategies that are riskless but still profitable should not exist. If they would exist, such strategies would be exploited by investors and hence vanish almost immediately. Such a strategy is called an *arbitrage opportunity*. In mathematical terms, an arbitrage opportunity is defined as follows

Definition 2.3 (Arbitrage opportunity). An *arbitrage opportunity* is an admissible self-financing strategy ϕ with initial wealth $V_0(\phi) = 0$ with $V_T(\phi) \geq 0$ $\mathbb{P}$ -a.s. and $\mathbb{P}[V_T(\phi) > 0] > 0$.

An arbitrage opportunity is thus a strategy that starts with zero value and produces money with a positive probability without risk, because the strategy is self-financing. Any reasonable market model should not allow such strategies and we call a financial market model *arbitrage-free* if arbitrage opportunities in the market do not exist.

Are there any characteristics for a financial market model to be arbitrage-free? It turns out that there is a theorem that gives us a sufficient condition for a market model to be arbitrage-free, called the *fundamental theorem of asset pricing*. Before we can introduce this theorem however, we introduce the following definition:

Definition 2.4 (Equivalent martingale measure). An *equivalent martingale measure* (EMM) $\mathbb{Q}$ is a probability measure $\mathbb{Q}$ equivalent to the probability measure $\mathbb{P}$ such that the process S_t^1 is a $\mathbb{Q}$ -martingale.

The following theorem now provides a sufficient condition for a market model to be arbitrage-free:

Theorem 2.1 (Fundamental theorem of asset pricing (FTAP)). *A financial market model is arbitrage-free if and only if there exists at least one equivalent martingale measure $\mathbb{Q}$ for S_t^1 .*

Proof. For the proof, we refer to [Sch12]. □

We are now interested in the following problem: if we introduce a new financial option to the financial market model and claim that the new financial market should not allow any arbitrage opportunities, what is then the price of this option? For our purposes, we only will consider *European* options. These are options on the asset S^1 where the payoff of the option is only dependent on the terminal value S_T^1 . In mathematical terms, we have the following definition.

Definition 2.5 (European option). An *European option* or payoff or contingent claim is a random variable $H \in L_+^0(\mathcal{F}_T)$, where $L_+^0(\mathcal{F}_T)$ denotes the space of all positive $\mathcal{F}_T$ -measurable random variables.

The random variable H describes the net payoff the owner of the option gets at time T . H should be greater or equal to zero is then a natural condition. Definition 2.5 can be extended to also support *American* options. Such options allow a certain degree of freedom to the owner of the option in choosing the time of the payoff.

To price a financial option H we try to construct a trading strategy in the market such that it has the same value as the option at all times $t \in [0, T]$. By arbitrage-free arguments the value of this strategy

should then coincide with the value of the option. The price of the option is then equal to the initial value of the trading strategy.

We provide some mathematical definitions for this concept. A payoff H (discounted) is called attainable if there exists a self-financing replicating strategy for it, i.e. a trading strategy ϕ such that $V_T(\phi) = H$ $\mathbb{P}$ -a.s. A financial market model is called *complete* if every payoff H is attainable. A sufficient condition for a financial market model to be complete is given by the following theorem.

Theorem 2.2 (Complete markets). *An arbitrage-free financial market model is complete if and only if the equivalent martingale measure is unique. For every payoff $H \in L_+^0(\mathcal{F}_T)$, there exists a unique price process denoted by V_t^H which admits no arbitrage. This price process is given by:*

$$V_t^H = \mathbb{E}_{\mathbb{Q}}[H | \mathcal{F}_t] \quad \forall t \in [0, T].$$

In particular, the option price is equal to

$$V_0^H = \mathbb{E}_{\mathbb{Q}}[H].$$

We have obtained a strong characterization for financial market models. If there exists a unique equivalent martingale measure $\mathbb{Q}$ for the process S_t^1 , the market is arbitrage-free and complete, and the price of any payoff H can be found using theorem 2.2.

We will now prove that our current market model is arbitrage-free, by showing that there exists an equivalent martingale measure. If we define

$$W_t^* = W_t + \frac{\mu - r}{\sigma}t = W_t + \int_0^t \lambda ds,$$

where $\lambda := \frac{\mu - r}{\sigma}$ denotes the instantaneous risk premium (also called the instantaneous *Sharpe ratio*), we can rewrite equation (2.7) to

$$(2.10) \quad \frac{dS_t^1}{S_t^1} = \sigma dW_t^*.$$

We now define the probability measure $\mathbb{Q}$ as

$$\frac{d\mathbb{Q}}{d\mathbb{P}} \Big|_{\mathcal{F}_T} = \mathcal{E} \left(- \int_0^T \lambda dW \right) = \exp \left(-\lambda W_T - \frac{1}{2} \lambda^2 T \right),$$

where $\mathcal{E}(\cdot)$ denotes the stochastic exponential function (see [Pro04]). By definition, the probability measure $\mathbb{Q}$ is equivalent to $\mathbb{P}$ (that is, $\mathbb{P}[A] = 0$ if and only if $\mathbb{Q}[A] = 0 \forall A \in \mathcal{F}_T$). We then get that W^* is a $\mathbb{Q}$ -Brownian motion by Girsanov's theorem. Looking at equation (2.10), we can represent the solution S_t^1 as a stochastic exponential:

$$S_t^1 = \mathcal{E} \left(\int_0^t \sigma dW_s^* \right) = \exp \left(\sigma W_t^* - \frac{1}{2} \sigma^2 t \right).$$

So S_t^1 is actually a geometric Brownian motion under the measure $\mathbb{Q}$, hence a $\mathbb{Q}$ -martingale. Thus, $\mathbb{Q}$ is an equivalent martingale measure for S^1 and our market model is arbitrage-free.

It can be proven that our model is complete as well. This result heavily depends on the assumption that the filtration is generated by a Brownian motion, so the choice of the filtration is actually very important. The proof for this statement will not be discussed here, but can be found in [Sch12]. We use theorem 2.2, which provides us the link to option pricing.

Let H be a payoff and $V_t := \mathbb{E}_{\mathbb{Q}}[H | \mathcal{F}_t]$ its value process for $t \in [0, T]$. The process V_t thus represents the value of the payoff at time t . By Itô's representation theorem we get

$$(2.11) \quad V_t = \mathbb{E}_{\mathbb{Q}}[H] + \int_0^t \psi_s^H dW_s^*,$$

where ψ^H is a predictable process with the property that $\int_0^t (\psi_s^H)^2 ds < \infty$ $\mathbb{Q}$ -a.s for each $t \geq 0$ such that $\int \psi^H dW^*$ is a $\mathbb{Q}$ -martingale. From equation (2.10) and the self-financing condition (2.8) we see that if we define

$$(2.12) \quad \begin{aligned} \pi_t^H &= \frac{\psi_t^H}{S_t^1 \sigma}, \\ \eta_t^H &= V_t - \pi_t^H S_t^1, \\ V_0 &= \mathbb{E}_{\mathbb{Q}}[H], \end{aligned}$$

the strategy $\phi^H = (V_0, \pi_t^H, \eta_t^H)$ can be interpreted a trading strategy that is self-financing and replicates the payoff H .

How do we obtain this replicating strategy? We will look at the case when $H = h(S_T^1)$. Notice we can rewrite

$$(2.13) \quad S_T^1 = S_t^1 \frac{S_T^1}{S_t^1} = S_t^1 \exp \left(\sigma(W_T^* - W_t^*) - \frac{1}{2}\sigma^2(T-t) \right),$$

The price of the payoff H is given by its value process V_t (we drop the dependency on H for notational convenience). This process is given by the equation

$$V_t = \mathbb{E}_{\mathbb{Q}} [h(S_T^1) | \mathcal{F}_t].$$

Notice that in equation (2.13), the factor S_t^1 obviously is $\mathcal{F}_t$ -measurable and that $W_T^* - W_t^*$ is independent of $\mathcal{F}_t$ and $\mathcal{N}(0, T-t)$ -distributed under $\mathbb{Q}$ since W^* is a $\mathbb{Q}$ -Brownian motion. Therefore it follows from (2.13) that

$$(2.14) \quad V_t = \mathbb{E}_{\mathbb{Q}} [h(S_T^1) | \mathcal{F}_t] = v(t, S_t^1).$$

The value of the option H at time t is thus a function of the time t and the value of S_t^1 . This function $v(t, x)$ is given by:

$$\begin{aligned} v(t, x) &= \mathbb{E}_{\mathbb{Q}} \left[h \left(x \exp \left(\sigma(W_T^* - W_t^*) - \frac{1}{2}\sigma^2(T-t) \right) \right) \right] \\ &= \mathbb{E}_{\mathbb{Q}} \left[h \left(x \exp \left(\sigma\sqrt{(T-t)}Y - \frac{1}{2}\sigma^2(T-t) \right) \right) \right] \\ &= \int_{-\infty}^{\infty} h \left(x e^{\left(\sigma\sqrt{(T-t)}y - \frac{1}{2}\sigma^2(T-t) \right)} \right) \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}y^2} dy, \end{aligned}$$

where Y follows a $\mathcal{N}(0, 1)$ distribution under $\mathbb{Q}$. If we assume this function is sufficiently smooth to apply Itô's lemma, we obtain:

$$\begin{aligned} dV_t &= dv(t, S_t^1) \\ &= v_t(t, S_t^1)dt + v_x(t, S_t^1)dS_t^1 + \frac{1}{2}v_{xx}(t, S_t^1)d\langle S^1 \rangle_t \\ &= v_x(t, S_t^1)\sigma S_t^1 dW_t^* + \left(v_t(t, S_t^1) + \frac{1}{2}v_{xx}(t, S_t^1)\sigma^2(S_t^1)^2 \right) dt, \end{aligned}$$

since for sharp bracket process of S_t^1 it holds that $\langle S_t^1 \rangle_t = \int_0^t (S_s^1)^2 \sigma^2 dW_s^*$. However, by definition V_t is a local $\mathbb{Q}$ -martingale, so the two terms in the right hand side of the previous equation also must be continuous local martingales. The dt -integral is of finite variation and since any continuous local martingale of finite variation must vanish, we get that:

$$dV_t = v_x(t, S_t^1)\sigma S_t^1 dW_t^* = v_x(t, S_t^1)dS_t^1 = v_x(t, S_t^1)S_t^1 \sigma dW_t^*.$$

Comparing this result with equation (2.11) and equation (2.12) we obtain the *hedging strategy* explicitly as:

$$\pi_t^H = \frac{\partial v}{\partial x}(t, S_t^1),$$

which is commonly referred to as the *delta* of the option. Furthermore, from the above local martingale argument we can obtain that the function v should satisfy the following PDE:

$$0 = \frac{\partial v}{\partial t} + \frac{1}{2}\sigma^2 x^2 \frac{\partial^2 v}{\partial x^2}, \quad \text{on } (0, T) \times (0, \infty),$$

with boundary condition $v(T, x) = h(x)$ on $(0, \infty)$. Switching back to undiscounted units by using the relation:

$$v(t, x) = e^{-rt} \tilde{v}(t, x e^{rt}),$$

we obtain

$$0 = \frac{\partial \tilde{v}}{\partial t} + r\tilde{x} \frac{\partial \tilde{v}}{\partial \tilde{x}} + \frac{1}{2}\sigma^2 \tilde{x}^2 \frac{\partial^2 \tilde{v}}{\partial \tilde{x}^2} - r\tilde{v},$$

with the boundary condition $\tilde{v}(T, \tilde{x}) = \tilde{h}(\tilde{x})$. This is the famous *Black-Scholes* option pricing PDE.

In the case of a *call option* one can obtain an analytical solution for the price of this option. Remember that the payout of this option can be represented as the function $h(S_T^1) = \max(S_T^1 - K, 0)$ where $K \in \mathbb{R}$ denotes the discounted strike and S_T^1 the discounted spot price at maturity. By substitution of h in (2.14)

and by following the same reasoning as above we get that the value at time t of this option is given by its value process V_t :

$$V_t = \mathbb{E}_{\mathbb{Q}} \left[\left(x \exp \left(\sigma \sqrt{(T-t)} Y - \frac{1}{2} \sigma^2 (T-t) \right) - K \right)^+ \right] \Big|_{x=S_t^1},$$

where Y follows a $\mathcal{N}(0, 1)$ distribution. We will use the following lemma:

Lemma 2.1 (From [Sch12]). If $x > 0, a > 0, b \geq 0$ and Y follows a $\mathcal{N}(0, 1)$ distribution under $\mathbb{Q}$, then:

$$\mathbb{E}_{\mathbb{Q}} \left[\left(x e^{aY - \frac{1}{2}a^2} - b \right)^+ \right] = x \Phi \left(\frac{\log \frac{x}{b} + \frac{1}{2}a^2}{a} \right) - b \Phi \left(\frac{\log \frac{x}{b} - \frac{1}{2}a^2}{a} \right)$$

Proof. Notice that the expectation is equal to

$$(2.15) \quad \mathbb{E}_{\mathbb{Q}} \left[x e^{aY - \frac{1}{2}a^2} 1_{\{x e^{aY - \frac{1}{2}a^2} > b\}} \right] - b \mathbb{Q} \left[x e^{aY - \frac{1}{2}a^2} > b \right].$$

Where 1_A is the indicator function of the event A . We compute both parts separately. Starting with the expectation term and noting that $A := \{x e^{aY - \frac{1}{2}a^2} > b\} = \{Y > \frac{\log \frac{b}{x} + \frac{1}{2}a^2}{a}\}$, one obtains

$$\begin{aligned} \mathbb{E}_{\mathbb{Q}} \left[x e^{aY - \frac{1}{2}a^2} 1_A \right] &= \int_A x e^{ay - \frac{1}{2}a^2} \frac{1}{\sqrt{2\pi}} e^{-\frac{y^2}{2}} dy \\ &= \int_{\frac{\log \frac{b}{x} + \frac{1}{2}a^2}{a}}^{\infty} \frac{x}{\sqrt{2\pi}} e^{-\frac{(y-a)^2}{2}} dy \\ &= \frac{x}{\sqrt{2\pi}} \int_{-\infty}^{\frac{\log \frac{x}{b} + \frac{1}{2}a^2}{a}} e^{-\frac{u^2}{2}} du \\ &= x \Phi \left(\frac{\log \frac{x}{b} + \frac{1}{2}a^2}{a} \right), \end{aligned}$$

where we used the substitution $u = a - y$ and flipped the integration boundaries in the third equation. Computing the second term we get:

$$\begin{aligned} b \mathbb{Q} [A] &= b \left(1 - \mathbb{Q} \left[Y \leq \frac{\log \frac{b}{x} + \frac{1}{2}a^2}{a} \right] \right) \\ &= b \Phi \left(-\frac{\log \frac{b}{x} + \frac{1}{2}a^2}{a} \right) \\ &= b \Phi \left(\frac{\log \frac{x}{b} - \frac{1}{2}a^2}{a} \right), \end{aligned}$$

and this concludes the proof. $\square$

Applying the above lemma with $x = S_t^1, a = \sigma \sqrt{T-t}$ and $b = K$ and passing to undiscounted units we get the *Black-Scholes formula* for a call option:

$$\tilde{V}_t = \tilde{S}_t^1 \Phi(d_1) - \tilde{K} e^{-r(T-t)} \Phi(d_2),$$

where

$$d_{1,2} = \frac{\log \frac{\tilde{S}_t^1}{\tilde{K}} + (r \pm \frac{1}{2}\sigma^2)(T-t)}{\sigma \sqrt{T-t}}.$$

3. INTRODUCTION TO BSDEs

In this section we will discuss some usual assumptions and important theorems on BSDEs. Furthermore, we will study the specific case of linear BSDEs and relevant theorems. We will consider an example in the beforementioned market model and explain the link between BSDEs and the Black-Scholes formula. We will also consider an extended market model with a bid-ask spread for interest rates.

3.1. Definitions and theorems. Throughout this section we will work on the usual probability space $(\Omega, \mathcal{F}, (\mathcal{F}_t)_{0 \leq t \leq T}, \mathbb{P})$, where $\mathcal{F}$ denotes the right-continuous version of the filtration generated by a Brownian motion and augmented with the $\mathbb{P}$ -nullsets so that the probability space is complete. This is the same probability space as in section 2. We define the following sets:

- (1) $\mathbb{S}^2(0, T)$ denotes the set of $\mathbb{R}$ -valued progressively measurable processes Y such that:

$$\mathbb{E} \left[\sup_{0 \leq t \leq T} |Y_t|^2 \right] < \infty.$$

- (2) $\mathbb{H}^2(0, T)^d$ denotes the set of $\mathbb{R}^d$ -valued progressively measurable processes Z such that:

$$\mathbb{E} \left[\int_0^T |Z_t|^2 dt \right] < \infty.$$

We will consider the following BSDE:

$$(3.1) \quad -dY_t = f(t, Y_t, Z_t)dt - Z_t dW_t,$$

with terminal condition $Y_T = \xi$, where $\xi : \Omega \rightarrow \mathbb{R}^d$ is an $\mathcal{F}_T$ -measurable random variable that serves as a terminal condition and $f : \Omega \times [0, T] \times \mathbb{R} \times \mathbb{R}^d \rightarrow \mathbb{R}$ is the driver of the process. Furthermore, we assume that:

- (1) $\xi \in L^2(\Omega, \mathcal{F}_T, \mathbb{P}, \mathbb{R})$
- (2) f is progressively measurable and Lipschitz-continuous in y and z
- (3) $f(\cdot, 0, 0) \in \mathbb{H}^2(0, T)^d$

We call f and ξ the *standard parameters* for the BSDE. A solution to (3.1) is a pair $(Y_t, Z_t) \in \mathbb{S}^2(0, T)^d \times \mathbb{H}^2(0, T)^d$ satisfying (3.1). Under the above mentioned assumptions one can prove the following theorem:

Theorem 3.1 ([Pha09]). *Given a pair of standard parameters (f, ξ) , the BSDE (3.1) has a unique solution (Y_t, Z_t) .*

A proof of this theorem can be found, for example, in [Pha09].

Example 3.1 (BSDE for Black-Scholes market model). We will show an example of a model for a financial market where a BSDE naturally arises. We will look at the following market model, which is the d -dimensional generalization of the Black-Scholes model discussed in section 2:

- (1) There is a riskless asset which behaviour is modelled by:

$$\tilde{S}_t^0 = \tilde{S}_t^0 r_t dt,$$

where r_t denotes the interest rate on the market at time t .

- (2) We model d risky assets whose dynamics are modelled by the following (forward) SDEs:

$$d\tilde{S}_t^i = \tilde{S}_t^i \left(\mu_t^i dt + \sum_{j=1}^d \sigma_t^{i,j} dW_t^j \right),$$

where μ_t^i is the so called *drift term*, $\sigma_t^{i,j}$ is the covariance between stock i and j and $\sqrt{\sum_{j=1}^d (\sigma_t^{i,j})^2}$ is the *volatility* of stock i .

- (3) We assume the existence of a predictable and bounded process λ_t (called the *risk premium*) such that

$$\mu_t - r_t \mathbf{1} = \sigma_t \lambda_t,$$

where μ_t is the vector $\mu_t = (\mu_t^1, \dots, \mu_t^d)$ and σ_t is the volatility matrix with entries $[\sigma_t]_{i,j} = \sigma_t^{i,j}$.

Let π_t denote the row vector of the holdings in each asset. We will only consider *self-financing* strategies. The wealth process Y_t then satisfies:

$$\begin{aligned} dY_t &= \underbrace{\sum_{i=1}^d \pi_t^i \frac{d\tilde{S}_t^i}{\tilde{S}_t^i}}_{\text{value from risky assets}} + \underbrace{\left(Y_t - \sum_{i=1}^d \pi_t^i\right) \frac{d\tilde{S}_t^0}{\tilde{S}_t^0}}_{\text{value from riskless asset}} \\ &= \pi_t (\sigma_t dW_t + b_t dt) + (Y_t - \pi_t \mathbf{1}) r_t dt \\ &= (r_t Y_t + \pi_t \sigma_t \lambda_t) dt + \pi_t \sigma_t dW_t. \end{aligned}$$

Let $Z_t := \pi_t \sigma_t$, then this equation is equal to

$$(3.2) \quad -dY_t = -r_t Y_t dt - Z_t \lambda_t dt - Z_t dW_t,$$

which is a linear BSDE. When we face the problem of contingent claim valuation, we define the terminal condition for the wealth process to be equal to the option payoff at maturity. The process Y_t then represents the value of the claim at time t and the process π_t represents the hedging portfolio.

To obtain our hedging process π_t we need to invert the volatility matrix σ_t , so another assumption we will make is that we assume that σ_t is invertible. In this case our market is actually *complete* in the sense that any option payoff $H \in L_+^0(\mathcal{F}_T)$ is attainable. Because the BSDE (3.2) has a unique solution (Y_t, Z_t) and since we assume that σ_t is invertible, we can obtain the corresponding hedging portfolio π_t for arbitrary H . The financial market model in this example is actually the same model discussed in section 2, the Black-Scholes model.

An extension of the usual Feynman-Kac formula gives a relation between BSDE and semilinear PDEs, which is highly relevant for the purposes of this theses. We will look at semilinear PDEs of the form:

$$(3.3) \quad -\frac{\partial v}{\partial t} - \mathcal{L}v - f(t, x, v, \bar{\sigma}^T D_x v) = 0,$$

with terminal condition $v(T, x) = g(x)$ and where $\mathcal{L}$ stands for the differential operator:

$$\mathcal{L}v = \bar{\mu}(x) D_x v(t, x) + \frac{1}{2} \text{Tr}(\bar{\sigma}(x) \bar{\sigma}^T(x) D_x^2 v(t, x)),$$

with $\bar{\sigma}^T$ denoting the transpose of $\bar{\sigma}$. This PDE has a probabilistic representation by means of the BSDE (3.1) with driver $f(t, X_t, Y_t, Z_t)$ and terminal condition $Y_T = g(X_T)$ and the forward SDE (taking values in $\mathbb{R}^d$)

$$(3.4) \quad dX_t = \bar{\mu}(X_t) dt + \bar{\sigma}(X_t) dW_t,$$

where X_0 is given.

Theorem 3.2 (Feynman-Kac analogue, [Pha09]). *Let v be a classical solution to (3.3), satisfying some growth conditions. Let $Y_t := v(t, X_t)$ and $Z_t := \bar{\sigma}^T(X_t) D_x v(t, X_t)$ for $0 \leq t \leq T$. Then the pair (Y_t, Z_t) is a solution to the BSDE*

$$(3.5) \quad -Y_t = f(t, X_t, Y_t, Z_t) dt - Z_t dW_t,$$

with the usual assumptions.

Proof. We apply Itô's lemma to $v(t, X_t)$:

$$\begin{aligned} dv(t, X_t) &= v_t(t, X_t) dt + \bar{\sigma}(X_t)^T D_x v(t, X_t) dW_t + \mathcal{L}v(t, X_t) dt \\ &= (v_t(t, X_t) + \mathcal{L}v(t, X_t)) dt + \bar{\sigma}(X_t)^T D_x v(t, X_t) dW_t \\ &= -f(t, X_t, v(t, X_t), \bar{\sigma}(X_t)^T D_x v(t, X_t)) dt + \bar{\sigma}(X_t)^T D_x v(t, X_t) dW_t \\ &= -f(t, X_t, Y_t, Z_t) dt + Z_t dW_t. \end{aligned}$$

□

A similar theorem in the other direction also exists, but will not be discussed here. The combination of equations (3.4) and (3.5) and an initial condition for the forward SDE and a terminal condition for the BSDE is called a *forward-backward stochastic differential equation* (FBSDE).

3.2. Linear BSDEs. In this section we will consider the case where the driver of a BSDE is a linear function in y and z . We assume the driver function is of the form

$$f(t, Y_t, Z_t) = A_t Y_t + B_t Z_t + C_t,$$

where A_t, B_t and C_t are assumed to be progressively measurable processes which are $\mathbb{R}$ and $\mathbb{R}^d$ -valued respectively, and C_t is a process in $\mathbb{H}^2(0, T)$. It turns out that we can actually solve this type of BSDE explicitly.

Proposition 3.3. *Consider the linear BSDE*

$$-dY_t = (A_t Y_t + B_t Z_t + C_t) dt - Z_t dW_t, \quad Y_t = \xi,$$

with the usual assumptions. The unique solution (Y_t, Z_t) is then given by

$$\Gamma_t Y_t = \mathbb{E} \left[\Gamma_T \xi + \int_t^T \Gamma_s C_s ds \mid \mathcal{F}_t \right],$$

where Γ_t is the solution to the linear SDE:

$$d\Gamma_t = \Gamma_t (A_t dt + B_t dW_t), \quad \Gamma_0 = 1,$$

and Z_t is given by Itô's representation theorem of the martingale

$$M_t := \mathbb{E} \left[\Gamma_T \xi + \int_0^T \Gamma_s C_s ds \mid \mathcal{F}_t \right].$$

The proof of this proposition can also be found in [Pha09]. Note that we can write down an explicit formula for the process Γ_t in the above proposition by using the stochastic exponential. We get

$$\begin{aligned} \Gamma_t &= \mathcal{E}(A_t dt + B_t dW_t) \\ &= \exp \left\{ \left(A_t - \frac{1}{2} B_t^2 \right) dt + B_t dW_t \right\}, \end{aligned}$$

where we used that $\mathcal{E}(X_t) = \exp \{ X_t - \frac{1}{2} \langle X_t \rangle \}$ if X_t is a continuous semimartingale. Since B_t is predictable and continuous and W_t is a martingale, we get by properties of the stochastic integral that the process $\int B_t dW_t$ is a continuous local martingale. Furthermore, the process $\int A_t dt$ is of finite variation which gives us that the process $\int A_t dt + \int B_t dW_t$ is a continuous semimartingale.

Example 3.2 (Contingent claim valuation). The setting is the same as in example 3.1. The problem of contingent claim valuation boils down to solving the BSDE

$$(3.6) \quad -dY_t = -r_t Y_t dt - Z_t \lambda_t dt - Z_t dW_t,$$

with terminal condition $Y_T = \tilde{H}$, where $\tilde{H}$ is the undiscounted contingent claim payoff. Y_t will represent our portfolio value and $Z_t = \pi_t \sigma_t$ will give us our hedging portfolio (by inversion of σ_t).

To simplify this example and to show the analogy with the Black-Scholes model we will assume from now on that $r_t = r$, $\lambda_t = \lambda$ and $\sigma_t = \sigma$, i.e. we assume a constant interest rate, risk premium and volatility. Furthermore we assume that $d = 1$. This corresponds to a market where there is only a bank account, respectively one asset ($\tilde{S}_t^1 = \tilde{S}_t$) and an option on that asset with undiscounted payoff $\tilde{H}$.

Notice that the driver of BSDE (3.2) is linear so that we can apply proposition 3.3. We then obtain for Γ_t the expression:

$$\Gamma_t = \exp \left\{ \left(-r - \frac{1}{2} \lambda^2 \right) dt - \lambda dW_t \right\}.$$

Applying proposition 3.3 now gives us:

$$\begin{aligned} Y_t &= \frac{1}{\Gamma_t} \mathbb{E}_{\mathbb{P}} [\Gamma_T H \mid \mathcal{F}_t] \\ &= \mathbb{E}_{\mathbb{P}} \left[\frac{\Gamma_T}{\Gamma_t} H \mid \mathcal{F}_t \right] \\ &= \mathbb{E}_{\mathbb{P}} \left[\exp \left\{ \left(-r - \frac{1}{2} \lambda^2 \right) (T - t) - \lambda (W_T - W_t) \right\} H \mid \mathcal{F}_t \right], \end{aligned}$$

where we used the fact that Γ_t is adapted to $\mathcal{F}_t$. If we now define

$$\begin{aligned} \left. \frac{d\mathbb{Q}}{d\mathbb{P}} \right|_{\mathcal{F}_T} &:= \mathcal{E} \left(- \int_0^T \lambda dW_s \right) \\ &= \exp \left\{ - \frac{1}{2} \int_0^T \lambda^2 ds - \int_0^T \lambda dW_s \right\}, \end{aligned}$$

which is the usual risk-neutral measure, we see that its density process D_t is equal to

$$D_t = \mathbb{E}_{\mathbb{P}} \left[\left. \frac{d\mathbb{Q}}{d\mathbb{P}} \right| \mathcal{F}_t \right] = \mathcal{E} \left(- \int_0^t \lambda dW_s \right) = \exp \left\{ -\lambda W_t - \frac{1}{2} \lambda^2 t \right\},$$

where we used the fact that $\mathcal{E} \left(- \int_0^t \lambda dW_s \right)$ is a geometric Brownian motion, hence it is a martingale. By Bayes' rule, we now get that the conditional expectation under the measure $\mathbb{P}$ is equal to:

$$Y_t = \mathbb{E}_{\mathbb{Q}} \left[\exp \{ -r(T-t) \} \tilde{H} \mid \mathcal{F}_t \right].$$

The value of the contingent claim is thus the discounted payoff under the usual risk-neutral measure, which is exactly the result from the Black-Scholes model.

The BSDE approach to the valuation of a contingent claim thus produces the exact same result in our market model as the Black-Scholes formula. However, unlike the Black-Scholes model, we do not need a change of measure and do all of our calculations under the real world $\mathbb{P}$ -measure. We can use theorem 3.2 to obtain the Black-Scholes PDE:

Example 3.3 (Black-Scholes PDE). If we consider the BSDE in equation (3.2) we get by theorem 3.2 that the pair

$$\begin{cases} Y_t := v(t, \tilde{S}_t) \\ Z_t := \tilde{S}_t \sigma \frac{\partial v}{\partial s} \left(t, \tilde{S}_t \right) \end{cases},$$

is a solution to the BSDE. The function $v(t, x)$ is a solution to the PDE (after simplifications):

$$0 = \frac{\partial v}{\partial t} + \frac{1}{2} \sigma^2 \tilde{S}^2 \frac{\partial^2 v}{\partial \tilde{S}^2} + r \tilde{S} \frac{\partial v}{\partial \tilde{S}} - r v,$$

with terminal condition $v(T, \tilde{S}_T) = H(\tilde{S}_T)$. This is exactly the Black-Scholes PDE!

Example 3.4 (Market with bid-ask spread for interest rates). In this example we will extend our initial market model. In this model, an investor borrows at an interest rate R_t and lends at a rate $r_t \leq R_t$ for all $t \in [0, T]$. This means that if $Y_t > \sum_{i=1}^d \pi_t^i$ we invest in the bond at an interest rate r_t , and we borrow at an interest rate R_t if $Y_t < \sum_{i=1}^d \pi_t^i$. The dynamics of the self-financing portfolio are now given by

$$\begin{aligned} dY_t &= \sum_{i=1}^d \pi_t^i \frac{d\tilde{S}_t^i}{\tilde{S}_t^i} + \left(Y_t - \sum_{i=1}^d \pi_t^i \right)^+ r_t dt - \left(Y_t - \sum_{i=1}^d \pi_t^i \right)^- R_t dt \\ &= \pi_t (\sigma_t dW_t + \mu_t dt) + (Y_t - \pi_t \mathbf{1}) r_t dt - (R_t - r_t) (Y_t - \pi_t \mathbf{1})^- dt \\ &= r_t Y_t dt + \pi_t \sigma_t \lambda_t^r dt + \pi_t \sigma_t dW_t - \underbrace{(R_t - r_t) (Y_t - \pi_t \mathbf{1})^- dt}_{\text{extra cost when borrowing}}. \end{aligned}$$

The driver of this BSDE is still Lipschitz-continuous, but not linear. We will, in general, not be able to find an explicit solution to this BSDE. Clearly, we need a numerical method to approximate solutions to these BSDEs and we will develop such a method in the next section.

4. NUMERICAL METHOD FOR BSDEs

In this section we will develop a numerical method which approximates the solution of a BSDE. We will look at the discretization of the BSDE and at the approximation of conditional expectations by discretizing the Brownian motion.

Throughout this section we will work with the following two equations:

$$(4.1) \quad \begin{cases} Y_t = g(X_T) + \int_t^T f(s, Y_s, Z_s) ds - \int_t^T Z_s dW_s, \\ X_t = X_0 + \int_0^t \bar{\mu}(X_s) ds + \int_0^t \bar{\sigma}(X_s) dW_s, \end{cases}$$

where we represented the BSDE in integral form, instead of differential form. Equation (4.1) is a combination of a forward SDE and a BSDE, where the terminal condition of the BSDE is now allowed to be dependent on the process X_t . As mentioned earlier, such type of equations is called a *forward-backward stochastic differential equation* (FBSDE). Notice that all of the examples discussed in section 2 were in fact such systems of a forward SDE and BSDE, where the forward SDE modelled the stock behaviour and the backward SDE modelled the portfolio dynamics.

In equation (4.1), we also left out the dependence of the driver f on the forward process X_t , since in all examples we will discuss the driver function will not depend on the process X_t . However, the numerical method developed in this section can easily be extended to support the dependency of the function f on the process X_t .

For the rest of this section, let Π be a partition of time points $0 = t_0 < t_1 < \dots < t_{\mathcal{M}} = T$ of $[0, T]$, with a fixed time step $\Delta t := t_{i+1} - t_i$. We will use the notation $X_m = X_{t_m}$, $Y_m = Y_{t_m}$, $Z_m = Z_{t_m}$ and $W_m = W_{t_m}$ and set $\Delta W_m = W_{m+1} - W_m$.

4.1. Discretization of forward SDE. In this section we will look at a method that approximates the numerical solution of the forward SDE in equation (4.1). To analyze the behaviour of the numerical methods we will develop throughout this thesis, we introduce the following definitions:

Definition 4.1 (Strong convergence of order α). A time-discretized approximation X_m^π converges to the stochastic process X in the strong sense with order α if there exists a constant $C \in \mathbb{R}$ such that:

$$\mathbb{E}[|X_m^\pi - X_m|] \leq C \Delta t^\alpha.$$

Definition 4.2 (Weak convergence of order α). A time-discretized approximation X_m^π converges to the stochastic process X in the weak sense with order α if there exists a constant $C \in \mathbb{R}$ such that for every infinitely often differentiable function $\phi : \mathbb{R}^d \rightarrow \mathbb{R}$ with at most polynomially growing derivatives it holds that:

$$|\mathbb{E}[\phi(X_m^\pi)] - \mathbb{E}[\phi(X_m)]| \leq C \Delta t^\alpha.$$

We will now look at a numerical scheme for the forward SDE. Just as in the deterministic case, we can derive numerical schemes by looking at the Taylor expansions of certain functions. We will now do the same, but in a stochastic setting. The following derivation can also be found in [KPS94].

We consider a function $k : \mathbb{R} \rightarrow \mathbb{R}$ that is twice differentiable and an Itô process X_t with drift $\bar{\mu}(X_t)$ and diffusion $\bar{\sigma}(X_t)$ also twice differentiable. Consider the integral form of the forward SDE in (4.1):

$$(4.2) \quad X_t = X_{t_0} + \int_{t_0}^t \bar{\mu}(X_s) ds + \int_{t_0}^t \bar{\sigma}(X_s) dW_s,$$

for $t \in [t_0, T]$. We further assume some suitable integrability conditions on $\bar{\mu}$ and $\bar{\sigma}$. By Itô's lemma we get:

$$\begin{aligned} k(X_t) &= k(X_{t_0}) + \int_{t_0}^t \bar{\mu}(X_s) k'(X_s) ds + \frac{1}{2} \bar{\sigma}(X_s)^2 k''(X_s) ds \\ &\quad + \int_{t_0}^t \bar{\sigma}(X_s) k'(X_s) dW_s. \end{aligned}$$

By defining:

$$(4.3) \quad L^0 k := \bar{\mu} k' + \frac{1}{2} \bar{\sigma}^2 k'',$$

$$(4.4) \quad L^1 k := \bar{\sigma} k',$$

we can rewrite (4.3) to:

$$(4.5) \quad k(X_t) = k(X_{t_0}) + \int_{t_0}^t L^0 k(X_s) ds + \int_{t_0}^t L^1 k(X_s) dW_s.$$

Consider now the integral form of X_t in (4.2). If we now substitute the functions $\bar{\mu}$ and $\bar{\sigma}$ for k in (4.5) and then substitute these expressions in equation (4.2), we get the so-called *Itô-Taylor* expansion of first order of X_t :

$$\begin{aligned} X_t &= X_{t_0} + \int_{t_0}^t \left(\bar{\mu}(X_{t_0}) + \int_{t_0}^s L^0 \bar{\mu}(X_p) dp + \int_{t_0}^s L^1 \bar{\mu}(X_p) dW_p \right) ds \\ &\quad + \int_{t_0}^t \left(\bar{\sigma}(X_{t_0}) + \int_{t_0}^s L^0 \bar{\sigma}(X_p) dp + \int_{t_0}^s L^1 \bar{\sigma}(X_p) dW_p \right) dW_s \\ &= X_{t_0} + \bar{\mu}(X_{t_0}) \int_{t_0}^t ds + \bar{\sigma}(X_{t_0}) \int_{t_0}^t dW_s + R \\ &= X_{t_0} + \bar{\mu}(X_{t_0}) (t - t_0) + \bar{\sigma}(X_{t_0}) (W_t - W_{t_0}) + R, \end{aligned}$$

where R is some remainder term consisting of double integrals. This Itô-Taylor expansion gives us the following *Euler-Maruyama scheme* as an approximation for the SDE in (4.2)

$$X_{m+1}^\pi = X_m^\pi + \bar{\mu}(X_m^\pi) \Delta t + \bar{\sigma}(X_m^\pi) \Delta W_m,$$

for $m = 0, \dots, \mathcal{M} - 1$ with $X_0^\pi = X_0$. One can prove the following theorem, concerning the Euler-Maruyama scheme:

Theorem 4.1 ([KPS94]). *In the above setting, the Euler-Maruyama method converges in the strong sense with order $\alpha = \frac{1}{2}$ and in the weak sense with order $\alpha = 1$.*

Example 4.1 (Approximation of GBM). To illustrate the meanings of both types of convergence, and to verify the above theorem, we discuss the simple example of approximating geometric Brownian motion. This process satisfies the following SDE:

$$\frac{dX_t}{X_t} = \mu dt + \sigma dW_t, \quad X_0 = x,$$

where W_t is a standard one-dimensional Brownian motion, $x, \mu \in \mathbb{R}$ and $\sigma \in \mathbb{R}_{>0}$. We can obtain a solution of this SDE by either using Itô's lemma or the stochastic exponential. Either way, we obtain the following expression for the solution:

$$X_t = X_0 \exp \left\{ \left(\mu - \frac{\sigma^2}{2} \right) t + \sigma W_t \right\}.$$

We approximate the geometric Brownian motion by the following Euler-Maruyama scheme:

$$X_{m+1} = X_m + \mu X_m \Delta t + \sigma X_m \Delta W_m.$$

The Brownian increments are simulated by using the `randn` (which produces normally distributed random variables) command in Matlab. The parameters used in this experiment are

$$\mu = 1, \sigma = 1, T = 1 \text{ and } X_0 = 1.5.$$

To illustrate weak convergence of the Euler-Maruyama method, we approximate the expectation of a geometric Brownian motion. One can derive from the lognormal properties of X_t that this expectation should be equal to $X_0 e^{\mu t}$. Figure 4.1 shows both the weak and strong error of this Euler-Maruyama approximation for different values of the time step Δt : We clearly see that the strong order of convergence is $\frac{1}{2}$ and the weak order of convergence is 1.

4.2. Discretization of the BSDE. In this section we will develop a discretization scheme for the backward SDE in (4.1). Firstly, we will approximate our terminal condition by substituting for X_t its approximation by the Euler scheme:

$$Y_{\mathcal{M}} = g(X_{\mathcal{M}}^\pi).$$

From theorem 3.2 we also obtain a terminal condition for the process Z_t . This theorem states that $(Y_t, Z_t) = (v(t, X_t), \bar{\sigma}(X_t)v_x(T, X_T))$ is the solution of equation (4.1). However, at time T we know that $v(t, X_T) = g(X_T)$ and hence we also know that $Z_T = \bar{\sigma}(X_T)g_x(X_T)$. The terminal condition for Z_t is obtained by substituting for X_T its Euler scheme:

$$Z_{\mathcal{M}} = \bar{\sigma}(X_{\mathcal{M}}^\pi)g_x(X_{\mathcal{M}}^\pi).$$

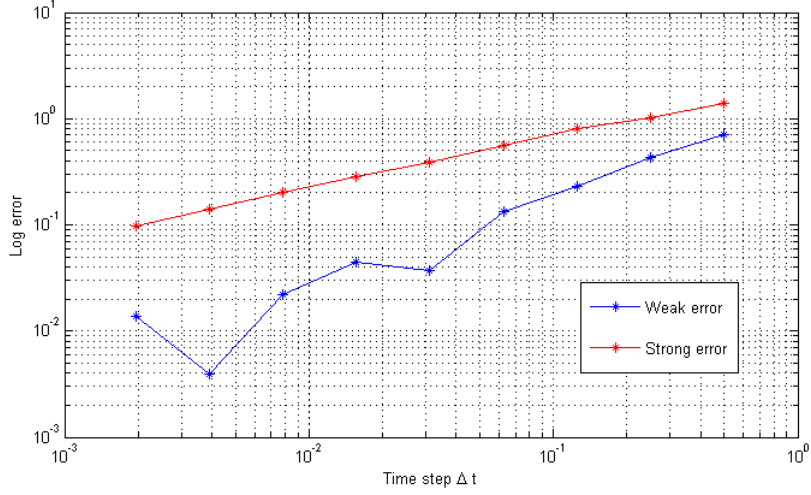


FIGURE 4.1. Log-log plot of the strong and weak error of the Euler-Maruyama approximation.

To develop a numerical scheme for the BSDE, we start with the following discrete version of the BSDE on the interval $[t_m, t_{m+1}]$:

$$(4.6) \quad Y_m = Y_{m+1} + \int_{t_m}^{t_{m+1}} f(s, Y_s, Z_s) ds - \int_{t_m}^{t_{m+1}} Z_s dW_s.$$

To obtain a numerical discretization for the process Y_t , we will take conditional expectations with respect to the filtration $\mathcal{F}_{t_m}$. For notational convenience we will define:

$$\mathbb{E}_m[\cdot] = \mathbb{E}[\cdot | \mathcal{F}_{t_m}].$$

Since Y_m and Z_m are adapted to $\mathcal{F}_{t_m}$, we get $\mathbb{E}_m[Z_m] = Z_m$ and $\mathbb{E}_m[Y_m] = Y_m$. Taking conditional expectations at both sides of equation (4.6) gives us:

$$\begin{aligned} Y_m &= \mathbb{E}_m[Y_{m+1}] + \mathbb{E}_m \left[\int_{t_m}^{t_{m+1}} f(s, Y_s, Z_s) ds \right] - \mathbb{E}_m \left[\int_{t_m}^{t_{m+1}} Z_s dW_s \right] \\ &= \mathbb{E}_m[Y_{m+1}] + \int_{t_m}^{t_{m+1}} \mathbb{E}_m[f(s, Y_s, Z_s)] ds - \mathbb{E}_m \left[\int_{t_m}^{t_{m+1}} Z_s dW_s \right], \end{aligned}$$

where we used Fubini's theorem to switch the order of integration in the first integral. Since we assumed that $Z_t \in \mathbb{H}_T^2(0, T)$, we get that the integral $\int_{t_m}^t Z_s dW_s$ (with $t \in [t_m, T]$) is a martingale and thus that the last integral in the above equation has conditional expectation zero with respect to $\mathcal{F}_{t_m}$. We thus arrive at:

$$(4.7) \quad Y_m = \mathbb{E}_m[Y_{m+1}] + \int_{t_m}^{t_{m+1}} \mathbb{E}_m[f(s, Y_s, Z_s)] ds.$$

We will approximate the integral in equation (4.7) with a theta approximation: a convex combination of an explicit (term at time t_{m+1}) and implicit (term at time t_m) term (also see [Ise09]). The resulting equation is:

$$\begin{aligned} Y_m &\approx \mathbb{E}_m[Y_{m+1}] + \Delta t \theta_1 f(t_m, Y_m, Z_m) \\ &\quad + \Delta t (1 - \theta_1) \mathbb{E}_m[f(t_{m+1}, Y_{m+1}, Z_{m+1})], \quad \theta_1 \in [0, 1]. \end{aligned}$$

Here we used the fact that the processes Y_t and Z_t are $\mathcal{F}_{t_m}$ -measurable. Such theta-methods are known to have order of convergence two when $\theta_1 = \frac{1}{2}$ and order of convergence equal to one otherwise. To obtain a numerical scheme for the process Z_t , we multiply both sides of equation (4.6) by ΔW_m and then take

conditional expectations:

$$\begin{aligned}
0 &= \mathbb{E}_m [\Delta W_m Y_{m+1}] + \mathbb{E}_m \left[\Delta W_m \int_{t_m}^{t_{m+1}} f(s, Y_s, Z_s) ds \right] \\
&\quad - \mathbb{E}_m \left[\Delta W_m \int_{t_m}^{t_{m+1}} Z_s dW_s \right] \\
&= \mathbb{E}_m [\Delta W_m Y_{m+1}] + \mathbb{E}_m \left[\int_{t_m}^{t_{m+1}} f(s, Y_s, Z_s) (W_{m+1} - W_s + W_s - W_m) ds \right] \\
&\quad - \mathbb{E}_m \left[\int_{t_m}^{t_{m+1}} dW_s \cdot \int_{t_m}^{t_{m+1}} Z_s dW_s \right] \\
&= \mathbb{E}_m [\Delta W_m Y_{m+1}] + \int_{t_m}^{t_{m+1}} \mathbb{E}_m [f(s, Y_s, Z_s) (W_{m+1} + W_s - W_s - W_m)] ds \\
&\quad - \int_{t_m}^{t_{m+1}} \mathbb{E}_m [Z_s] ds \\
&= \mathbb{E}_m [\Delta W_m Y_{m+1}] + \int_{t_m}^{t_{m+1}} \mathbb{E}_m [f(s, Y_s, Z_s)] \mathbb{E}_m [(W_{m+1} - W_s)] ds \\
&\quad + \int_{t_m}^{t_{m+1}} \mathbb{E}_m [f(s, Y_s, Z_s) (W_s - W_m)] ds - \int_{t_m}^{t_{m+1}} \mathbb{E}_m [Z_s] ds \\
&= \mathbb{E}_m [\Delta W_m Y_{m+1}] + \int_{t_m}^{t_{m+1}} \mathbb{E}_m [f(s, Y_s, Z_s) (W_s - W_m)] ds \\
&\quad - \int_{t_m}^{t_{m+1}} \mathbb{E}_m [Z_s] ds,
\end{aligned}$$

where we again used Fubini's theorem to rewrite the first integral and we used the Itô isometry to rewrite the last integral. We approximate both integrals by a theta-method:

$$\begin{aligned}
0 &= \mathbb{E}_m [\Delta W_m Y_{m+1}] + \Delta t (1 - \theta_2) \mathbb{E}_m [f(t_{m+1}, Y_{m+1}, Z_{m+1}) \Delta W_m] \\
&\quad - \Delta t \theta_2 Z_m - \Delta t (1 - \theta_2) \mathbb{E}_m [Z_{m+1}], \quad \theta_2 \in [0, 1],
\end{aligned}$$

where we used some adaptedness properties of Y_m and Z_m and the fact that $\Delta W_m | \mathcal{F}_{t_m} \sim \mathcal{N}(0, \Delta t)$. We can rewrite both equations for Y_m and Z_m to obtain the following numerical scheme:

$$\begin{aligned}
Y_{\mathcal{M}}^\pi &= g(X_{\mathcal{M}}^\pi), \quad Z_{\mathcal{M}}^\pi = \bar{\sigma}(X_{\mathcal{M}}^\pi) g_x(X_{\mathcal{M}}^\pi) \\
Z_m^\pi &= -\theta_2^{-1} (1 - \theta_2) \mathbb{E}_m [Z_{m+1}^\pi] + \frac{1}{\Delta t} \theta_2^{-1} \mathbb{E}_m [Y_{m+1}^\pi \Delta W_m] \\
&\quad + \theta_2^{-1} (1 - \theta_2) \mathbb{E}_m [f(t_{m+1}, Y_{m+1}^\pi, Z_{m+1}^\pi) \Delta W_m], \\
Y_m^\pi &= \mathbb{E}_m [Y_{m+1}^\pi] + \Delta t \theta_1 f(t_m, Y_m^\pi, Z_m^\pi) + \Delta t (1 - \theta_1) \mathbb{E}_m [f(t_{m+1}, Y_{m+1}^\pi, Z_{m+1}^\pi)],
\end{aligned}$$

for $m = \mathcal{M} - 1, \dots, 0$. Notice that when $\theta_1 = 0$ we obtain an explicit scheme for Y_m^π and that $\theta_1 \in (0, 1]$ results in an implicit scheme. The implicit scheme is solved by using Picard iterations, which will be discussed in the next section. To solve for Z_m^π we clearly need $\theta_2 \neq 0$ and we then obtain an explicit scheme for Z_m^π .

It can be shown that, since X_t^π is a Markov process and since the terminal condition is a function of $X_{\mathcal{M}}^\pi$, there are deterministic functions $y(t_m, x)$ and $z(t_m, x)$, such that

$$(4.8) \quad Y_m^\pi = y(t_m, X_m^\pi), \quad Z_m^\pi = z(t_m, X_m^\pi)$$

We can thus obtain these functions by using the following scheme:

$$\begin{aligned}
y(t_{\mathcal{M}}, x) &= g(x), \quad z(t_{\mathcal{M}}, x) = \bar{\sigma}(x) g_x(x) \\
z(t_m, x) &= -\theta_2^{-1} (1 - \theta_2) \mathbb{E}_m^x [z(t_{m+1}, X_{m+1}^\pi)] + \frac{1}{\Delta t} \theta_2^{-1} \mathbb{E}_m^x [y(t_{m+1}, X_{m+1}^\pi) \Delta W_m] \\
&\quad + \theta_2^{-1} (1 - \theta_2) \mathbb{E}_m^x [f(t_{m+1}, y(t_{m+1}, X_{m+1}^\pi), z(t_{m+1}, X_{m+1}^\pi)) \Delta W_m], \\
y(t_m, x) &= \mathbb{E}_m^x [y(t_{m+1}, X_{m+1}^\pi)] + \Delta t \theta_1 f(t_m, y(t_m, X_m^\pi), z(t_m, X_m^\pi)) \\
&\quad + \Delta t (1 - \theta_1) \mathbb{E}_m^x [f(t_{m+1}, y(t_{m+1}, X_{m+1}^\pi), z(t_{m+1}, X_{m+1}^\pi))],
\end{aligned}$$

for $m = \mathcal{M} - 1, \dots, 0$. Here we used the notation $\mathbb{E}_m^x [\cdot] = \mathbb{E} [\cdot | X_m^\pi = x]$.

It is proven in [ZWP09] that this scheme converges with order of convergence equal to two in the case $\theta_1 = \theta_2 = \frac{1}{2}$ and with order of convergence equal to one all other cases. For the case $\theta_1 = \theta_2 = 1$ we obtain a scheme encountered often in the literature (for example in [Zha04] and [BT04]) and which is referred to as the *Euler scheme* for the BSDE. In the case $\theta_1 \neq 0$ we are trying to find the fixed-point y of the equation

$$(4.9) \quad \begin{aligned} y &= \mathbb{E}_m^x [y(t_{m+1}, X_{m+1}^\pi)] + \Delta t \theta_1 f(t_m, y, z(t_m, X_m^\pi)) \\ &\quad + \Delta t (1 - \theta_1) \mathbb{E}_m^x [f(t_{m+1}, y(t_m, X_{m+1}^\pi), z(t_m, X_{m+1}^\pi))] \\ &= \Delta t \theta_1 f(t_m, y, z(t_m, X_m^\pi)) + h(t_m, X_{m+1}^\pi) \end{aligned}$$

Assuming the driver function is bounded and Lipschitz continuous in the variable y with Lipschitz constant equal to L , we get that for time steps Δt , such that $\Delta t \theta_1 L < 1$, a unique fixed-point of equation (4.9) exists by the Banach fixed-point theorem. It also follows that in this case the iteration converges to this fixed-point for any initial guess y_0 . The rate of convergence is equal to $\Delta t \theta_1 L$, so it is dependent on the size of the Lipschitz constant of the function f .

To use this theta-discretization, we need to approximate the conditional expectations $\mathbb{E}_m^x[\cdot]$. These conditional expectations will be approximated by using a binomial tree approximation.

4.3. Binomial tree approximation. Binomial trees have been used for decades in option pricing problems. They are an attractive tool because they are easy to understand and still provide accurate numerical approximations, at least to most vanilla option pricing problems. In the binomial option pricing model we are also approximating conditional expectations of the form

$$\mathbb{E}_{\mathbb{Q}}[H | \mathcal{F}_t],$$

where H represents the discounted option payoff, $\mathbb{Q}$ the risk-neutral martingale measure and $\mathcal{F}_t$ the filtration on the market. This suggests we could use some sort of binomial method to approximate the conditional expectations in our time-discretization of the FBSDE. Our final algorithm will simulate the driving Brownian motion and the FSDE in a forward manner and then approximate the BSDE in a backwards fashion.

Firstly, we will approximate the Brownian motion that drives the FBSDE by a scaled random walk:

$$(4.10) \quad W_m \approx \sqrt{\Delta t} \sum_{j=1}^m \epsilon_j = W_{m-1} + \sqrt{\Delta t} \epsilon_m$$

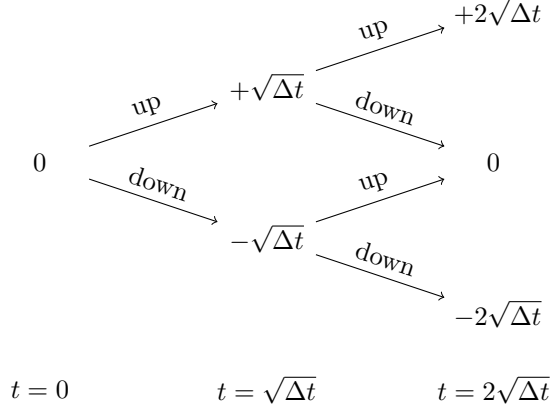
where we still use the notation from the previous section and where the ϵ_j are independent and such that:

$$\mathbb{P}(\epsilon_j = 1) = \mathbb{P}(\epsilon_j = -1) = \frac{1}{2}.$$

As an initial value, set $W_0 = 0$. If we define the process W_t^π as the process equal to W_m on the time points t_m and interpolate in a linear manner on the time intervals $[t_k, t_{k+1})$ for $k \in \{0, 1, \dots, \mathcal{M} - 1\}$, the central limit theorem guarantees the convergence of this process W_t^π to a $\mathcal{N}(0, t)$ -distributed random variable.

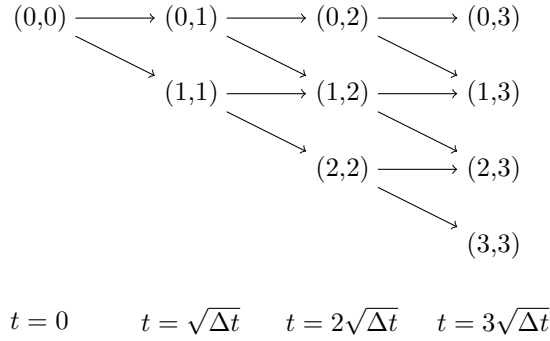
Since, furthermore $W_0^\pi = 0$, $W_t^\pi - W_s^\pi$ is $\mathcal{N}(0, t - s)$ distributed, W_t^π has independent increments and W_t^π is continuous, we get that W_t^π converges in distribution to a standard Brownian motion. This implies that this approximation is of weak order of convergence equal to one.

Because after every time step the approximation in (4.10) can only attain two different values, it can be represented in a binomial tree as follows:



An attractive property of this binomial tree is that it is *recombining*, a downward move followed by an upward move is the same as an upward move followed by a downward move. Such trees are computationally less expensive, because the number of nodes in a recombining tree at time point k is equal to $k + 1$ and for a non-recombining tree the number of nodes at time k is equal to 2^k . In practice, non-recombining trees are not used frequently because of the exploding number of nodes in the tree.

For easier description, we will use an upper-triangular form for the binomial tree and denote a node in the tree by a pair (i, j) , where i denotes the space movement of W_m and j denotes the current time point. This notation is borrowed from [Pen10]. The following figure illustrates our notation:



For example, node $(0,1)$ denotes a downwards move of the Brownian motion after the first time step and node $(1,1)$ denotes an upwards move of the Brownian motion. As a second example: node $(1,2)$ denotes either a downward move in the first time step and a upward move in the second time step, or a upward move in the first step and a downward move in the second step.

The forward SDE is approximated by the Euler-Maruyama approximation

$$(4.11) \quad X_{m+1}^\pi = X_m^\pi + \bar{\mu}(X_m^\pi)\Delta t + \bar{\sigma}(X_m^\pi)\Delta W_m,$$

as mentioned in sections 4.1 and 4.2. Since the Brownian motion is approximated by the random walk in (4.10), the approximation X_m^π also inherits the binomial tree structure of W_m . From (4.10) it also follows that $\Delta W_m = \sqrt{\Delta t}\epsilon_j$, so the binomial tree for equation (4.11) now becomes:

$$(4.12) \quad \begin{aligned} X_{i,j+1}^\pi &= X_{i,j}^\pi + \bar{\mu}(X_{i,j}^\pi)\Delta t - \bar{\sigma}(X_{i,j}^\pi)\sqrt{\Delta t} \\ X_{i+1,j+1}^\pi &= X_{i,j}^\pi + \bar{\mu}(X_{i,j}^\pi)\Delta t + \bar{\sigma}(X_{i,j}^\pi)\sqrt{\Delta t}, \end{aligned}$$

for all $j = 1, \dots, \mathcal{M}$ and $i = 1, \dots, j$ and where $X_{i,j}^\pi$ represents the value of the approximation X^π at node (i, j) . We will call approximation (4.12) the *binomial approximation* of the forward SDE.

This binomial tree approximation does not have to be recombining by definition, as this property is dependent on the functions $\bar{\mu}$ and $\bar{\sigma}$. Since for $\mathcal{M} = 15$ the amount of nodes in a non-recombining tree is already equal to $2^{15} = 32768$, such trees are not feasible from a computational point of view. A practical algorithm for BSDEs can not be constructed with such trees, therefore we will only consider a recombining binomial tree structure for approximation (4.12) in our algorithm.

Example 4.2 (Binomial tree approximation of GBM). In this example we will illustrate the approximation given in equation (4.12) and discuss the convergence of this approximation. The SDE which we will

approximate will be the following

$$\frac{dX_t}{X_t} = \mu dt + \sigma dW_t,$$

with $\mu \in \mathbb{R}$, $\sigma \in \mathbb{R}_{>0}$ and $X_0 \in \mathbb{R}$. This is the same SDE considered in example 4.1 and from that we know that the solution of the above SDE is a geometric Brownian motion. The binomial approximation to this SDE is

$$(4.13) \quad \begin{aligned} X_{i,j+1}^\pi &= X_{i,j} \left(1 + \mu \Delta t - \sigma \sqrt{\Delta t} \right) \\ X_{i+1,j+1}^\pi &= X_{i,j} \left(1 + \mu \Delta t + \sigma \sqrt{\Delta t} \right), \end{aligned}$$

where we use the same notation defined earlier. The values used for μ , σ and X_0 are

$$\mu = 1, \sigma = 1, T = 1 \text{ and } X_0 = 1.5,$$

and we set the time step equal to $dt = \frac{T}{h}$ where $h = 10 \cdot k$ with $k = 1, \dots, 1000$. In each time step we use $M = 200$ simulations. Figure 4.2 shows the value of the weak error for each time step: The convergence

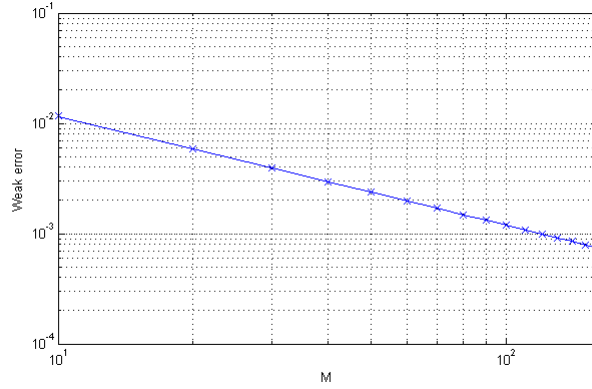


FIGURE 4.2. Plot of the weak error of the binomial tree approximation.

matches with what we already know: this binomial approximation of the Brownian motion results only in weak convergence, and the order of weak convergence is equal to one. Since we are only interested in computing *conditional expectations*, we see from the definition of the weak error that weak convergence is sufficient for our goals.

Since we are now able to compute the terminal values of the process X_t at each terminal node of the binomial tree, we can compute the terminal conditions for the processes Y_t and Z_t of our BSDE according to the numerical discretization in (4.9). This enables us to compute the values of Y_t and Z_t at earlier nodes in the tree.

Suppose we want to approximate the value of Z_t at node (i, j) by scheme (4.9). One of the quantities we then have to compute is the conditional expectation

$$\mathbb{E}_m [Y_{j+1}^\pi \Delta W_j] = \mathbb{E}_m^x [y(t_{j+1}, X_{j+1}^\pi) \Delta W_j] = \sqrt{\Delta t} \mathbb{E}_m^x [y(t_{j+1}, X_{j+1}^\pi) \epsilon_j].$$

The conditional expectation with respect to the process X_t tells us in which node we currently are, say this is node (i, j) . Since we are working within a binomial tree, the discretized Brownian motion can only move up $(+\sqrt{\Delta t})$ or down $(-\sqrt{\Delta t})$ with probability equal to $\frac{1}{2}$ each in the next time step. There are thus only two possible values for the random variable $Y_{j+1} = y(t_{j+1}, X_{j+1}^\pi)$ in this model, given the current value of X_j and they each occur with probability equal to $\frac{1}{2}$. Using the definition of the conditional expectation, we can thus approximate:

$$\mathbb{E}_m^x [Y_{j+1}^\pi \Delta W_j] \approx \frac{\sqrt{\Delta t}}{2} (Y_{i+1,j+1}^\pi - Y_{i,j+1}^\pi).$$

Approximating the other conditional expectations in our discretization, we finally arrive at the following backward scheme:

$$\begin{aligned}
Y_{i,\mathcal{M}}^\pi &= g(X_{i,\mathcal{M}}^\pi), \quad Z_{i,\mathcal{M}}^\pi = \bar{\sigma}(X_{i,\mathcal{M}}^\pi)g_x(X_{i,\mathcal{M}}^\pi) \quad \forall i = 1, \dots, \mathcal{M} \\
Z_{i,j}^\pi &= \frac{\theta_2 - 1}{2\theta_2} [Z_{i,j+1}^\pi + Z_{i+1,j+1}^\pi] + \frac{1}{2\theta_2\sqrt{\Delta t}} [Y_{i,j+1}^\pi - Y_{i+1,j+1}^\pi] + \\
&\quad \frac{1 - \theta_2}{2\theta_2} \sqrt{\Delta t} [f(t_{j+1}, Y_{i,j+1}^\pi, Z_{i,j+1}^\pi) - f(t_{j+1}, Y_{i+1,j+1}^\pi, Z_{i+1,j+1}^\pi)] \\
Y_{i,j}^\pi &= \frac{1}{2} [Y_{i,j+1}^\pi + Y_{i+1,j+1}^\pi] + \Delta t \theta_1 f(t_j, Y_{i,j}^\pi, Z_{i,j}^\pi) + \\
&\quad \Delta t (1 - \theta_1) \frac{1}{2} [f(t_{j+1}, Y_{i,j+1}^\pi, Z_{i,j+1}^\pi) + f(t_{j+1}, Y_{i+1,j+1}^\pi, Z_{i+1,j+1}^\pi)],
\end{aligned}$$

for $i = 1, \dots, j$ and $j = 1, \dots, \mathcal{M}$ and where $Z_{i,j}$ and $Y_{i,j}$ denote the value of the processes Y_t and Z_t at node (i, j) in the tree, $\theta_1 \in [0, 1]$ and $\theta_2 \in (0, 1]$. Notice that we have an implicit expression for $Y_{i,j}^\pi$. Picard iterations are used to approximate the solution to this implicit equation. Starting with an initial estimate

$$(Y_{i,j}^\pi)_0 = \mathbb{E}_m[Y_{j+1}] = \frac{1}{2} (Y_{i,j+1}^\pi + Y_{i+1,j+1}^\pi),$$

we then calculate P Picard iterations as follows,

$$\begin{aligned}
(Y_{i,j}^\pi)_1 &= \frac{1}{2} [Y_{i,j+1}^\pi + Y_{i+1,j+1}^\pi] + \Delta t \theta_1 f(t_j, (Y_{i,j}^\pi)_0, Z_{i,j}^\pi) + \\
&\quad \Delta t (1 - \theta_1) \frac{1}{2} [f(t_{j+1}, Y_{i,j+1}^\pi, Z_{i,j+1}^\pi) + f(t_{j+1}, Y_{i+1,j+1}^\pi, Z_{i+1,j+1}^\pi)] \\
&\quad \vdots \\
(Y_{i,j}^\pi)_P &= \frac{1}{2} [Y_{i,j+1}^\pi + Y_{i+1,j+1}^\pi] + \Delta t \theta_1 f(t_j, (Y_{i,j}^\pi)_{P-1}, Z_{i,j}^\pi) + \\
&\quad \Delta t (1 - \theta_1) \frac{1}{2} [f(t_{j+1}, Y_{i,j+1}^\pi, Z_{i,j+1}^\pi) + f(t_{j+1}, Y_{i+1,j+1}^\pi, Z_{i+1,j+1}^\pi)],
\end{aligned}$$

and finally set $Y_{i,j}^\pi := (Y_{i,j}^\pi)_P$. With the scheme developed in this section, we are now able to numerically approximate solutions of FBSDEs. In the next section we will test the numerical method on a variety of FBSDEs, most problems from the field of mathematical finance, to analyze the convergence behaviour of the method.

5. NUMERICAL RESULTS AND CONVERGENCE ANALYSIS

In this section we will apply the numerical method developed in section 4 to several examples and analyze their convergence behaviour. The examples discussed in this section are mainly problems from mathematical finance (in particular we will look at contingent claim valuation problems) and some were already introduced in section 3. MATLAB R2010b is used for the implementation of the method. To analyze the behaviour of the theta-scheme we discuss the following schemes:

$$\begin{array}{ll} \text{Scheme A: } \theta_1 = 0, \theta_2 = 1 & \text{Scheme C: } \theta_1 = 1, \theta_2 = 1 \\ \text{Scheme B: } \theta_1 = \frac{1}{2}, \theta_2 = 1 & \text{Scheme D: } \theta_1 = \frac{1}{2}, \theta_2 = \frac{1}{2} \end{array}$$

Furthermore, we set the number of Picard iterations in the implicit schemes equal to $P = 5$.

5.1. Stochastic FitzHugh-Nagumo equation. Consider, first of all, the following nonlinear backward stochastic differential equation:

$$(5.1) \quad -dY_t = -Y_t(1 - Y_t)(\gamma - Y_t) - Z_t dW_t, \quad t \in [0, 1],$$

with parameter $\gamma \in (0, 1)$. In this equation the process Y_t represents the potential of a membrane. This equation is called the stochastic *FitzHugh-Nagumo equation* (see [RGG00]) and is used in genetics and biology, among other fields. We take

$$Y_1 = \frac{1}{1 + \exp\left\{-W_1 - \frac{1}{4}\right\}},$$

as the terminal condition and set $\gamma = \frac{3}{4}$. It is known that the exact solution of this BSDE is

$$(5.2) \quad (Y_t, Z_t) = \left(\frac{1}{1 + \exp\left\{-W_t - \frac{t}{4}\right\}}, \frac{\exp\left\{-W_t - \frac{t}{4}\right\}}{\left(1 + \exp\left\{-W_t - \frac{t}{4}\right\}\right)^2} \right),$$

and specifically one has:

$$(Y_0, Z_0) = \left(\frac{1}{2}, \frac{1}{4} \right).$$

From (5.2) one can see that the relationship $Y_t(1 - Y_t) = Z_t$ holds and that BSDE (5.1) is equal to the BSDE

$$-dY_t = -Z_t(\gamma - Y_t) - Z_t dW_t, \quad t \in [0, 1].$$

We can use this fact to test the sensitivity of our numerical method to dependencies of the driver f of the BSDE on the process Z_t . Notice that the driving forward process of this BSDE is just the Brownian motion W_t . The results of the binomial tree theta-scheme for different values of M can be found in figure 5.1.

We see that for all schemes the approximated values Y_0 and Z_0 converge with order $O(\Delta t)$. The higher order convergence properties of the discretization of scheme D does not seem to result in higher order convergence of the method in general. We notice however that the absolute error of schemes B and D is lower than the errors of the other schemes for both the Y and Z component of the BSDE.

Why doesn't scheme D result in higher order of convergence than the other three schemes? This is due to the fact that the order of convergence for the binomial tree approximation of the conditional expectations in our scheme is equal to one. This error dominates the error originating from the theta-discretization, resulting in a scheme that only has order of convergence equal to one.

The results of the numerical method for BSDE (5.1) with the driver depending on both the processes Y_t and Z_t can be found in figure 5.2. We observe a clear difference from figure 5.1. The dependency of the driver on the process Z_t still results in the same order of convergence $O(\Delta t)$ for all schemes, but this time scheme D underperforms relative to the other schemes for the Y component and scheme B and D both underperform for the Z component.

5.2. European call option in Black-Scholes market. In this next example we will compute the price of a European call option in a Black-Scholes market. The forward process in this case is a geometric Brownian motion

$$d\tilde{S}_t = \mu\tilde{S}_t dt + \sigma\tilde{S}_t dW_t.$$

The solution of this problem is the Black-Scholes price which can be computed from the Black-Scholes formula, see section 2. The BSDE for this problem was already derived in example 3.1 and is given by

$$(5.3) \quad -dY_t = \left(-rY_t - \frac{\mu - r}{\sigma} Z_t \right) dt - Z_t dW_t, \quad t \in [0, T],$$

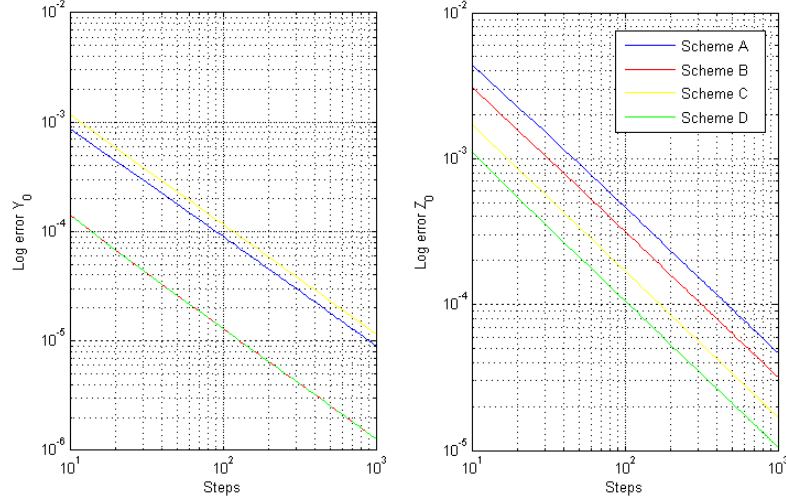


FIGURE 5.1. Results of example 5.1 with driver $f(y) = -y(1 - y)(\frac{3}{4} - y)$. Left: error Y_0 . Right: error Z_0 .

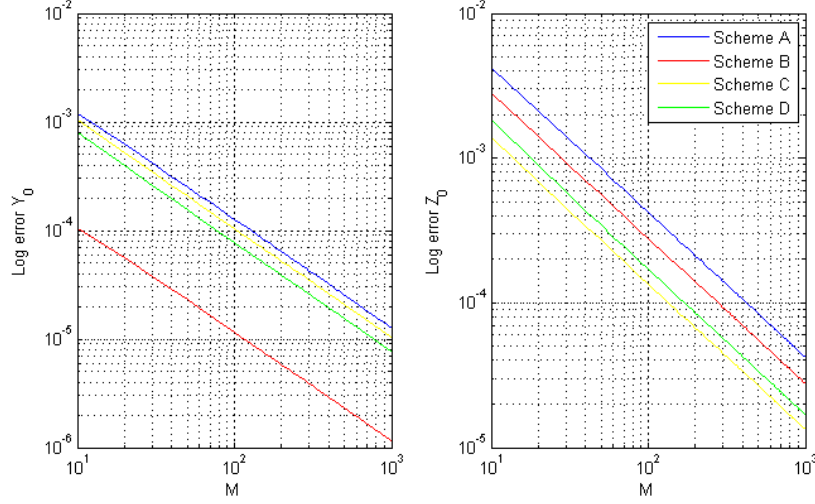


FIGURE 5.2. Results of example 5.1 with driver $f(y, z) = -z(\frac{3}{4} - y)$. Left: error Y_0 . Right: error Z_0 .

with terminal condition $Y_T = \max(\tilde{S}_T - \tilde{K}, 0)$. Here r is the interest rate in the market, K is the strike of the call option and T the maturity of the option. It was shown earlier that Y_t represents the value of the call option and that Z_t is related to the hedging strategy due to the relation $Z_t = \pi_t \sigma \tilde{S}_t$, where π_t represents the hedging portfolio at time t . For our tests, we use the following parameters:

$$\tilde{S}_0 = 100, r = 0.1, \mu = 0.2, \sigma = 0.25, \tilde{K} = 100, T = 0.1.$$

The exact solution of this problem is given by $(Y_0, Z_0) = (3.65997, 14.14823)$. Results of the numerical method can be found in figure 5.3. We see that for all schemes there is approximately $O(\Delta t)$ convergence in the Y_t component. However, scheme D behaves badly for the approximation of Z_0 .

A possible explanation for this phenomenon lies in our approximation of the forward process by a binomial tree. In the current problem the terminal condition for Y is not differentiable at the point $\tilde{S} = \tilde{K}$, which gives a discontinuous terminal condition (there is a jump at $\tilde{S} = \tilde{K}$) for the Z component. However, for our binomial approximation a certain smoothness of the solution is necessary. When $\theta_2 \neq 1$, the jump in the solution for the Z component is approximated by just two points and this causes problems in our numerical scheme. When $\theta_2 = 1$ we do not deal with this discontinuity.

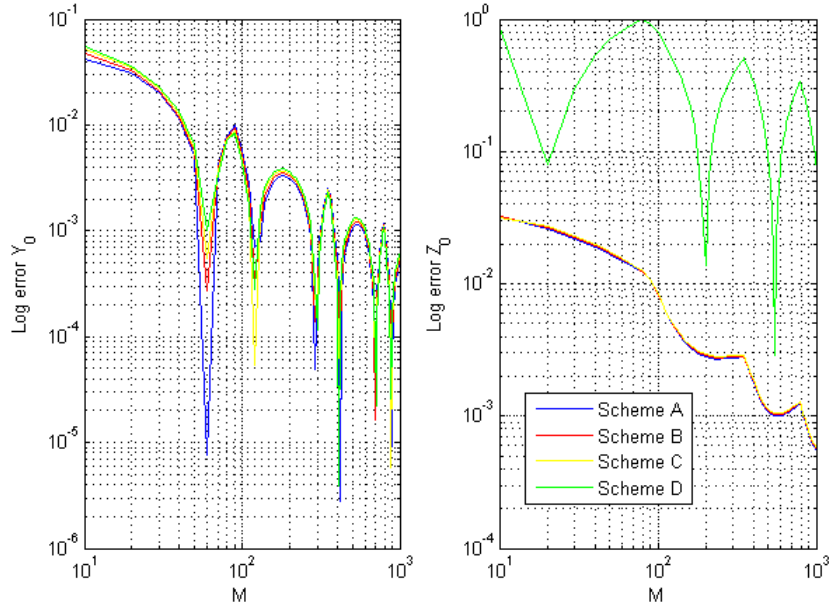


FIGURE 5.3. Results of example 5.2. Left: error Y_0 . Right: error Z_0 .

As an attempt to fix this problem, we set $\theta_2 = 1$ in time step $\mathcal{M} - 1$ and $\theta_2 = \frac{1}{2}$ for all other time steps in scheme D. In this case the jump in the solution for Z_t should not influence our approximation. The results of this adapted scheme can be found in figure 5.4. We see that the problem is only partially fixed: scheme D now exhibits much smoother convergence for the Z component, but still underperforms relative to the other schemes. Since in all other examples we observe the same ‘wild’ behaviour of the solution Z_0 when using scheme D, we will only use the adjusted version of scheme D (but still denote this scheme by ‘scheme D’).

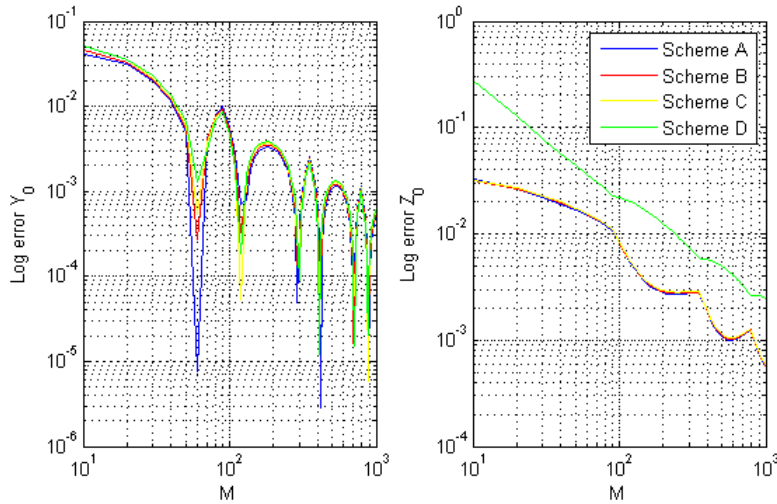


FIGURE 5.4. Results of example 5.2 with adapted scheme D. Left: error Y_0 . Right: error Z_0 .

5.3. European call spread in a market with an bid-ask spread for interest rates. The following model introduces a market with an bid-ask spread for interest rates and was discussed earlier in example 3.4. Let r denote the interest rate at which an investor can invest in a bank account and let $R > r$ denote the rate at which an investor borrows money. We furthermore assume these are constant and work with the same variables as in example 2. The BSDE that describes the value of an European option $g(\tilde{S}_T)$ in

this market is given by

$$(5.4) \quad -dY_t = -rY_t - \frac{\mu - r}{\sigma} Z_t - (R - r) \min \left(Y_t - \frac{Z_t}{\sigma}, 0 \right) dt - Z_t dW_t, \quad t \in [0, T],$$

with terminal condition $Y_T = g(\tilde{S}_T)$. In this market model the driver of the BSDE is nonlinear, but still Lipschitz, and no analytical solution exists. By theorem 3.2 there exists a corresponding PDE for the value function, from which the value of an option also can be computed.

We adapt an example from [BS12], which prices a position of one long call option with strike $\tilde{K}_1 = 95$ and two short call options with strike $\tilde{K}_2 = 105$. The payoff function is given by

$$g(\tilde{S}) = (\tilde{S} - \tilde{K}_1)^+ - 2(\tilde{S} - \tilde{K}_2)^+,$$

and they use parameters

$$\tilde{S}_0 = 100, r = 0.01, \mu = 0.05, \sigma = 0.2, T = 0.25.$$

We will perform tests with the interest rates $R = 0.06$ and $R = 3.01$. For the case $R = 0.06$ the solution is given by $(Y_0, Z_0) = (2.9584544, 0.55319)$ and for the case $R = 3.01$ the solution is given by $(Y_0, Z_0) = (6.3748, -4.690)$ (also see [RO13]). Of course, an interest rate of 301% is not realistic, but these parameters allow us to analyze the behaviour of the numerical method in more extreme circumstances. The results of our numerical scheme for $R = 0.06$ are shown in figure 5.5.

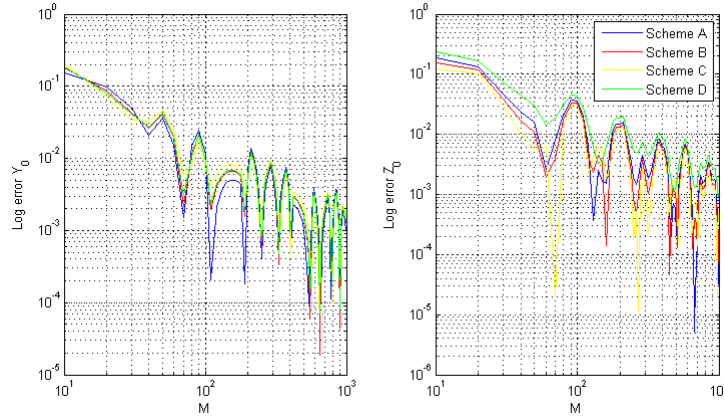


FIGURE 5.5. Results of example 5.3 with $R = 0.06$. Left: error Y_0 . Right: error Z_0 .

All schemes show $O(\Delta t)$ convergence, but none of the schemes is significantly better than the other schemes. Compared to the results in figure 5.4 the absolute error seems a bit higher, so the nonlinearity in the driver function f of the BSDE does seem to influence the numerical approximation.

In figure 5.6 the results of the scheme can be found for the case where $R = 3.01$. In this case the absolute error is clearly higher than the case where $R = 0.06$. This is caused due to the fact that the Lipschitz constant of the driver is equal to $L = \frac{R-r}{\sigma}$ and the fact that the error of the θ -discretization of the BSDE is dependent on this Lipschitz constant. In the case when $R = 3.01$ the Lipschitz constant is relatively big, causing a larger error.

5.4. Binary option in Black-Scholes model. As a final example, we consider the pricing of a binary option in the Black Scholes model. This option pays out a predefined notional if the price of the underlying is above a certain level $\tilde{K}$ (the strike) at maturity T . The BSDE considered in this example is the same as the one from example 2, but with a different terminal condition. The BSDE is given by

$$(5.5) \quad -dY_t = \left(-rY_t - \frac{\mu - r}{\sigma} Z_t \right) dt - Z_t dW_t, \quad t \in [0, T],$$

with terminal condition $Y_T = N1_{\{\tilde{S}_T > \tilde{K}\}}$, where N is some fixed amount of money called the *notional* of the option. We assume the notional is equal to one unit of money, the payoff for the option is thus equal to 1 unit of money when $\tilde{S}_T > \tilde{K}$ and zero otherwise. Since the terminal condition is still an

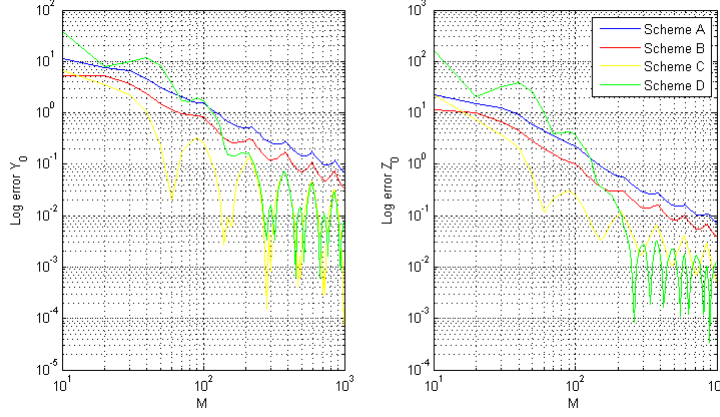


FIGURE 5.6. Results of example 5.3 with $R = 3.01$. Left: error Y_0 . Right: error Z_0 .

$\mathcal{F}_T$ -measurable random variable, its value can be determined by standard Black-Scholes analysis. We know from section 2 that the option price is equal to

$$\begin{aligned} Y_t &= e^{-r(T-t)} \mathbb{E}_{\mathbb{Q}} \left[1_{\{\tilde{S}_T > \tilde{K}\}} | \mathcal{F}_t \right] \\ &= e^{-r(T-t)} \mathbb{Q} \left[\tilde{S}_T > \tilde{K} | \mathcal{F}_t \right] \\ &= e^{-r(T-t)} \Phi(d_2), \end{aligned}$$

where $\mathbb{Q}$ is the usual risk-neutral measure and $d_2 = \frac{\log \frac{\tilde{S}_0}{\tilde{K}} + (r - \frac{\sigma^2}{2})(T-t)}{\sigma \sqrt{T-t}}$. The option delta is the derivative of the value function with respect to the stock price $\tilde{S}$ and equal to

$$\pi_t = e^{-r(T-t)} \frac{N'(d_2)}{\sigma \tilde{S}_t \sqrt{T-t}}.$$

From the relation $Z_0 = \sigma \tilde{S}_0 \pi_0$ we obtain the solution

$$(Y_0, Z_0) = \left(e^{-rT} N(d_2), e^{-rT} \frac{N'(d_2)}{\sqrt{T}} \right).$$

The binary option poses an interesting problem for our numerical method taking in consideration example 2. In this example we encountered a problem when the terminal condition for Z was discontinuous.

In the case of a binary option both terminal conditions for Y and Z are discontinuous. For Z the terminal condition is rather worse: it is equal to zero everywhere except for a spike to infinity at $\tilde{S} = \tilde{K}$. We will analyze how the numerical method will cope with these problems. The terminal condition for Z is approximated by the following function:

$$(5.6) \quad g(\tilde{S}) = c \times 1_{C_1 < \tilde{S} < C_2}$$

where $C_1 < C_2$ and $c \in \mathbb{R}$. In our scheme we then set $Z_{i,\mathcal{M}}^\pi = \sigma \tilde{S}_{\mathcal{M}} g(S_{\mathcal{M}})$ for all $i = 1, \dots, \mathcal{M}$.

For our tests we use the following configuration:

$$\tilde{S}_0 = 100, C_1 = 97, C_2 = 103, \tilde{K} = 100, r = 0.1, \mu = 0.2, \sigma = 0.25, T = 0.1$$

and we distinguish between the cases $c = 10$ and $c = 50$. For the former case, the results are given by figure 5.7 and for the latter case by figure 5.8.

The convergence pattern looks the same for both cases. However, we see that in the case when $c = 50$ the absolute error in the Z component is a larger than the case $c = 10$. The magnitude of the spike does seem to influence the numerical error.

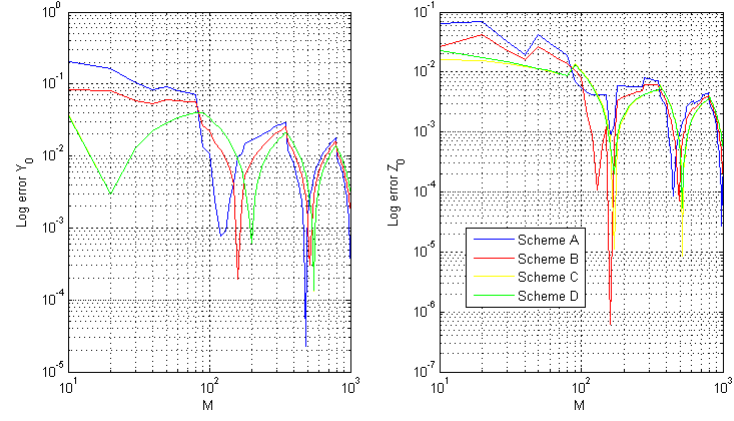


FIGURE 5.7. Results of example 5.4 with $c = 10$. Left: error Y_0 . Right: error Z_0 .

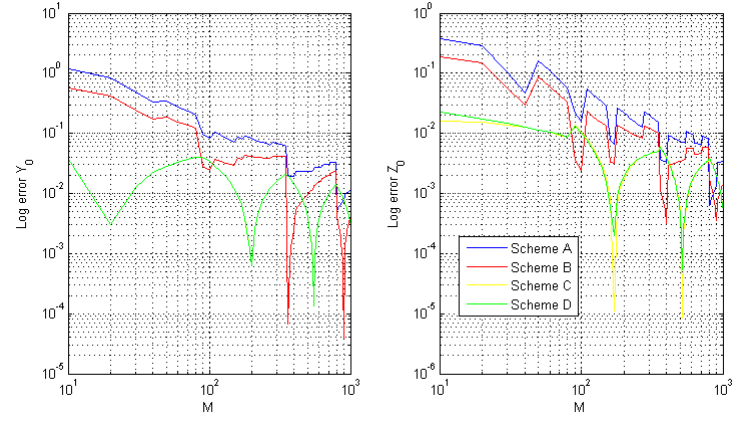


FIGURE 5.8. Results of example 5.4 with $c = 50$. Left: error Y_0 . Right: error Z_0 .

6. REFLECTED BSDEs

In this section we will introduce another type of BSDE, called a *reflected backward stochastic differential equation* (RBSDE). We discuss the usual assumptions underlying these types of BSDEs and present existence and uniqueness results and a link to a certain class of PDEs. An example of this type of BSDE is discussed and we extend our original numerical method to a method that solves RBSDEs.

6.1. Theorems and examples. We consider the following type of equation:

$$(6.1) \quad Y_t = \xi + \int_t^T f(s, Y_s, Z_s) ds + K_T - K_t - \int_t^T Z_s dW_s$$

$$(6.2) \quad Y_t \geq U_t$$

$$(6.3) \quad 0 = \int_0^T (Y_t - U_t) dK_t,$$

where ξ is the terminal condition and f is the driver of this *reflected BSDE*. We assume the pair (ξ, f) satisfies the usual assumptions from section 3. In addition we assume that

- $\xi \geq U_t$.
- $U_t \in \mathbb{S}^2(0, T)$.
- K is a continuous increasing process with $K_0 = 0$ and adapted to the filtration $\mathcal{F}_t$.

The process U_t is called the *obstacle* of the RBSDE and the process K_t serves to push the solution Y_t above the obstacle U_t . Condition (6.3) means that this is done in a minimal way: the process K_t is increased only when $Y_t = U_t$.

A solution to (6.1) is a triple $(Y_t, Z_t, K_t) \in \mathbb{S}^2(0, T) \times \mathbb{H}^2(0, T)^d \times \mathbb{S}^2(0, T)$. One can proof the following theorem:

Theorem 6.1. *Under the above mentioned assumptions, the RBSDE (6.1) - (6.3) has a unique solution.*

The proof for this statement can be found in [Pha09].

Just as with regular BSDEs there is a link between the solution of a RBSDE and a certain type of PDE. Let $\mathcal{L}$ denote the operator:

$$\mathcal{L}v = \bar{\mu}(x)D_x v(t, x) + \frac{1}{2} \text{Tr}(\bar{\sigma}(x)\bar{\sigma}(x)^T(x)D_x^2 v(t, x)).$$

Furthermore, assume we are given a diffusion:

$$dX_t = \bar{\mu}(X_t)dt + \bar{\sigma}(X_t)dW_t,$$

with $\bar{\mu}$ and $\bar{\sigma}$ Lipschitz continuous functions. Let $U_t = h(X_t)$ and $\xi = g(X_T)$ and assume the driver of the RBSDE may also be dependent on the value of the diffusion X_t . The solution to this problem is related to the PDE:

$$(6.4) \quad \min \left[-\frac{\partial v}{\partial t} - \mathcal{L}v - f(\cdot, v, \bar{\sigma}^T D_x v), v - h \right] = 0, \quad \text{on } [0, T) \times \mathbb{R}^d$$

$$v(T, \cdot) = g, \quad \text{on } \mathbb{R}^d,$$

by the following theorem:

Theorem 6.2. *Let v be a classical solution to (6.4) satisfying some growth conditions. Then the triple (Y, Z, K) defined by*

$$\begin{aligned} Y_t &= v(t, X_t) \\ Z_t &= \bar{\sigma}^T(X_t)D_x v(t, X_t) \\ K_t &= \int_0^t \left(-\frac{\partial v}{\partial t}(s, X_s) - \mathcal{L}v(s, X_s) - f(t, X_s, Y_s, Z_s) \right) ds, \end{aligned}$$

is the solution to the RBSDE (6.1).

Proof. Applying Itô's lemma as in the proof of theorem 3.2 we obtain that

$$\begin{aligned} dv(t, X_t) &= v_t(t, X_t)dt + \mathcal{L}v(t, X_t)dt + v_{xx}(t, X_t)\bar{\sigma}(X_t)dW_t \\ &= dK_t + f(t, X_t, Y_t, Z_t)dt + v_{xx}(t, X_t)\bar{\sigma}(X_t)dW_t \\ &= f(t, X_t, Y_t, Z_t)dt + Z_t dW_t + dK_t, \end{aligned}$$

the triple (Y, Z, K) thus satisfies the RBSDE. Because the function v satisfies equation (6.4) the integrand in the definition for K_t is nonnegative, hence K_t is nondecreasing. The minimality condition for K_t is satisfied in a straightforward way due to equality (6.4). From (6.4) it can also be seen that either $v - h = 0$, and thus $Y_t = h$, or that $v - h > 0$, in which case $Y_t > h$. So $Y_t \geq h$ and the proof is done. $\square$

Notice that this theorem is an exact analogue of theorem 3.2 for RBSDEs. PDEs of the form (6.4) are also called *variational inequalities*.

Example 6.1 (American call option). The value of an American call option at a time $t \in [0, T]$ satisfies a RBSDE. Assume $d = 1$ and we work in the general Black-Scholes market model. In this case the driver is given by:

$$f(t, x, y, z) = -ry - \lambda z,$$

where r is the interest rate and λ the risk premium on the market. If we set

$$h(\tilde{S}_t) = g(\tilde{S}_t) = \max(\tilde{S}_t - \tilde{K}, 0),$$

as the payoff of an American call option, the RBSDE obtained then describes the value of the option at an arbitrary time $t \in [0, T]$. The additional process K_t is needed for this problem because there exists no replicating strategy for the option. We have to use a super-replicating strategy with a consumption process K_t . The minimality condition on K_t just states that we only invest money in the portfolio when $V_t < h(\tilde{S}_t)$.

Using theorem 6.2 we also obtain the corresponding pricing PDE:

$$(6.5) \quad \min \left[\frac{\partial v}{\partial t} + \frac{1}{2} \sigma^2 \tilde{S}^2 \frac{\partial^2 v}{\partial \tilde{S}^2} + r \tilde{S} \frac{\partial v}{\partial \tilde{S}} - rv, v - h \right] = 0,$$

with $v(T, \cdot) = \max(\tilde{S}_T - K, 0)$. This is exactly what one obtains by risk-neutral considerations in the Black-Scholes model.

Example 6.2 (Bermudan options). Note that we can easily extend the framework to the pricing of Bermudan options. Assume we are given a set of exercise dates for the Bermudan option $\mathbb{T} = \{t_1, t_2, \dots, t_N = T\}$. If we set the obstacle U_t in the RBSDE as

$$\tilde{h}(\tilde{S}_t) = h(\tilde{S}_t) 1_{t \in \mathbb{T}},$$

the resulting equation then describes the price process of a Bermudan call option.

6.2. Numerical method for RBSDEs. In this section we will extend the original numerical method to a method that can approximate RBSDEs. Let $[0, T]$ be the interval of interest and let Π be a partition of time points $0 = t_0 < t_1 < \dots < t_M$ with fixed time step $\Delta t = t_{i+1} - t_i$. The notation from section 4 will be used here.

Suppose we have the following RBSDE and FSDE:

$$(6.6) \quad X_t = X_0 + \int_0^t \bar{\mu}(X_s) ds + \int_0^t \bar{\sigma}(X_s) dW_s$$

$$(6.7) \quad Y_t = g(X_T) + \int_t^T f(s, Y_s, Z_s) ds - \int_t^T Z_s dW_s + K_T - K_t$$

$$(6.8) \quad Y_t \geq h(X_t), \quad 0 \leq t \leq T.$$

with the usual definitions. Again we leave out the dependence of the function f on the forward process X_t , since in all of the examples we will discuss the driver function f does not depend on the forward process X_t . The forward SDE will be approximated by the usual Euler-Maruyama scheme introduced in section 4.

For the approximation of Y_t and Z_t we use the following technique: in each time step we first solve the unreflected BSDE

$$Y_{m-1} = Y_m + \int_{t_{m-1}}^{t_m} f(s, Y_s, Z_s) ds - \int_{t_{m-1}}^{t_m} Z_s dW_s,$$

and subsequently we check whether the solution does not go below the barrier. If it does, we set the value of the numerical approximation equal to the barrier and increase the value of the process K_t . Intuitively, the process K_t pushes the solution Y_t above the barrier. A similar idea is also used in [MZ05], where convergence of this approximation is proved.

To solve the unreflected BSDE on the interval $[t_m, t_{m+1}]$, we use the θ -discretization developed in section 4:

$$\begin{aligned} Y_{\mathcal{M}}^{\pi} &= g(X_{\mathcal{M}}^{\pi}), \quad Z_{\mathcal{M}}^{\pi} = \bar{\sigma}(X_{\mathcal{M}}^{\pi})g_x(X_{\mathcal{M}}^{\pi}) \\ Z_m^{\pi} &= -\theta_2^{-1}(1-\theta_2)\mathbb{E}_m[Z_{m+1}^{\pi}] + \frac{1}{\Delta t}\theta_2^{-1}\mathbb{E}_m[Y_{m+1}^{\pi}\Delta W_m] \\ &\quad + \theta_2^{-1}(1-\theta_2)\mathbb{E}_m[f(t_{m+1}, Y_{m+1}^{\pi}, Z_{m+1}^{\pi})\Delta W_m], \\ \tilde{Y}_m^{\pi} &= \mathbb{E}_m[Y_{m+1}^{\pi}] + \Delta t\theta_1 f(t_m, \tilde{Y}_m^{\pi}, Z_m^{\pi}) + \Delta t(1-\theta_1)\mathbb{E}_m[f(t_{m+1}, Y_{m+1}^{\pi}, Z_{m+1}^{\pi})], \end{aligned}$$

for $m = 1, \dots, \mathcal{M}$ and with the same definitions as before. The value for Y_m^{π} is computed by using Picard iterations, as before. We compare the numerical approximation with the barrier:

$$Y_m^{\pi} = \max \left\{ h(X_m^{\pi}), \tilde{Y}_m^{\pi} \right\}.$$

Note that in the above θ -scheme we use the known solution Y_{m+1}^{π} for all computations including an explicit term (the computed values for time t_{m+1}) when computing the numerical solution (Y_m^{π}, Z_m^{π}) . The process K_t is approximated by

$$K_m^{\pi} = \sum_{i=1}^m \left(Y_{i-1}^{\pi} - \tilde{Y}_{i-1}^{\pi} \right)$$

One can see from this approximation that:

$$\begin{aligned} K_m^{\pi} - K_{m-1}^{\pi} &= \sum_{i=1}^m \left(Y_{i-1}^{\pi} - \tilde{Y}_{i-1}^{\pi} \right) - \sum_{i=1}^{m-1} \left(Y_{i-1}^{\pi} - \tilde{Y}_{i-1}^{\pi} \right), \\ &= Y_{m-1}^{\pi} - \tilde{Y}_{m-1}^{\pi} \\ &= Y_{m-1}^{\pi} - Y_m^{\pi} - \int_{t_{m-1}}^{t_m} f(s, Y_s^{\pi}, Z_s^{\pi})ds + \int_{t_{m-1}}^{t_m} Z_s^{\pi}dW_s, \end{aligned}$$

because $\tilde{Y}_{m-1}^{\pi}$ is assumed to solve the unreflected BSDE on $[t_{m-1}, t_m]$. From this it directly follows that:

$$(6.9) \quad Y_{m-1}^{\pi} = Y_m^{\pi} + \int_{t_{m-1}}^{t_m} f(s, Y_s^{\pi}, Z_s^{\pi})ds - \int_{t_{m-1}}^{t_m} Z_s^{\pi}dW_s + K_m^{\pi} - K_{m-1}^{\pi}.$$

Since furthermore $Y_m^{\pi} > g(X_m^{\pi})$ for all $m = 1, \dots, \mathcal{M}$ this motivates our numerical approximation of the RBSDE (6.6), because equation (6.9) is the discrete version of the RSBDE.

For the approximation of the conditional expectations in this discretization we use the binomial tree approximation from section 4.3, since we first solve the unreflected BSDE on each time interval. Finally, we get the following numerical scheme:

$$\begin{aligned} Y_{i,\mathcal{M}}^{\pi} &= g(X_{i,\mathcal{M}}^{\pi}), \quad Z_{i,\mathcal{M}}^{\pi} = \bar{\sigma}(X_{i,\mathcal{M}}^{\pi})g_x(X_{i,\mathcal{M}}^{\pi}), \quad \forall i = 1, \dots, \mathcal{M}, \\ Z_{i,j}^{\pi} &= \frac{\theta_2 - 1}{2\theta_2} [Z_{i,j+1}^{\pi} + Z_{i+1,j+1}^{\pi}] + \frac{1}{2\theta_2\sqrt{\Delta t}} [Y_{i,j+1}^{\pi} - Y_{i+1,j+1}^{\pi}] + \\ &\quad \frac{1-\theta_2}{2\theta_2}\sqrt{\Delta t} [f(t_{j+1}, Y_{i,j+1}^{\pi}, Z_{i,j+1}^{\pi}) - f(t_{j+1}, Y_{i+1,j+1}^{\pi}, Z_{i+1,j+1}^{\pi})], \\ \tilde{Y}_{i,j}^{\pi} &= \frac{1}{2} [Y_{i,j+1}^{\pi} + Y_{i+1,j+1}^{\pi}] + \Delta t\theta_1 f(t_j, \tilde{Y}_{i,j}^{\pi}, Z_{i,j}^{\pi}) + \\ &\quad \Delta t(1-\theta_1) \frac{1}{2} [f(t_{j+1}, Y_{i,j+1}^{\pi}, Z_{i,j+1}^{\pi}) + f(t_{j+1}, Y_{i+1,j+1}^{\pi}, Z_{i+1,j+1}^{\pi})], \\ Y_{i,j}^{\pi} &= \max \left\{ h(X_{i,j}^{\pi}), \tilde{Y}_{i,j}^{\pi} \right\}, \\ K_{i,j+1}^{\pi} &= K_{i,j}^{\pi} + \left(Y_{i,j}^{\pi} - \tilde{Y}_{i,j}^{\pi} \right), \end{aligned}$$

for $i = 1, \dots, j$ and $j = 1, \dots, \mathcal{M}$, where $Z_{i,j}^{\pi}$, $Y_{i,j}^{\pi}$ and $K_{i,j}^{\pi}$ denote the values of the processes Y_t , Z_t and K_t at node (i, j) in the tree and where $\theta_1 \in [0, 1]$ and $\theta_2 \in (0, 1]$. Notice that, as before, we have an implicit expression for $\tilde{Y}_{i,j}^{\pi}$. Picard iterations are used to approximate the solution to this implicit equation. Starting with an initial estimate

$$(\tilde{Y}_{i,j}^{\pi})_0 = \mathbb{E}_m[Y_{j+1}] = \frac{1}{2} (Y_{i,j+1}^{\pi} + Y_{i+1,j+1}^{\pi}),$$

we then calculate P Picard iterations:

$$\begin{aligned}
(\tilde{Y}_{i,j}^\pi)_1 &= \frac{1}{2} [Y_{i,j+1}^\pi + Y_{i+1,j+1}^\pi] + \Delta t \theta_1 f(t_j, (\tilde{Y}_{i,j}^\pi)_0, Z_{i,j}^\pi) + \\
&\quad \Delta t (1 - \theta_1) \frac{1}{2} [f(t_{j+1}, Y_{i,j+1}^\pi, Z_{i,j+1}^\pi) + f(t_{j+1}, Y_{i+1,j+1}^\pi, Z_{i+1,j+1}^\pi)] \\
&\quad \vdots \\
(\tilde{Y}_{i,j}^\pi)_P &= \frac{1}{2} [Y_{i,j+1}^\pi + Y_{i+1,j+1}^\pi] + \Delta t \theta_1 f(t_j, (\tilde{Y}_{i,j}^\pi)_{P-1}, Z_{i,j}^\pi) + \\
&\quad \Delta t (1 - \theta_1) \frac{1}{2} [f(t_{j+1}, Y_{i,j+1}^\pi, Z_{i,j+1}^\pi) + f(t_{j+1}, Y_{i+1,j+1}^\pi, Z_{i+1,j+1}^\pi)] ,
\end{aligned}$$

and finally set $\tilde{Y}_{i,j}^\pi := (\tilde{Y}_{i,j}^\pi)_P$.

7. CONVERGENCE ANALYSIS OF RBSDE SCHEME

In this section we will apply the numerical method developed in the section 6 and analyze its convergence behaviour. Here we are mainly considered with the pricing of American options in different market models.

The θ -schemes used to test our method are

$$\begin{array}{ll} \text{Scheme A: } \theta_1 = 0, \theta_2 = 1 & \text{Scheme C: } \theta_1 = 1, \theta_2 = 1 \\ \text{Scheme B: } \theta_1 = \frac{1}{2}, \theta_2 = 1 & \text{Scheme D: } \theta_1 = \frac{1}{2}, \theta_2 = \frac{1}{2} \end{array}$$

and the number of Picard iterations we use is again set to $P = 5$.

7.1. American put option in Black-Scholes market. In this example we work in the standard Black-Scholes setting. We consider the problem of pricing an American put option, whose value satisfies the RBSDE

$$(7.1) \quad \begin{aligned} -dY_t &= \left(-r \cdot Y_t - \frac{\mu - r}{\sigma} Z_t \right) dt - Z_t dW_t + dK_t, \\ Y_T &= h(\tilde{S}_T), \\ Y_t &\geq g(\tilde{S}_t), \quad 0 \leq t \leq T, \end{aligned}$$

where $g(\tilde{S}_t) = h(\tilde{S}_t) = \max(\tilde{K} - \tilde{S}_t, 0)$, $\tilde{K}$ denotes the strike of the option, r the interest rate, μ the drift of the stock, σ the volatility and T denotes the maturity. As mentioned earlier, there is no analytical solution for the price of an American put option available. The parameters used are the following:

$$\tilde{S} = 40, \tilde{K} = 40, r = 0.0488, \sigma = 0.3, T = 0.5833.$$

In this case the value of the option is equal to 3.1696 and the delta of the option is equal to -0.4256, as reported by [HSY96]. Using the relationship between Z_t and the delta of the option we get that $Z_0 = -5.1072$. The results of our method can be found in figure 7.1. All schemes seem to have $O(\Delta t)$ convergence in the Y component, but again scheme D behaves very poorly in the Z component.

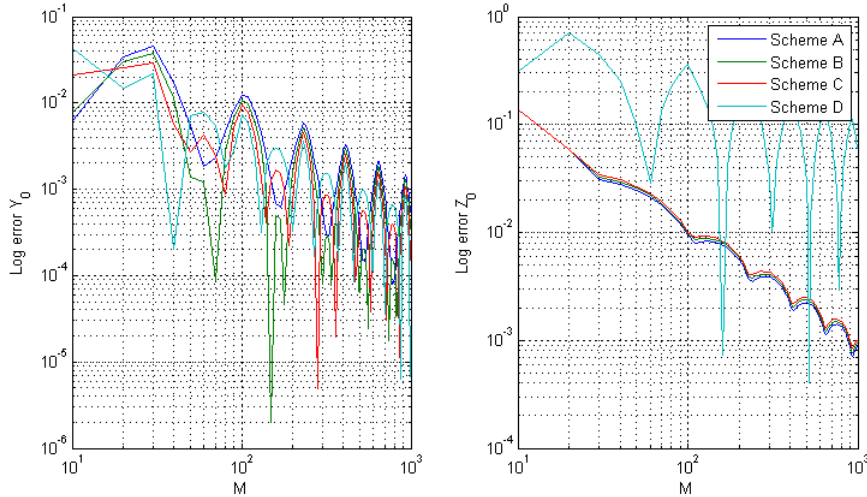


FIGURE 7.1. Results of example 7.1. Left: error Y_0 . Right: error Z_0 .

The explanation for this can again be found in the nature of our approximation. The binomial method does not seem to be able to cope well with discontinuities in the Z component. As an attempt to solve this problem, we apply the same fix as in section 5.2. The results of our adjusted method are then given by figure 7.2. In this case the behaviour of the error of Z_0 for scheme D is much smoother, but still larger than the error of the other schemes.

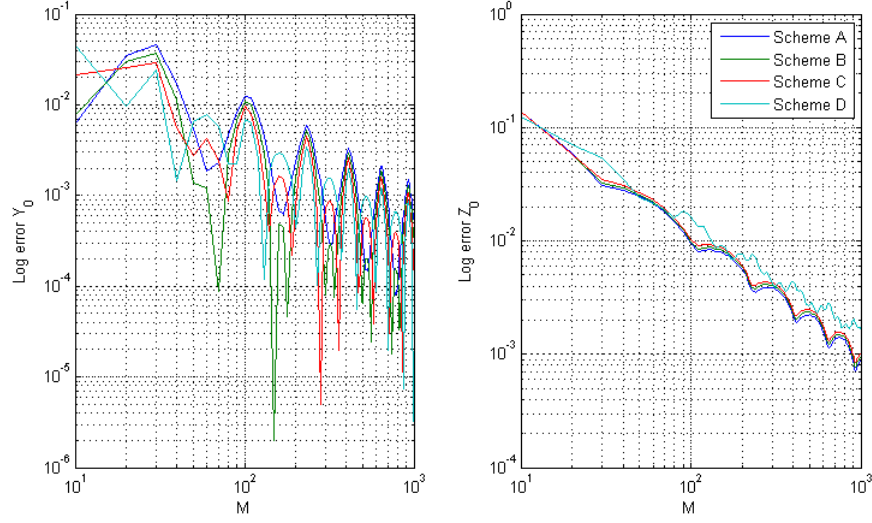


FIGURE 7.2. Results of example 7.1 with scheme D adjusted. Left: error Y_0 . Right: error Z_0 .

7.2. American call option in a market with a bid-ask spread for interest rates. We consider the valuation of an American call option in the presence of a bid-ask spread for interest rates. Combining the theory of examples 3.4 and 6.1 we get that the value of this option satisfies the following RBSDE

$$(7.2) \quad \begin{aligned} -dY_t &= -rY_t - \frac{\mu - r}{\sigma} Z_t - (R - r) \min \left(Y_t - \frac{Z_t}{\sigma}, 0 \right) dt - Z_t dW_t, \\ Y_T &= h(\tilde{S}_T), \\ Y_t &\geq g(\tilde{S}_t), \quad 0 \leq t \leq T, \end{aligned}$$

with the usual definitions for all symbols and where $g(\tilde{S}_t) = h(\tilde{S}_t) = \max(\tilde{S}_t - \tilde{K}, 0)$. To the knowledge of the author of this thesis, there is no known benchmark solution for this problem. The following settings are used:

$$\tilde{S}_0 = 100, \tilde{K} = 100, r = 0.01, R = 0.06, \mu = 0.05, \sigma = 0.2, T = 0.25.$$

Figure 7.3 shows the numerical solution against different time steps M . We observe convergence for both the components of the BSDE for all schemes, although the convergence for the Y component is a bit slow.

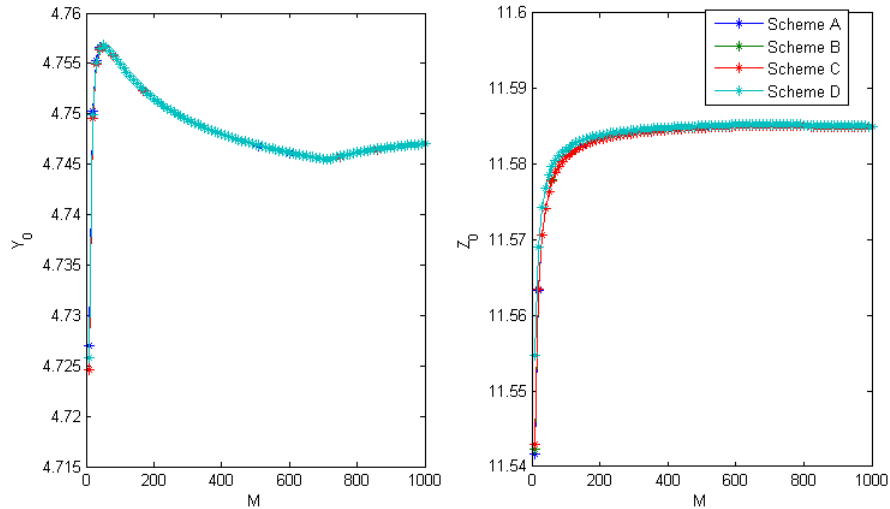


FIGURE 7.3. Results of example 7.2. Left: value of Y_0 . Right: value of Z_0 .

Taking $M = 5000$ we obtain a numerical solution of

$$(Y_0, Z_0) = (4.747036599903693, 11.585125024175634),$$

with scheme C. We use this approximation as a reference value, we can construct error plots as in all previous examples. Figure 7.4 shows the results.

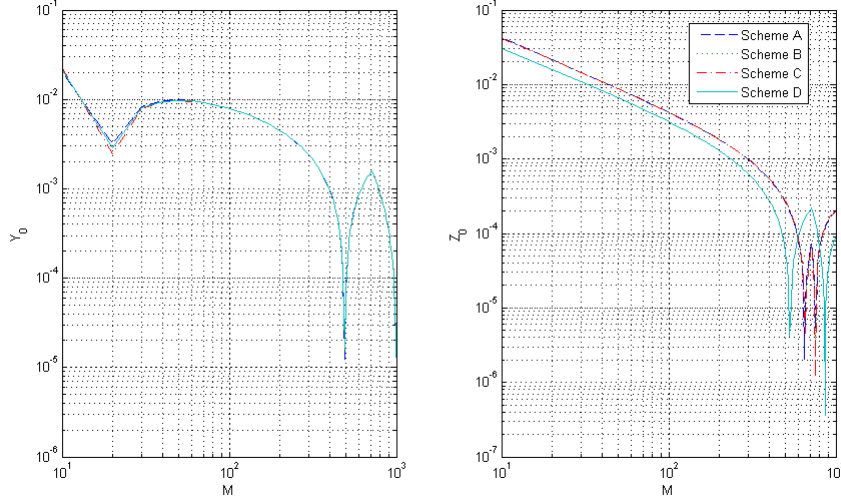


FIGURE 7.4. Error plot of example 7.2. Left: error of Y_0 . Right: error of Z_0 .

For all schemes we observe $O(\Delta t)$ convergence. Compared to the case of a European option, the magnitude of the numerical error seems smaller. However, this may be a consequence of using our own approximated solution obtained with $M = 5000$ as a benchmark value to compare all other results against.

8. FBSDEs WITH A $\mathbb{R}^2$ -VALUED FORWARD SDE

In this section we extend the numerical method to cope with backward stochastic differential equations where the terminal condition of the BSDE is allowed to dependent on the value of *two* forward SDEs. We are interested in these equations because they arise naturally when pricing options on two correlated assets. By extending the numerical method to solve these equations we can price a large set of European correlation products, for example max call options, basket options and spread options.

We discuss the extension of the method to the case where two forward SDEs are two correlated geometric Brownian motions. In more mathematical terms, we will look at the following equations

$$(8.1) \quad \begin{pmatrix} X_t^1 \\ X_t^2 \end{pmatrix} = \begin{pmatrix} X_0^1 \\ X_0^2 \end{pmatrix} + \int_0^t \begin{pmatrix} \mu_1 X_s^1 \\ \mu_2 X_s^2 \end{pmatrix} ds + \int_0^t \begin{pmatrix} \sigma_1 X_s^1 & 0 \\ \rho \sigma_2 X_s^2 & \sigma_2 \sqrt{1 - \rho^2} X_s^2 \end{pmatrix} d \begin{pmatrix} W_s^1 \\ W_s^2 \end{pmatrix}$$

$$(8.2) \quad Y_t = g(\mathbf{X}_T) + \int_t^T f(s, Y_s, \mathbf{Z}_s) ds - \int_t^T \mathbf{Z}_s d\mathbf{W}_s$$

In these equations, $g : \mathbb{R}^2 \rightarrow \mathbb{R}$ is the terminal condition and $\mathbf{W}_t = (W_t^1, W_t^2)$, $\mathbf{X}_t = (X_t^1, X_t^2)$ and $\mathbf{Z}_t = (Z_t^1, Z_t^2)$ are two-dimensional vectors. Furthermore, $\mu_1, \mu_2 \in \mathbb{R}$ describe the drift of the processes X_t^1 and X_t^2 , $\sigma_1, \sigma_2 \in \mathbb{R}_{>0}$ describe the volatility of X_t^1 and X_t^2 and ρ is the correlation between the processes X_t^1 and X_t^2 . The two Brownian motions W_t^1 and W_t^2 are assumed to be independent. The interpretation of these equations is that (8.1) describes the behaviour of two correlated stocks in a Black-Scholes market and equation (8.2) is the BSDE whose solution describes the price of a financial option on both assets.

Theorems 3.1 and 3.2 also apply for equations (8.1) - (8.2). Theorem 3.1 provides us existence of solutions to these equations. Furthermore, we also get terminal conditions for the processes $\mathbf{Z}_t$ and Y_t by theorem 3.2.

8.1. Extension of the numerical method. In this section we extend the numerical method to solve equations (8.1) - (8.2). As before, let $[0, T]$ be the interval of interest and let Π be a partition of time points $0 = t_0 < t_1 < \dots < t_M$ with fixed time step $\Delta t = t_{i+1} - t_i$. The notation from sections 4 and 6 is also used throughout this section. We start by discussing the approximation of the forward SDE (8.1).

We want to approximate the two-dimensional forward SDE by a binomial approximation. Since the forward SDE is now a two-dimensional SDE we will end up with a two-dimensional binomial tree, which is called a *binomial pyramid* (see [Rub94], although we will use a slightly different approach). It is essential from a computational point of view that this tree is recombining, since the amount of nodes in a non-recombining two-dimensional tree grows like 4^n .

The following methodology can easily be extended to a situation where the matrix in the Itô-integral in equation (8.1) is of the following form:

$$\begin{pmatrix} \bar{\sigma}_{11}(x) & \bar{\sigma}_{12}(x) \\ \bar{\sigma}_{21}(x) & \bar{\sigma}_{22}(x) \end{pmatrix},$$

where $\bar{\sigma}_{ij} : \mathbb{R} \rightarrow \mathbb{R}$ are linear functions for $i, j = 1, 2$. Otherwise it is not trivial to obtain a recombining tree.

To obtain a recombining tree for (8.1) we look at the discretized version of (8.1) and rewrite it as

$$(8.3) \quad \begin{aligned} \Delta \mathbf{X}_{m+1} &= \begin{pmatrix} \mu_1 X_m^1 \\ \mu_2 X_m^2 \end{pmatrix} \Delta t + \begin{pmatrix} \sigma_1 X_m^1 & 0 \\ \rho \sigma_2 X_m^2 & \sigma_2 \sqrt{1 - \rho^2} X_m^2 \end{pmatrix} \begin{pmatrix} \Delta W_m^1 \\ \Delta W_m^2 \end{pmatrix} \\ &= \begin{pmatrix} \mu_1 X_m^1 \\ \mu_2 X_m^2 \end{pmatrix} \Delta t + \begin{pmatrix} \sigma_1 X_m^1 \Delta \hat{W}_m^1 \\ \sigma_2 X_m^2 \Delta \hat{W}_m^2 \end{pmatrix}, \end{aligned}$$

where $\Delta \hat{W}_m^1$ and $\Delta \hat{W}_m^2$ are two Brownian motions connected to ΔW_m^1 and ΔW_m^2 by the relation

$$(8.4) \quad \begin{pmatrix} \Delta W_m^1 \\ \Delta W_m^2 \end{pmatrix} = \begin{pmatrix} \sigma_1 & 0 \\ \rho \sigma_2 & \sigma_2 \sqrt{1 - \rho^2} \end{pmatrix}^{-1} \begin{pmatrix} \sigma_1 \Delta \hat{W}_m^1 \\ \sigma_2 \Delta \hat{W}_m^2 \end{pmatrix}.$$

Equation (8.1) is now rewritten as a two-dimensional SDE, but such that the two Brownian motions $\hat{W}_t^1$ and $\hat{W}_t^2$ appearing in (8.3) are correlated with correlation parameter ρ . The Brownian motions $\hat{W}_m^1$ and

$\hat{W}_m^2$ are approximated by the scaled random walks

$$\begin{aligned}\hat{W}_m^1 &= \sqrt{\Delta t} \sum_{j=1}^m \epsilon_j = \hat{W}_{m-1}^1 + \sqrt{\Delta t} \epsilon_m \\ \hat{W}_m^2 &= \sqrt{\Delta t} \sum_{j=1}^m \eta_j = \hat{W}_{m-1}^2 + \sqrt{\Delta t} \eta_m,\end{aligned}$$

where η_j and ϵ_j are random variables with values in $\{+1, -1\}$. From this it follows that the pair $(\Delta \hat{W}_m^1, \Delta \hat{W}_m^2)$ can attain four different values:

$$(8.5) \quad (\Delta \hat{W}_m^1, \Delta \hat{W}_m^2) = \begin{cases} (\sqrt{\Delta t}, \sqrt{\Delta t}) & \text{with probability } p_{uu}, \\ (\sqrt{\Delta t}, -\sqrt{\Delta t}) & \text{with probability } p_{ud}, \\ (-\sqrt{\Delta t}, \sqrt{\Delta t}) & \text{with probability } p_{du}, \\ (-\sqrt{\Delta t}, -\sqrt{\Delta t}) & \text{with probability } p_{dd}. \end{cases}$$

The increments of both approximations should have expected value equal to zero, since a Brownian motion also satisfies this property. Furthermore, the Brownian motions $\hat{W}_t^1$ and $\hat{W}_t^2$ should be correlated with correlation parameter ρ . This means that the following equalities should be satisfied:

$$(8.6) \quad \begin{aligned} 0 &= \mathbb{E} [\Delta \hat{W}_m^1] &= p_{uu}\sqrt{\Delta t} + p_{ud}\sqrt{\Delta t} - p_{du}\sqrt{\Delta t} - p_{dd}\sqrt{\Delta t} \\ 0 &= \mathbb{E} [\Delta \hat{W}_m^2] &= p_{uu}\sqrt{\Delta t} - p_{ud}\sqrt{\Delta t} + p_{du}\sqrt{\Delta t} - p_{dd}\sqrt{\Delta t} \\ \rho &= \mathbb{E} [\Delta \hat{W}_m^1 \Delta \hat{W}_m^2] &= p_{uu}\Delta t - p_{ud}\Delta t - p_{du}\Delta t + p_{dd}\Delta t. \end{aligned}$$

A solution to these equations is

$$(8.7) \quad p_{uu} = p_{dd} = \frac{1}{4}(1 + \rho), \quad p_{ud} = p_{du} = \frac{1}{4}(1 - \rho).$$

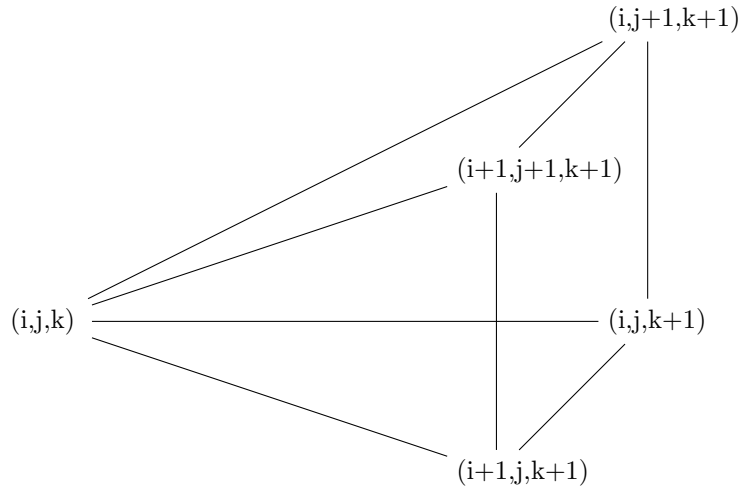
This provides us a scheme to approximate the two-dimensional forward SDE. We can approximate $\Delta \hat{W}_t^1$ and $\Delta \hat{W}_t^2$ by equations (8.5) and (8.7). Equation (8.3) then gives the approximation for X_t^1 and X_t^2 . This equation is equivalent to the two separate equations

$$(8.8) \quad \Delta X_{m+1}^1 = \mu_1 X_m^1 \Delta t + \sigma_1 X_m^1 \Delta \hat{W}_m^1$$

$$(8.9) \quad \Delta X_{m+1}^2 = \mu_2 X_m^2 \Delta t + \sigma_2 X_m^2 \Delta \hat{W}_m^2.$$

These are exactly the same type of equations we already encountered in section 4 and since the two Brownian motions are approximated by a random walk the numerical approximation of the components X^1 and X^2 has the structure of a recombining binomial tree. Relation (8.4) will be used later on, when we compute the conditional expectations in the theta-scheme for the BSDE.

The distribution of the pair $(\Delta \hat{W}_m^1, \Delta \hat{W}_m^2)$ is given by equation (8.5). The evolution of this pair over time has the structure of a two-dimensional binomial tree, a binomial pyramid. The following figure shows the time evolution of the binomial pyramid for one arbitrary node in the pyramid:



where we use the convention that k denotes the time step, i the movement in $\hat{W}_m^1$ and j the movement in $\hat{W}_m^2$ as follows:

- node $(i,j,k+1)$ represents the pair $(\Delta\hat{W}^1, \Delta\hat{W}^2) = \sqrt{\Delta t}(1, 1)$.
- node $(i,j+1,k+1)$ represents the pair $(\Delta\hat{W}^1, \Delta\hat{W}^2) = \sqrt{\Delta t}(1, -1)$.
- node $(i+1,j,k+1)$ represents the pair $(\Delta\hat{W}^1, \Delta\hat{W}^2) = \sqrt{\Delta t}(-1, 1)$.
- node $(i+1,j+1,k+1)$ represents the pair $(\Delta\hat{W}^1, \Delta\hat{W}^2) = \sqrt{\Delta t}(-1, -1)$.

This approximation enables the computation of $g(\mathbf{X}_T)$ and we now only have to approximate Y_t and $\mathbf{Z}_t$.

The numerical scheme developed in section 4 is still applicable, the derivation is not much different from the one-dimensional case. The terminal condition for Y_t is approximated by

$$Y_{i,j,\mathcal{M}} = g(X_{i,j,\mathcal{M}}^1, X_{i,j,\mathcal{M}}^2),$$

and the terminal condition for the process $\mathbf{Z}_t$ is again obtained by the Feynman-Kac analogue for BSDEs:

$$\mathbf{Z}_{i,j,\mathcal{M}} = \begin{pmatrix} \sigma_1 X_{i,j,\mathcal{M}}^1 & 0 \\ \rho\sigma_2 X_{i,j,\mathcal{M}}^2 & \sigma_2 \sqrt{1-\rho^2} X_{i,j,\mathcal{M}}^2 \end{pmatrix}^T \begin{pmatrix} g_{x_1}(X_{i,j,\mathcal{M}}^1, X_{i,j,\mathcal{M}}^2) \\ g_{x_2}(X_{i,j,\mathcal{M}}^1, X_{i,j,\mathcal{M}}^2) \end{pmatrix}.$$

The only problem left is the computation of the conditional expectations in the schemes. However, since we have approximated the processes X_t^1 and X_t^2 by a binomial process, these expectations can be easily computed as done before. The increments ΔW_m^1 and ΔW_m^2 are computed from the increments $\Delta\hat{W}^1$ and $\Delta\hat{W}^2$ by using relation (8.4). The resulting numerical scheme is the following:

$$\begin{aligned} Y_{i,j,\mathcal{M}}^\pi &= g(X_{i,j,\mathcal{M}}^1, X_{i,j,\mathcal{M}}^2), \quad \forall i, j = 1, \dots, \mathcal{M}, \\ \mathbf{Z}_{i,j,\mathcal{M}}^\pi &= \begin{pmatrix} \sigma_1 X_{i,j,\mathcal{M}}^1 & 0 \\ \rho\sigma_2 X_{i,j,\mathcal{M}}^2 & \sigma_2 \sqrt{1-\rho^2} X_{i,j,\mathcal{M}}^2 \end{pmatrix}^T \begin{pmatrix} g_{x_1}(X_{i,j,\mathcal{M}}^1, X_{i,j,\mathcal{M}}^2) \\ g_{x_2}(X_{i,j,\mathcal{M}}^1, X_{i,j,\mathcal{M}}^2) \end{pmatrix}, \quad \forall i, j = 1, \dots, \mathcal{M}, \\ \mathbf{Z}_{i,j,k}^\pi &= \frac{\theta_1 - 1}{\theta_2} [p_{uu}\mathbf{Z}_{i,j,k+1}^\pi + p_{ud}\mathbf{Z}_{i,j+1,k+1}^\pi + p_{du}\mathbf{Z}_{i+1,j,k+1}^\pi + p_{dd}\mathbf{Z}_{i+1,j+1,k+1}^\pi] + \\ &\quad \frac{1}{\Delta t \theta_2} [p_{uu}\mathbf{Z}_{i,j,k+1}^\pi \Delta \mathbf{W}_{i,j,k+1} + p_{ud}\mathbf{Z}_{i,j+1,k+1}^\pi \Delta \mathbf{W}_{i,j+1,k+1} + \\ &\quad p_{du}\mathbf{Z}_{i+1,j,k+1}^\pi \Delta \mathbf{W}_{i+1,j,k+1} + p_{dd}\mathbf{Z}_{i+1,j+1,k+1}^\pi \Delta \mathbf{W}_{i+1,j+1,k+1}] + \\ &\quad \frac{1 - \theta_2}{\theta_2} \sqrt{\Delta t} [p_{uu}f(t_{k+1}, Y_{i,j,k+1}^\pi, \mathbf{Z}_{i,j,k+1}^\pi) \Delta \mathbf{W}_{i,j,k+1} + p_{ud}f(t_{k+1}, Y_{i,j+1,k+1}^\pi, \mathbf{Z}_{i,j+1,k+1}^\pi) \Delta \mathbf{W}_{i,j+1,k+1} + \\ &\quad p_{du}f(t_{k+1}, Y_{i+1,j,k+1}^\pi, \mathbf{Z}_{i+1,j,k+1}^\pi) \Delta \mathbf{W}_{i+1,j,k+1} + p_{dd}f(t_{k+1}, Y_{i+1,j+1,k+1}^\pi, \mathbf{Z}_{i+1,j+1,k+1}^\pi) \Delta \mathbf{W}_{i+1,j+1,k+1}], \\ Y_{i,j,k}^\pi &= [p_{uu}Y_{i,j,k+1}^\pi + p_{ud}Y_{i,j+1,k+1}^\pi + p_{du}Y_{i+1,j,k+1}^\pi + p_{dd}Y_{i+1,j+1,k+1}^\pi] + \Delta t \theta_1 f(t_k, Y_{i,j,k}^\pi, \mathbf{Z}_{i,j,k}^\pi) + \\ &\quad \Delta t (1 - \theta_1) [p_{uu}f(t_{k+1}, Y_{i,j,k+1}^\pi, \mathbf{Z}_{i,j,k+1}^\pi) + p_{ud}f(t_{k+1}, Y_{i,j+1,k+1}^\pi, \mathbf{Z}_{i,j+1,k+1}^\pi) + \\ &\quad p_{du}f(t_{k+1}, Y_{i+1,j,k+1}^\pi, \mathbf{Z}_{i+1,j,k+1}^\pi) + p_{dd}f(t_{k+1}, Y_{i+1,j+1,k+1}^\pi, \mathbf{Z}_{i+1,j+1,k+1}^\pi)], \end{aligned}$$

for $i, j = 1, \dots, k$ and $k = 1, \dots, \mathcal{M}$, where $X_{i,j,k}^1$, $X_{i,j,k}^2$, $Y_{i,j,k}^\pi$ and $\mathbf{Z}_{i,j,k}^\pi$ denote the values of the approximations of the processes X_t^1 , X_t^2 , Y_t and $\mathbf{Z}_t$ at node (i, j, k) in the pyramid. The value $Y_{i,j,k}^\pi$ in the last equation is computed by Picard iterations in the same way as in sections 4 and 6, so convergence of these iterations is guaranteed.

The above scheme can easily be extended to the case where equation (8.2) is not a regular BSDE, but a reflected BSDE. In this case an additional equation has to be added to equations (8.1)-(8.2), which is the barrier equation for the RBSDE. The scheme developed in this section can then easily be extended to support these equations as well, by the same procedure discussed in section 6.

9. CONVERGENCE ANALYSIS OF THE TWO-DIMENSIONAL SCHEME

In this section we will apply the numerical method developed in section 8 and analyze its convergence behaviour. Due to time constraints we were able to test the numerical method on only one example. We will only consider the pricing of a European spread option on two correlated assets. However, the method is capable of solving a more general class of European correlation products as well by adjusting the terminal conditions for the BSDE. The θ -schemes used to test the method are

$$\begin{array}{ll} \text{Scheme A: } \theta_1 = 0, \theta_2 = 1 & \text{Scheme C: } \theta_1 = 1, \theta_2 = 1 \\ \text{Scheme B: } \theta_1 = \frac{1}{2}, \theta_2 = 1 & \text{Scheme D: } \theta_1 = \frac{1}{2}, \theta_2 = \frac{1}{2} \end{array}$$

and the number of Picard iterations we use is set to $P = 5$. In sections 5 and 7 we observed that scheme D behaved very poorly for the Z_t -component. Therefore, we already apply the fix proposed for scheme D in this section: at time $t_{\mathcal{M}-1}$ we let $\theta_1 = \theta_2 = 1$ and choose $\theta_1 = \theta_2 = \frac{1}{2}$ otherwise, where $\mathcal{M}$ denotes the number of time steps.

9.1. European spread option in a Black-Scholes market. In this example we work in the setting already introduced in section 8. We consider two assets that are correlated with correlation parameter ρ . Their behaviour is modelled by the forward SDE

$$\begin{pmatrix} \tilde{S}_t^1 \\ \tilde{S}_t^2 \end{pmatrix} = \begin{pmatrix} \tilde{S}_0^1 \\ \tilde{S}_0^2 \end{pmatrix} + \int_0^t \begin{pmatrix} \mu_1 \tilde{S}_s^1 \\ \mu_2 \tilde{S}_s^2 \end{pmatrix} ds \\ + \int_0^t \begin{pmatrix} \sigma_1 \tilde{S}_s^1 & 0 \\ \rho \sigma_2 \tilde{S}_s^2 & \sigma_2 \sqrt{1-\rho^2} \tilde{S}_s^2 \end{pmatrix} d \begin{pmatrix} W_t^1 \\ W_t^2 \end{pmatrix},$$

where $\tilde{S}_t^1$ and $\tilde{S}_t^2$ describe the stock prices of both stocks, $\mu_1, \mu_2 \in \mathbb{R}$ are the drifts of both stocks, $\sigma_1, \sigma_2 \in \mathbb{R}_{>0}$ denote the volatilities of both stocks, W_t^1 and W_t^2 are two independent Brownian motions and ρ is the correlation between both stocks. The BSDE corresponding to this valuation problem is

$$Y_t = g(\mathbf{S}_T) + \int_t^T f(s, Y_s, \mathbf{Z}_s) ds - \int_t^T \mathbf{Z}_s d\mathbf{W}_s.$$

with driver function $f(t, Y_t, \mathbf{Z}_t) = -rY_t - \left(\begin{pmatrix} \sigma_1 & 0 \\ \rho \sigma_2 & \sigma_2 \sqrt{1-\rho^2} \end{pmatrix}^{-1} (\mu - r) \mathbf{Z}_t \right)$, where r is the interest rate and $\mu = (\mu_1, \mu_2)$ the drift vector. The terminal condition of this BSDE is the option payoff of a spread option. A spread option on two stocks $\tilde{S}^1$ and $\tilde{S}^2$ pays out $\tilde{S}^1 - \tilde{S}^2$ if $\tilde{S}^1 > \tilde{S}^2$ at maturity, and zero otherwise. In mathematical terms, the terminal condition is equal to

$$g \left(\begin{pmatrix} \tilde{S}_T^1 \\ \tilde{S}_T^2 \end{pmatrix} \right) = \max(\tilde{S}_T^1 - \tilde{S}_T^2, 0).$$

The analytical solution to this problem is given by Margrabe's formula [Mar78]. For our tests, we use the parameters:

$$\tilde{S}_0^1 = \tilde{S}_0^2 = 100, r = 0.05, \mu_1 = \mu_2 = 0.1, \sigma_1 = 0.25, \sigma_2 = 0.3, \rho = 0.3, T = 0.1.$$

The solution to this problem is then equal to

$$(Y_0, Z_0^1, Z_0^2) = (4.1345, 8.7029, -13.715).$$

The results of the numerical method for different values of M are given in figure 9.1. For all components we observe order $O(\Delta t)$ convergence. There is no scheme that is particularly better than other schemes. For the component Z^1 scheme D performs worse than the other schemes but for the component Z^2 it is the other way around. To obtain a better analysis of the behaviour of our method, one could investigate what happens for different values of ρ . One could also vary the drifts of both stocks, to see how this algorithm performs when the Lipschitz constant of the driver function f is varied.

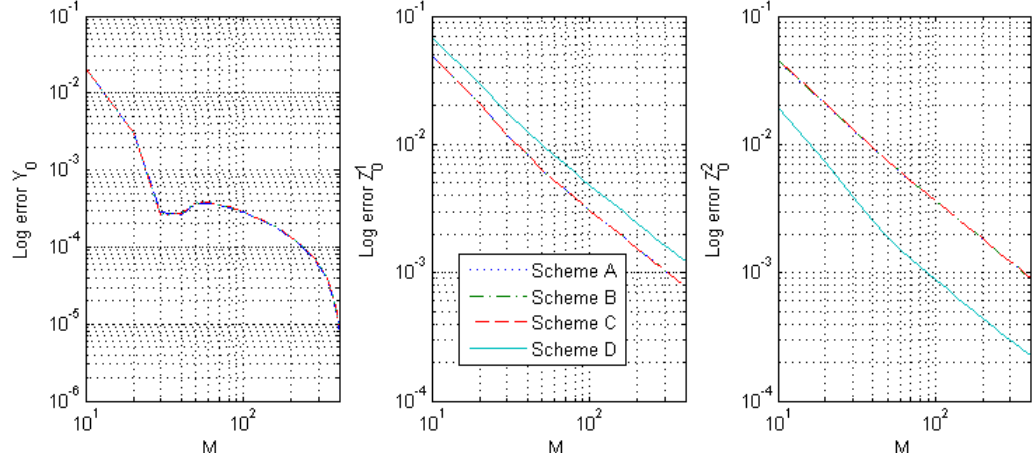


FIGURE 9.1. Results of of the numerical method. Left: Error Y_0 . Middle: Error Z_0^1 . Right: Error Z_0^2 .

10. CONCLUSIONS AND FURTHER RESEARCH

In this section we will discuss the main conclusions of the research conducted on the numerical methods developed in thesis. We will also provide some topics for further research.

10.1. Conclusions. In this section we will provide conclusions for the numerical method for FBSDEs as well as for the method of reflected FBSDEs and the numerical method for FBSDEs with a two-dimensional forward SDE, since the conclusions for both methods are quite similar. From the examples discussed in sections 5 and 6.2 we can form the following conclusions:

- For all schemes and examples we observed $O(\Delta t)$ convergence. However, the numerical scheme with $\theta_1 = \theta_2 = \frac{1}{2}$ does not result in order of convergence equal to two. Although theoretically the numerical discretization of the (R)BSDE should be of order two of this scheme, the error of the binomial approximation of the underlying Brownian motion (which is of order one) dominates the final numerical error.
- The magnitude of the error for different choices of θ_1 and θ_2 is dependent on the dependency of the driver function f on the process Z_t . Especially in the case where $\theta_1 = \theta_2 = \frac{1}{2}$, the numerical error grows when the driver f is dependent of the process Z_t . However, this dependency does not influence the order of convergence.
- Also the nonlinearity of the driver f also influences the numerical error. In the linear case, we get a much smoother error curve, but also a much smaller error than in the nonlinear case. In the nonlinear case the numerical error is bigger and the error curve oscillates more as well. This last point is not a problem however, the order of convergence is still the same for both cases.
- The magnitude of the Lipschitz constant of the driver function f is also important. In [ZWP09] it is shown, among others, that the θ -discretization discussed in this thesis converges and the bound for the error one obtains includes the Lipschitz constant. The larger this constant, the larger the magnitude of the numerical error.
- Finally, the behaviour of the terminal condition is also of great importance. A discontinuous terminal condition amplifies the numerical error, we also saw that the error gets even bigger when the terminal condition is non-differentiable as well.

The numerical method works well when the driver function f is linear and when the terminal conditions are differentiable. When trying to numerically approximate the solution of such a problem, our method would work well. The binomial tree method is intuitive and easy to grasp and the implementation of this numerical method is fairly easy. Even when the problem we are facing is nonlinear in nature, or has discontinuous terminal conditions, the numerical method exhibits $O(\Delta t)$ convergence and works well.

However, when dealing with problems that do not have these features another numerical method may be more favorable. In all the discussed examples with such features the error of our method was larger and if one wants to have an accuracy in the order of 10^{-6} at least 10^6 time steps are needed, which asks a significant amount of computation time and computational memory.

Another important topic is the fact that our numerical method is only ‘nice’ in computational terms if the underlying binomial tree is *recombining*, which it is in our examples. In this case an upward movement following a downward movement in the tree is the same as a downward movement followed by an upward movement. If the underlying tree is not recombining, the amount of nodes at time M is equal to 2^M . For large M , we get computational problems pretty easily, for $M = 30$ for example, there are already $1.0737 \cdot 10^9$ nodes in the tree.

There are methods in the literature with higher order of convergence ([RO13], for example) which may be suited better for this task. The problems encountered in practice are more nonlinear or wilder in nature than the problems discussed in this thesis, so further research to find a better numerical method for (R)BSDEs is still needed. However, our method converges and is still reliable, even when facing nonlinear problems. As noted before, the binomial tree approximation is intuitive, easy to grasp and easily implemented. For practitioners, this is an advantage of this method.

10.2. Further research. There are a few topics that could be researched to improve the current binomial tree algorithm. For example, the following topics could be subjects for further research:

- Instead of a binomial approximation to the underlying Brownian motion, one could investigate the *trinomial tree* approximation to Brownian motion. This approximation has higher order

of convergence than the regular binomial approximation and this may increase the order of convergence of the numerical method in general as well.

- Another research topic could be the extension of the numerical method to be able to cope with jump processes. Since the underlying binomial tree approximation of a jump process does not necessarily have to be recombining, this is not trivial.
- One could try to extend the numerical method to (reflected) FBSDEs where the coefficients of the forward SDE are nonlinear. In this case the underlying binomial tree also is not recombining anymore. Such a numerical method would allow us to look at problems from local volatility models in mathematical finance, for instance.
- Finally, one could extend the numerical method to cope with more general systems of a multi-dimensional forward stochastic differential equation and a one-dimensional backward stochastic equation. Such a method would be capable of pricing, for example, a variety of American correlation products on more than one financial asset.

REFERENCES

- [BS12] C. Bender, J. Steiner. Least-squares Monte Carlo for backward SDEs. *Numerical Methods in Finance*, 12:257-289 2012.
- [BT04] B. Bouchard and N. Touzi. Discrete-time approximation and Monte-Carlo Simulation of backward stochastic differential equations. *Stochastic Processes and their applications*, 111(2):175-206, 2002.
- [EKPQ97] N. El Karoui, S. Peng and M. C. Quenez. Backward stochastic differential equations in finance. *Mathematical finance*, 7(1):1-71, 1997.
- [Eyr05] A. Eyraud-Loisel. Backward stochastic differential equations with enlarged filtration, option hedging of an insider trader in a financial market with jumps. *Stochastic Processes And Their Applications*, 115:1745-1763, 2005.
- [FTW11] A. Fahim, N. Touzi, and X. Warin. A probabilistic numerical method for fully nonlinear parabolic PDEs. *The Annals of Applied Probability*, 21(4):1322-1364, 2011.
- [Gob10] E. Gobet. Numerics of Backward SDEs. Presentation slides, 2010. Summer School in Probability Theory - Disentis - 26-30 July 2010.
- [HSY96] J. Huang, M. Subrahmanyam, G. Yu. Pricing and Hedging American Options: A Recursive Integration Method. *The Review of Financial Studies*, 9(1):277-300, 1996.
- [Ise09] A. Iserles. *First course in numerical solutions of differential equations*, Cambridge Texts in Applied Mathematics. Cambridge University Press, 2009.
- [JP04] J. Jacod, P. Protter. *Probability Essentials*, 2nd Edition. Springer, 2004.
- [KPS94] P. Kloeden, E. Platen and H. Schurz. *Numerical solution of stochastic differential equations through computer experiments*, Springer, 1994.
- [LL11] C. Labart, J. Lelong. A Parallel Algorithm for solving BSDEs-Application to the pricing and hedging of American options. *arXiv preprint arXiv:1102.4666*, 2011.
- [Mar78] W. Margrabe. The Value of an Option to Exchange One Asset for Another. *Journal of Finance*, 33(1):177-186, 1978.
- [MPSMT02] J. Ma, P. Protter, J. San Martin and S. Torres. Numerical Method for Backward SDE's. *The Annals of Applied Probability*, 12(1):302-316, 2002.
- [MZ05] J. Ma, J. Zhang. Representations and regularities for solutions to BSDEs with reflections. *Stochastic Processes and their Applications*, 115(4):539-569, 2005.
- [Pen10] Y. Peng, et al. Parallel Computing for Option Pricing Based on the Backward Stochastic Differential Equation. *High Performance Computing and Applications*, 325-330 2010.
- [Pha09] H. Pham. *Continuous-time stochastic control and optimization with financial applications*. Springer, 2009.
- [Pro04] P. Protter. *Stochastic Integration and Differential Equations*, 2nd Edition. Springer, 2004.
- [RGG00] C. Roc soreanu, A. Georgescu, N. Giurgiteanu. The FitzHugh-Nagumo model, *Mathematical Modelling: Theory and Applications*, Volume 10, Kluwer Academic Publishers, 2000
- [RO13] M. Ruijter, C. Oosterlee. A Fourier-cosine method for an efficient computation of solutions to BSDEs, 2013.
- [Rub94] M. Rubinstein. Return to Oz, *Risk*, 11:67-71, 1994.
- [Sch12] M. Schweizer. Lecture notes for Mathematical Foundations for Finance, 2012.
- [SS08] M. Schroder, C. Skiadas. Optimality and state pricing in constrained financial markets with recursive utility under continuous and discontinuous information. *Mathematical finance*, 18(2):199-238, 2008.
- [Zha04] J. Zhang. A numerical scheme for BSDEs. *The Annals of Applied Probability*, 14(1):459-488, 2004.
- [ZWP09] W. Zhao, J. Wang and S. Peng. Error-estimates of the theta-scheme for backward stochastic differential equations. *Discrete and Continuous Dynamical Systems - Series B*, 12(4):905-924, 2009.

APPENDIX

- Euler-Maruyama scheme and computation of strong error.

```

randn('state',100)

sigma = 1;
mu = 1;
Xzero = 1.5;
T = 1;
N = 2^9;
dt = T/N;
M = 1000;
Xerr = zeros(M,9);

for s = 1:M

    dW = sqrt(dt)*randn(1,N);
    W = cumsum(dW);
    Xtrue = Xzero*exp((sigma-0.5*mu^2)+mu*W(end));

    for p = 1:9

        R = 2^(p-1);
        Dt = R*dt;
        L = N/R; %
        Xtemp = Xzero;

        for j = 1:L
            Winc = sum(dW(R*(j-1)+1:R*j));
            Xtemp = Xtemp + Dt*sigma*Xtemp + mu*Xtemp*Winc;
        end

        Xerr(s,p) = abs(Xtemp - Xtrue);
    end
end
Dtvals = dt*(2.^([0:8]));
loglog(Dtvals,mean(Xerr),'r*-'); hold on

```

- Euler-Maruyama scheme and computation of weak error.

```

%randn('state',100);
sigma = 1; mu = 1; Xzero = 1.5;T=1;

M = 100000; % number of paths sampled
Xem = zeros(9,1);

for p = 1:9
    Dt = 2^(p-10);
    L = T/Dt;
    Xtemp = Xzero*ones(M,1);

    for j = 1:L
        Winc = sqrt(Dt)*randn(M,1);
        Xtemp = Xtemp + Dt*sigma*Xtemp + mu*Xtemp.*Winc;
    end

    Xem(p) = mean(Xtemp);
end

Xerr2 = abs(Xem - 1.5*exp(sigma));
Dtvals2 = 2.^([1:5]-10);
loglog(Dtvals,Xerr2,'b*-'); hold on;

```

- Binomial approximation of GBM and computation of weak error.

```
function [Xweak] = binomEM(T,n)
    dt = T/n;
    mu = 0.4;
    sigma = 0.1;
    X0 = 1;
    X = zeros(n+1,n+1);
    X(1,1) = X0;

    % Compute binomial tree approximation of GBM

    for j = 2:n+1
        X(1:j-1,j) = X(1:j-1,j-1) + mu*X(1:j-1,j-1)*dt - sigma*X(1:j-1,j-1)*sqrt(dt);
        X(j,j) = X(j-1,j-1) + mu*X(j-1,j-1)*dt + sigma*X(j-1,j-1)*sqrt(dt);
    end

    % Compute expected value of X_T
    val=pascal_triangle(n);
    Xweak = sum(val(end,:)'.*(0.5^n).*X(:,end));
end

clear all;
hold off;

% Euler-Maruyama approximations of geometric brownian motion

T = 1; % time interval length [0,T]
h = 10:10:200;
end_vals1 = zeros(length(h),1);
c = 1;

for j = h
    [X] = binomEM(T,j);
    end_vals1(c,1) = X;
    c = c+1;
end

weak_error = abs(end_vals1 - exp(0.4));
loglog(h,weak_error,'-b');
legend('Weak error');
xlabel('M');
ylabel('Weak error');
```

- θ -algorithm for FBSDEs.

```
function [X,Y,Z] = binomBSDE(T,n,P,X0,mu,sigma,theta1,theta2,f,g,g_x)
    % Initialize variables
    dt = T/n;
    X = zeros(n+1,n+1);
    X(1,1) = X0;
    Y = zeros(n+1,n+1);
    Z = zeros(n+1,n+1);

    % Compute the binomial tree approximation of the forward SDE
    for j = 2:n+1
        X(1:j-1,j) = X(1:j-1,j-1) + mu(X(1:j-1,j-1))*dt - sigma(X(1:j-1,j-1))*sqrt(dt);
        X(j,j) = X(j-1,j-1) + mu(X(j-1,j-1))*dt + sigma(X(j-1,j-1))*sqrt(dt);
    end

    % Terminal conditions for the BSDE
```

```

Y(:,end) = g(X(:,end));
Z(:,end) = sigma(X(:,end)).*g_x(X(:,end));

% Compute numerical solution (Y_t,Z_t)

p = 0.5;
t2 = theta2; % this is for the 'fix'
for j = n:-1:1;
    if theta2 == 1/2 && j==n
        theta2 = 1;
    end

    Z(1:j,j) = (theta2 - 1)/(theta2)*((1-p)*Z(1:j,j+1) + p*Z(2:(j+1),j+1)) ...
        + 1/(theta2*sqrt(dt)) * (p*Y(2:(j+1),j+1) - (1-p)*Y(1:j,j+1)) + ...
        (1-theta2)/(theta2) *sqrt(dt)* ...
        (p*f(j+1,Y(2:(j+1),j+1),Z(2:(j+1),j+1)) - ...
        (1-p)*f(j+1,Y(1:j,j+1),Z(1:j,j+1)));
    picard = zeros(j,P);
    picard(:,1) = (1-p)*Y(1:j,j+1) + p*Y(2:(j+1),j+1);
    for k = 2:P
        picard(:,k) = (1-p)*Y(1:j,j+1) + p*Y(2:(j+1),j+1) + dt*theta1*f(j, ...
            picard(:,k-1), Z(1:j,j)) + ...
            dt*(1-theta1)*((1-p)*f(j+1,Y(1:j,j+1),Z(1:j,j+1)) + ...
            p*f(j+1,Y(2:(j+1),j+1), Z(2:(j+1),j+1)));
    end
    Y(1:j,j) = picard(:,P);
    theta2 = t2;
end
end

```

- Main file that calls the binomBSDE-function and computes the error.

```

clear all;
close all;
% Initialize variables

steps = 10:10:200;
theta = [0, 1;
        1/2, 1;
        1, 1;
        1/2, 1/2];
n_schemes = size(theta,1);
amount_steps = length(steps);

% Initialize variables that will hold Y_0 and Z_0 for each scheme
Y = zeros(amount_steps,n_schemes);
Z = zeros(amount_steps,n_schemes);
% Counter for the loop;
c = 1;
% Configuration of the problem
drift = 0.2;
diffusion = 0.25;
T = 0.1; % time interval [0,T]
X0 = 100;

% Define mu and sigma for the forward SDE (functions of X_t)
mu = @(x) drift*x;
sigma = @(x) diffusion*x;

% Compute the numerical solution for each scheme
for j = 1:n_schemes
    c = 1;
    for i = steps;

```

```

[X,A,B] = binomBSDE(T,i,10,X0,mu,sigma,theta(j,1),theta(j,2),@f,@g,@g_x);
Y(c,j) = A(1,1);
Z(c,j) = B(1,1);
c = c+1;
end
end

% Compute the absolute error for each scheme
eY = abs(Y - 3.65997);
eZ = abs(Z - 14.14823);

% FitzHugh-Nagamo equation
% eY = abs(Y - 0.5);
% eZ = abs(Z - 0.25);

%error for bid-ask spread (small)
% eY = abs(Y - 2.9584544);
% eZ = abs(Z - 0.55319);

%error for bid-ask spread (big)
% eY = abs(Y - 6.3748);
% eZ = abs(Z + 4.690);

% error for digital option

% eY = abs(Y - exp(-0.1*T)*normcdf(d2));
% eZ = abs(Z - exp(-0.1*T)*(1/sqrt(2*pi*T))*exp(-0.5*d2^2));

%error plot
subplot(1,2,1);
loglog(steps,eY);
ylabel('Log error Y_0');
xlabel('Steps');
subplot(1,2,2);
loglog(steps,eZ);
ylabel('Log error Z_0');
xlabel('Steps');
legend('Scheme A', 'Scheme B', 'Scheme C', 'Scheme D');

```

- θ -algorithm for RBSDEs.

```

function [X,Y,Z,K] = binomRBSDE(T,n,P,X0,mu,sigma,theta1,theta2,f,g,g_x,h)
% Initialize some variables
dt = T/n;
X = zeros(n+1,n+1);
X(1,1) = X0;
Y = zeros(n+1,n+1);
Z = zeros(n+1,n+1);
K = zeros(n+1,n+1);

% Compute tree values for the brownian motion and solve the forward SDE
% dX_t = mu(X_t) dt + sigma(X_t) dW_t
for j = 2:n+1
    X(1:j-1,j) = X(1:j-1,j-1) + mu(X(1:j-1,j-1))*dt - sigma(X(1:j-1,j-1))*sqrt(dt);
    X(j,j) = X(j-1,j-1) + mu(X(j-1,j-1))*dt + sigma(X(j-1,j-1))*sqrt(dt);
end

% Terminal conditions
Y(:,end) = g(X(:,end));
Z(:,end) = sigma(X(:,end)).*g_x(X(:,end));

% Compute numerical solution (Y_t,Z_t)
p = 0.5; %prob of up move

```

```

t2 = theta2;

for j = n:-1:1;
    if theta2 == 1/2 && j==n
        theta2 = 1;
    end

    Z(1:j,j) = (theta2 - 1)/(theta2)*((1-p)*Z(1:j,j+1) + p*Z(2:(j+1),j+1)) ...
        + 1/(theta2*sqrt(dt)) * (p*Y(2:(j+1),j+1) - (1-p)*Y(1:j,j+1)) + ...
        (1-theta2)/(theta2) *sqrt(dt)* ...
        (p*f(j+1,Y(2:(j+1),j+1),Z(2:(j+1),j+1)) - ...
        (1-p)*f(j+1,Y(1:j,j+1),Z(1:j,j+1)));
    picard = zeros(j,P);
    picard(:,1) = (1-p)*Y(1:j,j+1) + p*Y(2:(j+1),j+1);
    for k = 2:P
        picard(:,k) = (1-p)*Y(1:j,j+1) + p*Y(2:(j+1),j+1) + dt*theta1*f(j, ...
            picard(:,k-1), Z(1:j,j)) + ...
            dt*(1-theta1)*((1-p)*f(j+1,Y(1:j,j+1),Z(1:j,j+1)) + ...
            p*f(j+1,Y(2:(j+1),j+1), Z(2:(j+1),j+1)));
    end
    Y(1:j,j) = max(h(X(1:j,j)),picard(:,P));
    K(1:j,j) = Y(1:j,j) - picard(:,P);
    theta2 = t2;
end

end

```

- Main file that calls the binomRBSDE-function and computes the error.

```

clear all;
close all;
% Initialize variables
steps = 10:10:200;
theta = [0, 1;
         1/2, 1;
         1, 1;
         1/2, 1/2];

n_schemes = size(theta,1);
amount_steps = length(steps);
% initialize variables that will hold Y_0 and Z_0 for each scheme
Y = zeros(amount_steps,n_schemes);
Z = zeros(amount_steps,n_schemes);
% counter for the loop;
c = 1;
% configuration of the problem
drift = 0.05;
diffusion = 0.2;
X0 = 100;
T = 0.25;

% define mu and sigma for the forward SDE (functions of X_t)

mu = @(x) drift.*x;
sigma = @(x) diffusion.*x;

% compute the numerical solution for each scheme
for j = 1:n_schemes
    c = 1;
    for i = steps;
        [X,A,B,K] = ...
            binomRBSDE(T,i,5,X0,mu,sigma,theta(j,1),theta(j,2),@f_spread,@g,@g_x,@g);
        Y(c,j) = A(1,1);
    end
end

```

```

        Z(c,j) = B(1,1);
        c = c+1;
    end
end

%compute the absolute error for each scheme

% eY = abs(Y - 3.1696);
% eZ = abs(Z + 0.4256*X0*diffusion);

eY = abs(Y - 4.747036599903693);
eZ = abs(Z - 11.585125024175634);

% error plot
subplot(1,2,1);
loglog(steps,eY);
xlabel('M');
ylabel('Y_0');
subplot(1,2,2);
loglog(steps,eZ);
legend('Scheme A','Scheme B','Scheme C','Scheme D')
xlabel('M');
ylabel('Z_0');

```

- θ -algorithm for a FBSDE with a two-dimensional forward SDE.

```

function [X1,X2,Y,Z1,Z2] = ...
    binom2dBSDE(T,n,P,X0,mu,sigma,rho,thetal,theta2,f,g,g_x1,g_x2)

% Initialize variables
dt = T/n;

Y = zeros(n+1,n+1,n+1);
Z1 = zeros(n+1,n+1,n+1);
Z2 = zeros(n+1,n+1,n+1);

sigma1 = sigma(1,1);
sigma2 = sigma(2,2)/sqrt(1-rho^2);

u1 = 1 + mu(1)*dt + sigma1*sqrt(dt);
d1 = 1 + mu(1)*dt - sigma1*sqrt(dt);
u2 = 1 + mu(2)*dt + sigma2*sqrt(dt);
d2 = 1 + mu(2)*dt - sigma2*sqrt(dt);

% Terminal values of the binomial pyramids for both stocks

X1 = X0(1)*d1.^([n:-1:0]).*u1.^([0:n]');
X1 = repmat(flipud(X1'),1,n+1);
X2 = X0(2)*d2.^([n:-1:0]).*u2.^([0:n]');
X2 = repmat(fliplr(X2),n+1,1);

% Terminal conditions for the BSDE
tsigma = sigma';
Y(:, :, end) = g(X1,X2);
Z1(:, :, end) = tsigma(1,1)*X1.*g_x1(X1,X2) + tsigma(1,2)*X2.*g_x2(X1,X2);
Z2(:, :, end) = tsigma(2,1)*X1.*g_x1(X1,X2) + tsigma(2,2)*X2.*g_x2(X1,X2);

% Compute numerical solution (Y,t,Z,t)

t2 = theta2; % this is for the 'fix'

p_uu = 0.25*(1+rho);
p_dd = p_uu;

```

```

p_ud = 0.25*(1-rho);
p_du = p_ud;

inv_sigma = inv(sigma);
dW_1 = @(dz1,dz2) (dz1*sigma1*inv_sigma(1,1) + sigma2*dz2*inv_sigma(1,2));
dW_2 = @(dz1,dz2) (sigma1*dz1*inv_sigma(2,1) + sigma2*dz2*inv_sigma(2,2));

for k = n:-1:1;
    for i = 1:k
        if theta2 == 1/2 && k == n
            theta2 = 1;
        end

        Z1(1:k,i,k) = (theta2 - ...
            1)/(theta2)*(p_uu*Z1(1:k,i,k+1)+p_ud*Z1(1:k,i+1,k+1) + ...
            p_du*Z1(2:(k+1),i,k+1) + p_dd*Z1(2:(k+1),i+1,k+1))...
            + ...
            1/(theta2*sqrt(dt))*(p_uu*Y(1:k,i,k+1)*dW_1(1,1)+p_ud*Y(1:k,i+1,k+1)*dW_1(1,-1)
            + p_du*Y(2:(k+1),i,k+1)*dW_1(-1,1) + ...
            p_dd*Y(2:(k+1),i+1,k+1)*dW_1(-1,-1))...
            + (1-theta2)/theta2 * sqrt(dt) * ...
            (p_uu*f(k+1,Y(1:k,i,k+1),[Z1(1:k,i,k+1),Z2(1:k,i,k+1)])*dW_1(1,1) ...
            + ...
            p_ud*f(k+1,Y(1:k,i+1,k+1),[Z1(1:k,i+1,k+1),Z2(1:k,i+1,k+1)])*dW_1(1,-1) ...
            + ...
            p_du*f(k+1,Y(2:(k+1),i,k+1),[Z1(2:(k+1),i,k+1),Z2(2:(k+1),i,k+1)])*dW_1(-1,1) .
            + ...
            p_dd*f(k+1,Y(2:(k+1),i+1,k+1),[Z1(2:(k+1),i+1,k+1),Z2(2:(k+1),i+1,k+1)])*dW_1(-
        Z2(1:k,i,k) = (theta2 - ...
            1)/(theta2)*(p_uu*Z2(1:k,i,k+1)+p_ud*Z2(1:k,i+1,k+1) + ...
            p_du*Z2(2:(k+1),i,k+1) + p_dd*Z2(2:(k+1),i+1,k+1))...
            + 1/(theta2*sqrt(dt))*(p_uu*Y(1:k,i,k+1)*dW_2(1,1) + ...
            p_ud*Y(1:k,i+1,k+1)*dW_2(1,-1) + ...
            p_du*Y(2:(k+1),i,k+1)*dW_2(-1,1) + ...
            p_dd*Y(2:(k+1),i+1,k+1)*dW_2(-1,-1))...
            + (1-theta2)/theta2 * sqrt(dt) * ...
            (p_uu*f(k+1,Y(1:k,i,k+1),[Z1(1:k,i,k+1),Z2(1:k,i,k+1)])*dW_2(1,1) ...
            + ...
            p_ud*f(k+1,Y(1:k,i+1,k+1),[Z1(1:k,i+1,k+1),Z2(1:k,i+1,k+1)])*dW_2(1,-1) ...
            + ...
            p_du*f(k+1,Y(2:(k+1),i,k+1),[Z1(2:(k+1),i,k+1),Z2(2:(k+1),i,k+1)])*dW_2(-1,1) .
            + ...
            p_dd*f(k+1,Y(2:(k+1),i+1,k+1),[Z1(2:(k+1),i+1,k+1),Z2(2:(k+1),i+1,k+1)])*dW_2(-

        picard = zeros(k,P);
        picard(:,1) = (p_uu*Y(1:k,i,k+1) + p_ud*Y(1:k,i+1,k+1) + ...
            p_du*Y(2:(k+1),i,k+1) + p_dd*Y(2:(k+1),i+1,k+1));
        for l = 2:P
            picard(:,l) = (p_uu*Y(1:k,i,k+1) + p_ud*Y(1:k,i+1,k+1) + ...
                p_du*Y(2:(k+1),i,k+1) + p_dd*Y(2:(k+1),i+1,k+1))...
                + dt*theta1*f(k, picard(:,l-1), [Z1(1:k,i,k),Z2(1:k,i,k)]) + ...
                dt*(1-theta1)*(p_uu*f(k+1,Y(1:k,i,k+1),[Z1(1:k,i,k+1),Z2(1:k,i,k+1)]) ...
                + ...
                p_ud*f(k+1,Y(1:k,i+1,k+1),[Z1(1:k,i+1,k+1),Z2(1:k,i+1,k+1)]) ...
                + ...
                p_du*f(k+1,Y(2:(k+1),i,k+1),[Z1(2:(k+1),i,k+1),Z2(2:(k+1),i,k+1)]) ...
                + ...
                p_dd*f(k+1,Y(2:(k+1),i+1,k+1),[Z1(2:(k+1),i+1,k+1),Z2(2:(k+1),i+1,k+1)]));
        end

        Y(1:k,i,k) = picard(:,P);

        theta2 = t2;

```

```

        end
    end
end

```

- Main file that calls the binom2dBSDE-function and computes the error.

```

clear all;
close all;
% Initialize variables
steps = [10:10:100, 100:20:400];
theta = [0, 1;
         1/2, 1;
         1, 1;
         1/2, 1/2];
n_schemes = size(theta,1);
amount_steps = length(steps);

% Initialize variables that will hold Y_0 and Z_0 for each scheme
Y = zeros(amount_steps,n_schemes);
Z1 = zeros(amount_steps,n_schemes);
Z2 = zeros(amount_steps,n_schemes);
% Counter for the loop;
c = 1;
% Configuration of the problem
sigma1 = 0.25;
sigma2 = 0.3;
rho = 0.3;
sigma= [sigma1, 0;
        rho*sigma2, sqrt(1-rho^2)*sigma2];
mu = [0.1,0.1];
r=0.05;
X0 = [100,100];
T = 0.1;
% Compute numerical solution Y_0, Z^1_0 and Z^2_0
for j = 1:n_schemes
    c = 1;
    for i = steps;
        [X1,X2,A,B1,B2] = ...
            binom2dBSDE(T,i,5,X0,mu,sigma,rho,theta(j,1),theta(j,2),@f_spread,@g_spread,@g_x1_spread,@g_x2_spread);
        Y(c,j) = A(1,1);
        Z1(c,j) = B1(1,1);
        Z2(c,j) = B2(1,1);
        c = c+1;
    end
end

% Compute exact solution
[price,greeks]=spreadOption(X0,sigma1,sigma2,rho,T,r);

tsigma = sigma';
Z1sol = tsigma(1,1)*X0(1)*greeks.delta1 + tsigma(1,2)*X0(2)*greeks.delta2;
Z2sol = tsigma(2,1)*X0(1)*greeks.delta1 + tsigma(2,2)*X0(2)*greeks.delta2;

% Compute errors
eY = abs(Y - price);
eZ1 = abs(Z1 - Z1sol);
eZ2 = abs(Z2 - Z2sol);

% Error plot
subplot(1,3,1);
loglog(steps,eY,'-');
hold on;
subplot(1,3,2);

```

```
loglog(steps,eZ1,'-');  
hold on;  
subplot(1,3,3);  
loglog(steps,eZ2,'-');  
hold on;
```