A CONVERTIBLE-BOND-PRICING METHOD BASED ON BOND PRICES ON MARKETS

by

Qiang Shi

A dissertation submitted to the faculty of
The University of North Carolina at Charlotte
in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in
Applied Mathematics

Charlotte

2010

Approved by:

Dr. You-lan Zhu

Dr. Joel Avrin

Dr. Richard Buttimer

Dr. Hae-Soo Oh

Dr. Tony Plath

ABSTRACT

QIANG SHI. A convertible-bond-pricing method based on bond prices on markets. (Under the direction of DR. YOU-LAN ZHU)

This thesis is devoted to evaluating two-factor convertible bonds. Different zero-coupon bond curves are inputted when evaluating convertible bonds issued by companies with different credit ratings. Thus the effect of the company's credit on the price of the convertible bond is easily and accurately included during the computation. In the model for the interest rate, the parameters in the variance are determined from the market data by statistics and the market price of risk is determined by a zero-coupon bond curve through solving an inverse problem. When we price the convertible bond, a free-boundary problem is solved. A Singularity-Separating Method (SSM) is proposed in order to solve this problem efficiently. Taking the market data as input, we can easily, quickly, and reasonably obtain the price of a convertible bond.

ACKNOWLEDGMENTS

I feel really fortunate and privileged to have this wonderful experience to study at University of North Carolina at Charlotte. In the past seven years, I received plentiful and continuous help, support and encouragement from my advisor, faculty members in the Department of Mathematics and Statistics, my colleagues, my friends and my family. They kept me motivated all the way through the arduous process of pursuing a Ph.D. I am deeply indebted and would like to take this opportunity to extend my sincere thanks to all of them.

First and foremost, I would like to thank my advisor, Professor You-Lan Zhu, who is a true mentor to me both academically and intellectually. To him, I owe much more than I can say. His scientific insight, practical guidance and timely advice helped me to stay focused on the essentials in my research, and to work on problems at the forefront of my field. His support and encouragement enabled me to overcome many burdens and move smoothly through my graduate career.

I am grateful to all the faculty members of the Department of Mathematics and Statistics for their help and excellent teaching in Mathematics and Finance. In particular, I would like to express my appreciation to Professor Joel Avrin for his advice and help. I am truly grateful to him for I have benefited a great deal from working with him.

I would like to thank all other faculty members in my thesis committee: Professor Richard Buttimer, Professor Hae-Soo Oh and Professor Tony Plath for their helpful discussion and suggestions that have made my thesis more robust.

The wonderful administrative staff in the Department of Mathematics and Statistics, Mark Hamrick, Alicia A. Scott and Michael Shafer, have provided me numerous help on various issues during my years at UNCC. I thank them for everything they have done for me and for the department.

I have been extremely lucky to overlap with so many excellent colleagues. They formed a friendly, cooperative and supportive environment that made my graduate career more enjoyable and memorable. I am appreciated chances to work and live with those friends: Won-Tak Hong, Songtao Guo, Yunfei Wang and Chang Xiao. Specially, I want to thank Chenggao Yang, who spent plenty of hours discussing about research topics with me and helped me with achieving market data and sources.

Last but most importantly, I would like to thank my family. My wonderful and dearest wife, Xuefei, is the ultimate support for me in all aspects of life. I simply could not achieve anything without her. I thank my parents, my brother, they stand behind me unconditionally and with love for every adventure that I choose to explore.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1: INTRODUCTION

Convertible bonds are issued by a company and its holder has the option to exchange the bonds for the company's stock in the future. Brennan & Schwartz [2], Druskin et al [3], Sun [8], and Zhu & Sun [13] have discussed how to price convertible bonds. It is important to take the possibility of default of the issuer into account in some way when a convertible bond is valued [5]. As far as we know, so far nobody provides a model so that this fact can be considered during the evaluation of a convertible bond. In this paper we propose a model so that the possibility of default can be considered in a reasonable way.

To evaluate the price of a convertible bond, two factors are considered in our mathematical model. One factor is the price of the company's stock which may influence the decision of holders to exchange the bond or not, and the other factor is the interest rate.[1] In our interest rate model the parameters in the variance are determined from the market data by statistics. Furthermore, the influence of the market price of risk for the interest rate is carefully considered in this paper. A simple way to deal with the market price of risk is to ignore "real" market price of risk by defining some specific function as the coefficient associated with the market price of risk [11]. Here an inverse problem has been proposed to get the market price of risk from the real market data. Although solving the inverse problem will make the model more complicated, it is worthy to do so. Using such a model, the convertible bonds can be evaluated reasonably.

Black and Scholes [1] proposed a partial differential equation to price European put and call options and found closed-form solutions for them. Merton [7] expanded the

[1] In this thesis, "interest rate" means "spot interest rate" if it is not specified.

mathematical understanding of the options-pricing model and coined the term "Black-Scholes" options-pricing model. However, the problem of evaluating convertible bonds is an American-style option problem and the American-style derivatives are different from the European-style derivatives because the former can be exercised at any time before maturity and the latter can be only exercised at maturity. In order to price an American-style derivative, we need to solve a linear complementarity (LC) problem.

For a LC problem, usually there is a free boundary and no closed-form solution exists. A problem with a free boundary is difficult to solve accurately by numerical methods if the location of the free boundary is not determined accurately. This is because on the free boundary the solution has some weak singularity, more precisely, the second derivative is discontinuous, and on the two sides different equations need to be used. A process was proposed in the paper by Zhu and Sun [13] in which the location of the free boundary is tracked explicitly and proper boundary conditions on the free boundary are used when the PDE is solved. In this way the problem can be solved more efficiently. Another issue that makes this problem hard to be solved accurately at the time near the maturity is the appearance of another singularity due to the discontinuity of the first derivative in the final condition. In order to make the solution to be computed numerically smoother, in this paper we compute the difference between the solution and an analytic solution satisfying the same final condition and a similar PDE. In this way, a coarse mesh can give a quite accurate result even at the time near the maturity. In this method, the singularities on the free boundary and at the final condition almost are taken away, so we call our method a Singularity-Separating Method (SSM).

The problem is discretized by a second-order implicit finite difference scheme. Because of the free boundary, the system of finite difference equations is nonlinear. A Gauss-Seidel-type iteration method is applied to solve the system. Numerical results can be obtained quickly.

The thesis is organized as follows. We discuss in chapter 2 the two-factor convertible bond model and provide some useful definitions. Free boundary problems are discussed in chapter 3. The Singularity-Separating Method is described in chapter 4. How to determine the market price of risk for the interest rate is discussed in chapter 5. Spatial and temporal discretization and details of algorithm are described in chapter 6, and Some testing numerical results are reported in chapter 7. How to use the code and the theoretical prices of convertible bond are given in chapter 8. The conclusion and summary of the thesis is offered in chapter 9.

CHAPTER 2: THE TWO-FACTOR CONVERTIBLE BOND PROBLEM

The interest rate derivatives are financial products derived from interest rates; among various interest rate derivatives, we will focus on the two-factor convertible bond which is in the bond category.

A bond is a long-term contract under which the issuer promises to pay the holder a specified amount of money on a specified date. The specified amount is called the face value of the bond, and the specified date is named the maturity date. Usually, the holder is also paid a specified amount at fixed times during the life of the contract. Such a specified amount is called a coupon. If there is no coupon payment, the bond is known as a zero-coupon bond. Also, the bond holder must pay a certain amount of money to the issuer when the bond is purchased. Bonds may be issued by both governments or companies. The main purpose of a bond issue is to raise capital, and the up-front premium can be thought of as a loan to the government or the company. The theoretical price of a bond can be calculated as the present value of all cash flows that will be received by the holder of the bond using the appropriate zero rates as discount rates[5].

A convertible bond (or convertible debenture) is a type of bond that can be converted into shares of stock in the issuing company, usually at some pre-announced ratio. It is a hybrid security with debt- and equity-like features. Although it typically has a low coupon rate, the holder is compensated with the ability to convert the bond to common stock.

For bond holders, a convertible bond offers them protection against adverse movement in stock markets. When the issuing company's stock price is high, a convertible bond behaves like the stock. If the stock price declines to a certain price, the convertible feature of the bond becomes worthless. The bond can still be traded based on its yield, like a regular bond. There is a price level that a convertible bond will fall no further as long as the issuing company is able to pay its interest and principle upon maturity. In a word, a convertible bond offers holders the downside protection of fixed income and the upside return potential of equities.

For the issuing company, a convertible bond appears to be less expensive because it carries over lower coupons than the regular bond. Furthermore, the issuer can raise more fund by issuing convertible bonds than selling new common stocks because the conversion price is higher than the current stock price. Some issuers ignore the financial obligations of the convertible bond since they are confident that the stock price will appreciate so that bond holders will surely make the conversion. For these reasons, a convertible bond may be underpriced when they are originally issued.[8]

Since the convertible bond may be mispriced by both issuers and holders and it is not as liquid as equities or regular bonds, it is very important to find the theoretical value of a convertible bond fast and accurately.

A two-factor convertible bond is the bond whose value is dependent on two factors as time t evolves. The first factor is the stock price S and the second factor is the interest rate r.

General derivations of partial differential equations (PDEs) for financial derivatives are summarized with generalized Itô's lemma; furthermore, the mathematical model of evaluating a two-factor convertible bond is introduced in this chapter. In this thesis, a convertible-bond-pricing method is introduced to value such a two-factor convertible bond.

2.1 General equations for financial derivatives

In this section, we will derive general equations for financial derivatives depending on several random (state) variables. Some of those state variables may be the price of an asset traded on the market, and some of them may not. For example, the volatility may be taken as a state variable, but it is not a price of any asset traded on the market.

We start with models of random variables, and then derive equations for financial derivatives via generalization of Itô's lemma. A proof of the uniqueness of the solution of the final value problem of the PDE is provided when state variables satisfy the reversion conditions.

2.1.1 Random variables and reversion conditions

Assume that a financial derivative depends on time t and n random state variables $S_1, \ldots, S_n$. Each state variable satisfies a stochastic differential equation:

$$dS_i = a_i dt + b_i dX_i, \quad i = 1, \ldots, n, \tag{2.1}$$

where a_i and b_i are functions of $S_1, \ldots, S_n$ and t; $dX_i = \phi_i \sqrt{dt}$ are Wiener processes. Although ϕ_i and ϕ_j are both normal distributions, they are not necessarily the same random variable. They are correlated by

$$\mathbf{E}[\phi_i \phi_j] = \rho_{ij}, \quad -1 \leq \rho_{ij} \leq 1. \tag{2.2}$$

If $\rho_{ij} = 0$, then ϕ_i and ϕ_j are not correlated; if $\rho_{ij} = \pm 1$, then ϕ_i and ϕ_j are completely correlated. It is obvious that $\rho_{ii} = 1$.

Definition 2.1 (The reversion conditions) *For a random variable S of the form (2.1) with a lower bound S_l and an upper bound S_u, S satisfies the reversion conditions at boundaries $S = S_l$ and $S = S_u$ if*

$$\begin{cases} a(S_l, t) - b(S_l, t)\dfrac{\partial}{\partial S} b(S_l, t) \geq 0, & 0 \leq t \leq T, \\ b(S_l, t) = 0, & 0 \leq t \leq T, \end{cases} \tag{2.3}$$

and

$$\begin{cases} a(S_u,t) - b(S_u,t)\dfrac{\partial}{\partial S}b(S_u,t) \leq 0, & 0 \leq t \leq T, \\ b(S_u,t) = 0, & 0 \leq t \leq T. \end{cases} \tag{2.4}$$

Or when $b(S,t)$ is differentiable, above two conditions become

$$\begin{cases} a(S_l,t) \geq 0, & 0 \leq t \leq T, \\ b(S_l,t) = 0, & 0 \leq t \leq T, \end{cases} \tag{2.5}$$

and

$$\begin{cases} a(S_u,t) \leq 0, & 0 \leq t \leq T, \\ b(S_u,t) = 0, & 0 \leq t \leq T. \end{cases} \tag{2.6}$$

The reversion conditions are important in the uniqueness of the solutions of equations for derivatives.

2.1.2 Generalization of Itô's lemma

Assume the price of a financial derivative has the form $V(S_1,\ldots,S_n,t)$, and random variables $S_1,\ldots,S_n$ follow stochastic differential equations (2.1). Consider the Taylor expansion of $V(S_1+dS_1,\cdots,S_n+dS_n,t+dt)$ at $(S_1,\ldots,S_n,t)$:

$$\begin{aligned} & V(S_1+dS_1,\cdots,S_n+dS_n,t+dt) \\ = \; & V(S_1,\cdots,S_n,t) + \sum_{i=1}^{n}\frac{\partial V}{\partial S_i}dS_i + \frac{\partial V}{\partial t}dt \\ + \; & \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\frac{\partial^2 V}{\partial S_i \partial S_j}dS_i dS_j + \sum_{i=1}^{n}\frac{\partial^2 V}{\partial S_i \partial t}dS_i dt + \text{h.o.t.} \end{aligned} \tag{2.7}$$

Let $dV \equiv V(S_1+dS_1,\cdots,S_n+dS_n,t+dt) - V(S_1,\cdots,S_n,t)$, then

$$\begin{aligned} dV \; = \; & \sum_{i=1}^{n}\frac{\partial V}{\partial S_i}dS_i + \frac{\partial V}{\partial t}dt + \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\frac{\partial^2 V}{\partial S_i \partial S_j}dS_i dS_j \\ + \; & \sum_{i=1}^{n}\frac{\partial^2 V}{\partial S_i \partial t}dS_i dt + \text{h.o.t.} \end{aligned} \tag{2.8}$$

Because

$$\lim_{dt\to 0} dS_i dS_j/dt = b_i b_j \rho_{ij},$$

and $dS_i dt$ is a quantity of order $(dt)^{3/2}$, the relation above as $dt \to 0$ becomes:

$$dV = f dt + \sum_{i=1}^{n} \frac{\partial V}{\partial S_i} dS_i,$$

where f is given as

$$f = \frac{\partial V}{\partial t} + \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \frac{\partial^2 V}{\partial S_i \partial S_j} b_i b_j \rho_{ij}.$$

This is called the generalized Itô's lemma.

2.1.3 Derivation of equations for financial derivatives

Let $S_1, S_2, \cdots, S_n$ be n random variables. Furthermore assume that $S_1, S_2, \cdots, S_m$, $m \leq n$, are prices of some assets and the k-th asset pays a dividend $D_k(S_1, S_1, \cdots, S_n, t)dt$ during the time interval $[t, t+dt]$. Let $V(S_1, S_1, \cdots, S_n, t)$ be the price of a derivative security depending on $S_1, S_2, \cdots, S_n, t$.

Now let us derive the general equation. Suppose that there are $n - m + 1$ distinct financial derivatives dependent on S_1, S_2, $\cdots$, S_n and t, and let V_k stand for the value of the k-th derivative, $k = 0, 1, \cdots, n - m$. They could have different expiries or different exercise prices. Even some of the derivatives may depend on only some of the random variables. According to the generalized Itô's lemma, we have

$$dV_k = f_k dt + \sum_{i=1}^{n} \nu_{i,k} dS_i, \quad k = 0, \ldots, n - m,$$

where

$$f_k = \frac{\partial V_k}{\partial t} + \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \frac{\partial^2 V_k}{\partial S_i \partial S_j} b_i b_j \rho_{ij}$$

and

$$\nu_{i,k} = \frac{\partial V_k}{\partial S_i}, \quad i = 1, \ldots, n.$$

Furthermore, let the coupon payment for the k-th derivative during the time interval $[t, t + dt]$ be $K_k dt$. Consider a portfolio consisting of the $n - m + 1$ derivatives and the m assets, whose prices are $S_1, S_2, \cdots, S_m$:

$$\Pi = \sum_{k=0}^{n-m} \Delta_k V_k + \sum_{k=n-m+1}^{n} \Delta_k S_{k-n+m},$$

where Δ_k is the amount of the k-th derivative for $k = 0, 1, \cdots, n-m$ and the amount of the $(k-n+m)$-th asset, for $k = n-m+1, n-m+2, \cdots, n$. During the time interval $[t, t+dt]$, the holder of this portfolio will earn

$$\begin{aligned}
& \sum_{k=0}^{n-m} \Delta_k \left(dV_k + K_k dt\right) + \sum_{k=n-m+1}^{n} \Delta_k \left(dS_{k-n+m} + D_{k-n+m} dt\right) \\
= & \sum_{k=0}^{n-m} \Delta_k \left(f_k dt + \sum_{i=1}^{n} \nu_{i,k} dS_i + K_k dt\right) \\
& + \sum_{k=n-m+1}^{n} \Delta_k \left(dS_{k-n+m} + D_{k-n+m} dt\right) \\
= & \sum_{k=0}^{n-m} \Delta_k \left(f_k + K_k\right) dt + \sum_{i=1}^{n} \left(\sum_{k=0}^{n-m} \Delta_k \nu_{i,k}\right) dS_i \\
& + \sum_{i=1}^{m} \Delta_{i+n-m} dS_i + \sum_{k=n-m+1}^{n} \Delta_k D_{k-n+m} dt \\
= & \sum_{k=0}^{n-m} \Delta_k \left(f_k + K_k\right) dt + \sum_{i=1}^{m} \left(\sum_{k=0}^{n-m} \Delta_k \nu_{i,k} + \Delta_{i+n-m}\right) dS_i \\
& + \sum_{i=m+1}^{n} \left(\sum_{k=0}^{n-m} \Delta_k \nu_{i,k}\right) dS_i + \sum_{k=n-m+1}^{n} \Delta_k D_{k-n+m} dt.
\end{aligned}$$

Let us choose Δ_k such that

$$\sum_{k=0}^{n-m} \Delta_k \nu_{i,k} + \Delta_{i+n-m} = 0, \qquad i = 1, 2, \cdots, m$$

and

$$\sum_{k=0}^{n-m} \Delta_k \nu_{i,k} = 0, \qquad i = m+1, m+2, \cdots, n.$$

In this case the portfolio is risk-free, so its return rate is r, i.e.,

$$\begin{aligned}&\sum_{k=0}^{n-m} \Delta_k (f_k + K_k)\, dt + \sum_{k=n-m+1}^{n} \Delta_k D_{k-n+m} dt \\ = \; & r\left[\sum_{k=0}^{n-m} \Delta_k V_k + \sum_{k=n-m+1}^{n} \Delta_k S_{k-n+m}\right] dt\end{aligned}$$

or

$$\sum_{k=0}^{n-m} \Delta_k (f_k + K_k - rV_k) + \sum_{k=n-m+1}^{n} \Delta_k (D_{k-n+m} - rS_{k-n+m}) = 0.$$

This relation and the relations which the chosen Δ_k satisfy can be written together in a matrix form:

$$\begin{bmatrix} \nu_{1,0} & \nu_{1,1} & \cdots & \nu_{1,n-m} & 1 & 0 & \cdots & 0 \\ \nu_{2,0} & \nu_{2,1} & \cdots & \nu_{2,n-m} & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \nu_{m,0} & \nu_{m,1} & \cdots & \nu_{m,n-m} & 0 & 0 & \cdots & 1 \\ \nu_{m+1,0} & \nu_{m+1,1} & \cdots & \nu_{m+1,n-m} & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \nu_{n,0} & \nu_{n,1} & \cdots & \nu_{n,n-m} & 0 & 0 & \cdots & 0 \\ g_0 & g_1 & \cdots & g_{n-m} & h_1 & h_2 & \cdots & h_m \end{bmatrix} \begin{bmatrix} \Delta_0 \\ \Delta 1 \\ \vdots \\ \Delta_{n-m} \\ \Delta n-m+1 \\ \Delta n-m+2 \\ \vdots \\ \Delta_n \end{bmatrix} = 0,$$

where $g_k = f_k + K_k - rV_k$, $k = 0, 1, \cdots, n-m$ and $h_k = D_{k-n+m} - rS_{k-n+m}$, $k = 1, 2, \cdots, m$. In order for the system to have a non-trivial solution, the determinant of the matrix must be zero, or the $n+1$ row vectors of the matrix must be linearly dependent. Therefore, it is expected that the last row can be expressed as a linear combination of the other rows with coefficients $\tilde{\lambda}_1, \tilde{\lambda}_2, \cdots, \tilde{\lambda}_n$:

$$g_k = \sum_{i=1}^{n} \tilde{\lambda}_i \nu_{i,k}, \quad k = 0, 1, \cdots, n-m$$

and

$$h_k = \tilde{\lambda}_k, \quad k = 1, 2, \cdots, m.$$

Using the last m relations, we can rewrite the first $n-m+1$ relations as

$$g_k - \sum_{i=1}^{m} h_k \nu_{i,k} - \sum_{i=m+1}^{n} \tilde{\lambda}_i \nu_{i,k} = 0, \quad k = 0, 1, \cdots, n-m,$$

which means that any derivative satisfies an equation of the form

$$f + K - rV - \sum_{i=1}^{m} h_k \frac{\partial V}{\partial S_i} - \sum_{i=m+1}^{n} \tilde{\lambda}_i \frac{\partial V}{\partial S_i} = 0,$$

or

$$\begin{aligned} &\frac{\partial V}{\partial t} + \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} b_i b_j \rho_{ij} \frac{\partial^2 V}{\partial S_i \partial S_j} + \sum_{i=1}^{m} (rS_i - D_i) \frac{\partial V}{\partial S_i} \\ &- \sum_{i=m+1}^{n} \tilde{\lambda}_i \frac{\partial V}{\partial S_i} - rV + K = 0, \end{aligned}$$

where b_i, ρ_{ij} are given functions in the models of S_i, $\tilde{\lambda}_i$ are unknown functions which are independent of $V_0, V_1, \cdots, V_{n-m}$ and could depend on $S_1, S_2, \cdots, S_n$ and t, and K depends on the individual derivative security. Usually $\tilde{\lambda}_i$ is written in the form:

$$\tilde{\lambda}_i = \lambda_i b_i - a_i$$

and λ_i is called the market price of risk for S_i. Using this notation, we finally arrive at

$$\frac{\partial V}{\partial t} + \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} b_i b_j \rho_{ij} \frac{\partial^2 V}{\partial S_i \partial S_j} + \sum_{i=1}^{m} (rS_i - D_i) \frac{\partial V}{\partial S_i} + \sum_{i=m+1}^{n} (a_i - \lambda_i b_i) \frac{\partial V}{\partial S_i} - rV + K = 0. \tag{2.9}$$

So far, we have derived the general equation for financial derivatives.

2.2 A two-factor convertible bond problem

2.2.1 Models for random variables and reversion conditions

A two-factor convertible bond is the bond whose value is dependent on two factors as time t evolves. The first factor is the stock price S and the second factor is the interest rate r. Thus we have the price of the financial derivative (the convertible bond) depends on the stock price S, the interest rate r and the time t.

For two state variables S and r, they are satisfying following stochastic equations:

$$dS = \mu(S,t)Sdt + \sigma(S,t)SdX_1, \quad 0 \le S, \quad 0 \le t \le T, \tag{2.10}$$

and

$$dr = u(r,t)dt + w(r,t)dX_2, \quad r_l \le r \le r_u, \quad 0 \le t \le T. \tag{2.11}$$

Where μ and σ are the expected rate of return and volatility of the underlying stock. The feature of the interest rate process distinguishes our model to Brennan and Schwarz[2], or Longstaff and Schwarz [6] by ensuring that the bond valuation can be made consistent with the market time value. dX_1 and dX_2 are two different Wiener processes. The correlation between dX_1 and dX_2 is

$$E[dX_1, dX_2] = \rho dt, \quad -1 \le \rho \le 1, \tag{2.12}$$

where ρ is a negative constant number in our model. For $u(r,t)$ and $w(r,t)$, we assume that the first derivative of $w(r,t)$ is bounded and the reversion conditions are hold at $r = r_l$ and $r = r_u$:

$$\begin{cases} u(r_l,t) \ge 0, \\ w(r_l,t) = 0, \end{cases} \tag{2.13}$$

and

$$\begin{cases} u(r_u,t) \le 0, \\ w(r_u,t) = 0. \end{cases} \tag{2.14}$$

These conditions are called the reversion conditions because if they hold, then r will not go to the outside of the interval $[r_l, r_u]$ in the future if r is in $[r_l, r_u]$ now (see the book by Gihman and Skorohod [4]). In finance, if the first conditions in (2.13) and (2.14) hold, then it is said that the model has the property of mean reverting [12]. However, without the second conditions in (2.13) and (2.14), r still could go to the outside of the interval $[r_l, r_u]$.

2.2.2 The partial differential equation

Let $B_c = B_c(S, r, t)$ be the value of the two-factor convertible bond. We already showed that for any financial derivative, the general equation is (2.9), so $B_c(S, r, t)$ satisfies the following partial differential equation:

$$\frac{\partial B_c}{\partial t} + \mathbf{L}_{S,r} B_c + kZ = 0, \quad 0 \le S, \quad r_l \le r \le r_u, \quad 0 \le t \le T \tag{2.15}$$

with

$$\begin{aligned} \mathbf{L}_{S,r} &= \frac{1}{2}\sigma^2 S^2 \frac{\partial^2}{\partial S^2} + \rho\sigma S w \frac{\partial^2}{\partial S \partial r} + \frac{1}{2} w^2 \frac{\partial^2}{\partial r^2} + (r - D_0) S \frac{\partial}{\partial S} \\ &\quad + (u - \lambda w) \frac{\partial}{\partial r} - r, \end{aligned} \tag{2.16}$$

where D_0 is the dividend yield a holder of the stock receives per unit time, and kZ is the coupon payment a holder of the bond receives per unit time, k being the coupon rate and Z being the face value. $\lambda(t)$ is the market price of risk for the interest rate [9] and appears in the valuation equation because the state variable r is not a value of traded asset.

When the coupon payments are discrete, i.e., there are n individual payments k_i paid at time $t_i \le T, i = 1, \cdots, n,$, then the PDE (2.15) becomes

$$\frac{\partial B_c}{\partial t} + \mathbf{L}_{S,r} B_c + \sum_{i=1}^{n} k_i \delta(t - t_i) = 0, \quad 0 \le S, \quad r_l \le r \le r_u, \quad 0 \le t \le T, \tag{2.17}$$

where the operator $\mathbf{L}_{S,r}$ is given in (2.16).

2.2.3 The final condition

When the bond is mature, the holder of the bond will achieve the maximum value between the face value of the bond, Z, and the value of n shares of stock, nS, i.e.

$$B_c(S, r, T) = \max(Z, nS), \quad 0 \le S, \quad r_l \le r \le r_u. \tag{2.18}$$

This condition is called the final condition for a convertible bond.

2.2.4 The uniqueness of the final-value problem

Here we would like to point out that if (2.13) and (2.14) hold, then the equation (2.15) with a final condition like (2.18) has a unique solution, which means that if (2.13) and (2.14) hold, equation (2.15) needs no boundary condition in order to find a unique solution.

To prove this, we consider the following transformation:

$$\begin{cases} \xi = \dfrac{S}{S+Z/n}, \\ \\ \tau = T - t. \end{cases} \tag{2.19}$$

and

$$U(\xi, r, \tau) = \frac{B(S, r, t)}{nS+Z}. \tag{2.20}$$

we have

$$\begin{cases} \dfrac{\partial B}{\partial t} &= -(nS+Z)\dfrac{\partial U}{\partial \tau}, \\ \dfrac{\partial B}{\partial S} &= nU + (nS+Z)\dfrac{\partial U}{\partial \xi}\dfrac{\partial \xi}{\partial S}, \\ \dfrac{\partial B}{\partial r} &= (nS+Z)\dfrac{\partial U}{\partial r}, \\ \dfrac{\partial^2 B}{\partial S^2} &= (nS+Z)\dfrac{\partial^2 U}{\partial \xi^2}\left(\dfrac{\partial \xi}{\partial S}\right)^2, \\ \dfrac{\partial^2 B}{\partial S \partial r} &= \left((nS+Z)\dfrac{\partial^2 U}{\partial \xi \partial r}\dfrac{\partial \xi}{\partial S} + n\dfrac{\partial U}{\partial r}\right), \\ \dfrac{\partial^2 B}{\partial r^2} &= (nS+Z)\dfrac{\partial^2 U}{\partial r^2}. \end{cases} \tag{2.21}$$

Substituting these relations into (2.15) and (2.18), we have the following problem in the domain $[0,1] \times [r_l, r_u] \times [0,T]$ in (ξ, r, τ)-space:

$$\begin{cases} \dfrac{\partial U}{\partial \tau} = \mathbf{L}_{\xi,r} U + a_7, \\ U(\xi, r, 0) = max(1-\xi, \xi), \quad 0 \le \xi \le 1, \quad r_l \le r \le r_u, \quad 0 \le \tau \le T. \end{cases} \tag{2.22}$$

Here the operator $\mathbf{L}_{\xi,r}$ is defined as follows:

$$\mathbf{L}_{\xi,r} = \frac{1}{2}a_1^2\frac{\partial^2}{\partial \xi^2} + a_2\frac{\partial^2}{\partial \xi \partial r} + \frac{1}{2}a_3^2\frac{\partial^2}{\partial r^2} + a_4\frac{\partial}{\partial \xi} + a_5\frac{\partial}{\partial r} + a_6$$

and the expressions for a_i, $i = 1, \cdots, 7$ are

$$\begin{aligned} a_1 &= \sigma\xi(1-\xi), \\ a_2 &= \rho\sigma w\xi(1-\xi) = \rho a_1 a_3, \\ a_3 &= w, \\ a_4 &= (r-D_0)\xi(1-\xi), \\ a_5 &= u - (\lambda - \rho\sigma\xi)w, \\ a_6 &= (r-D_0)\xi - r, \\ a_7 &= k(1-\xi). \end{aligned}$$

Since the transform (2.19), (2.10) can be written as

$$d\xi = [\mu\xi(1-\xi) - \sigma^2\xi^2(1-\xi)]dt + \sigma\xi(1-\xi)dX_1, \quad 0 \leq \xi \leq 1, \quad 0 \leq t \leq T. \tag{2.23}$$

We can find that ξ satisfies the reversion conditions (2.3) and (2.4) at the boundaries $\xi = 0$ and $\xi = 1$. Based the reversion conditions (2.13) and (2.14), we will prove that this final value problem has a unique solution.

Proof.[14] Suppose that u_1 and u_2 are two solutions of (2.22) and let $u \equiv u_1 - u_2$. It is clear that u is the solution of (2.22) with $u(\xi, r, 0) \equiv 0$.

Define

$$W(\tau) = \int_{r_l}^{r_u} \int_0^1 u^2(\xi, r, T-\tau) d\xi dr. \tag{2.24}$$

Since the equation in (2.22) can be rewritten as

$$\begin{aligned} \frac{\partial u}{\partial \tau} &= \frac{1}{2}\frac{\partial}{\partial \xi}\left(a_1^2 \frac{\partial u}{\partial \xi} + a_2 \frac{\partial u}{\partial r}\right) + \frac{1}{2}\frac{\partial}{\partial r}\left(a_2 \frac{\partial u}{\partial \xi} + a_3^2 \frac{\partial u}{\partial r}\right) \\ &+ \left(a_4 - a_1 \frac{\partial a_1}{\partial \xi} - \frac{1}{2}\frac{\partial a_2}{\partial r}\right)\frac{\partial u}{\partial \xi} \\ &+ \left(a_5 - a_3 \frac{\partial a_3}{\partial r} - \frac{1}{2}\frac{\partial a_2}{\partial \xi}\right)\frac{\partial u}{\partial r} + a_6 u, \end{aligned}$$

we have

$$\begin{aligned}
\frac{1}{2}\frac{dW(\tau)}{d\tau} &= \int_{r_l}^{r_u}\int_0^1 u\frac{\partial u}{\partial \tau}d\xi dr \\
&= \int_{r_l}^{r_u}\int_0^1 \frac{u}{2}\frac{\partial}{\partial \xi}\left(a_1^2\frac{\partial u}{\partial \xi}+a_2\frac{\partial u}{\partial r}\right)d\xi dr \\
&+ \int_{r_l}^{r_u}\int_0^1 \frac{u}{2}\frac{\partial}{\partial r}\left(a_2\frac{\partial u}{\partial \xi}+a_3^2\frac{\partial u}{\partial r}\right)d\xi dr \\
&+ \int_{r_l}^{r_u}\int_0^1 u\left(a_4-a_1\frac{\partial a_1}{\partial \xi}-\frac{1}{2}\frac{\partial a_2}{\partial r}\right)\frac{\partial u}{\partial \xi}d\xi dr \\
&+ \int_{r_l}^{r_u}\int_0^1 u\left(a_5-a_3\frac{\partial a_3}{\partial r}-\frac{1}{2}\frac{\partial a_2}{\partial \xi}\right)\frac{\partial u}{\partial r}d\xi dr \\
&+ \int_{r_l}^{r_u}\int_0^1 a_6u^2 d\xi dr. \qquad (2.25)
\end{aligned}$$

Consider the first four terms of the right hand side of the equation (2.25) with the reversion conditions (2.13) and (2.14).

The first and second term can be rewritten as

$$\begin{aligned}
&\int_{r_l}^{r_u}\int_0^1 \frac{u}{2}\frac{\partial}{\partial \xi}\left(a_1^2\frac{\partial u}{\partial \xi}+a_2\frac{\partial u}{\partial r}\right)d\xi dr \\
=\ &\frac{1}{2}\int_{r_l}^{r_u}\left\{\left[u\left(a_1^2\frac{\partial u}{\partial \xi}+a_2\frac{\partial u}{\partial r}\right)\right]\Bigg|_0^1-\int_0^1\left(a_1^2\frac{\partial u}{\partial \xi}+a_2\frac{\partial u}{\partial r}\right)\frac{\partial u}{\partial \xi}d\xi\right\}dr \\
=\ &-\frac{1}{2}\int_{r_l}^{r_u}\int_0^1\left(a_1^2\frac{\partial u}{\partial \xi}+a_2\frac{\partial u}{\partial r}\right)\frac{\partial u}{\partial \xi}d\xi dr \qquad (2.26)
\end{aligned}$$

and

$$\begin{aligned}
&\int_{r_l}^{r_u}\int_0^1 \frac{u}{2}\frac{\partial}{\partial r}\left(a_2\frac{\partial u}{\partial \xi}+a_3^2\frac{\partial u}{\partial r}\right)d\xi dr \\
=\ &\frac{1}{2}\int_0^1\left\{\left[u\left(a_2\frac{\partial u}{\partial \xi}+a_3^2\frac{\partial u}{\partial r}\right)\right]\Bigg|_{r_l}^{r_u}-\int_{r_l}^{r_u}\left(a_2\frac{\partial u}{\partial \xi}+a_3^2\frac{\partial u}{\partial r}\right)\frac{\partial u}{\partial r}dr\right\}d\xi \\
=\ &-\frac{1}{2}\int_{r_l}^{r_u}\int_0^1\left(a_2\frac{\partial u}{\partial \xi}+a_3^2\frac{\partial u}{\partial r}\right)\frac{\partial u}{\partial r}d\xi dr. \qquad (2.27)
\end{aligned}$$

Since for any r, $a_1(0,r,t)=0$, then $\frac{\partial}{\partial r}(\rho a_1a_3)|_{\xi=0}=0$. Similarly, we have $\frac{\partial}{\partial r}(\rho a_1a_3)|_{\xi=1}=0$, $\frac{\partial}{\partial \xi}(\rho a_1a_3)|_{r=r_l}=0$ and $\frac{\partial}{\partial \xi}(\rho a_1a_3)|_{r=r_u}=0$. Based on these equalities and reversion conditions (2.13) and (2.14), the third and fourth term of the right hand side of the equation (2.25) can be rewritten as

$$
\begin{aligned}
& \int_{r_l}^{r_u} \int_0^1 u \left(a_4 - a_1 \frac{\partial a_1}{\partial \xi} - \frac{1}{2} \frac{\partial a_2}{\partial r} \right) \frac{\partial u}{\partial \xi} d\xi dr \\
= & \ \frac{1}{2} \int_{r_l}^{r_u} \left\{ \left[u^2 \left(a_4 - a_1 \frac{\partial a_1}{\partial \xi} - \frac{1}{2} \frac{\partial a_2}{\partial r} \right) \right] \Big|_0^1 \right. \\
- & \left. \int_0^1 u^2 \frac{\partial}{\partial \xi} \left(a_4 - a_1 \frac{\partial a_1}{\partial \xi} - \frac{1}{2} \frac{\partial a_2}{\partial r} \right) d\xi \right\} dr \\
\leq & \ -\frac{1}{2} \int_{r_l}^{r_u} \int_0^1 u^2 \frac{\partial}{\partial \xi} \left(a_4 - a_1 \frac{\partial a_1}{\partial \xi} - \frac{1}{2} \frac{\partial a_2}{\partial r} \right) d\xi dr
\end{aligned} \tag{2.28}
$$

and

$$
\begin{aligned}
& \int_{r_l}^{r_u} \int_0^1 u \left(a_5 - a_3 \frac{\partial a_3}{\partial r} - \frac{1}{2} \frac{\partial a_2}{\partial \xi} \right) \frac{\partial u}{\partial r} d\xi dr \\
= & \ \frac{1}{2} \int_0^1 \left\{ \left[u^2 \left(a_5 - a_3 \frac{\partial a_3}{\partial r} - \frac{1}{2} \frac{\partial a_2}{\partial \xi} \right) \right] \Big|_{r_l}^{r_u} \right. \\
- & \left. \int_{r_l}^{r_u} u^2 \frac{\partial}{\partial r} \left(a_5 - a_3 \frac{\partial a_3}{\partial r} - \frac{1}{2} \frac{\partial a_2}{\partial \xi} \right) dr \right\} d\xi \\
\leq & \ -\frac{1}{2} \int_{r_l}^{r_u} \int_0^1 u^2 \frac{\partial}{\partial r} \left(a_5 - a_3 \frac{\partial a_3}{\partial r} - \frac{1}{2} \frac{\partial a_2}{\partial \xi} \right) d\xi dr.
\end{aligned} \tag{2.29}
$$

Adding (2.26) and (2.27) together and considering that $|\rho| \leq 1$ and $a_2^2 - a_1^2 a_3^2 = (\rho^2 - 1) a_1^2 a_3^2 \leq 0$, we have

$$
\begin{aligned}
& \int_{r_l}^{r_u} \int_0^1 \frac{u}{2} \frac{\partial}{\partial \xi} \left(a_1^2 \frac{\partial u}{\partial \xi} + a_2 \frac{\partial u}{\partial r} \right) d\xi dr \\
+ & \int_{r_l}^{r_u} \int_0^1 \frac{u}{2} \frac{\partial}{\partial r} \left(a_2 \frac{\partial u}{\partial \xi} + a_3^2 \frac{\partial u}{\partial r} \right) d\xi dr \\
= & \ -\frac{1}{2} \int_{r_l}^{r_u} \int_0^1 \left[\left(a_1 \frac{\partial u}{\partial \xi} \right)^2 + 2 a_2 \frac{\partial u}{\partial \xi} \frac{\partial u}{\partial r} + \left(a_3 \frac{\partial u}{\partial r} \right)^2 \right] d\xi dr \\
\leq & \ 0.
\end{aligned} \tag{2.30}
$$

Substitute (2.26)-(2.29) into (2.25) and applying (2.30), we have

$$\begin{aligned}\frac{1}{2}\frac{dW(\tau)}{d\tau} &\leq -\frac{1}{2}\int_{r_l}^{r_u}\int_0^1 u^2\left\{\frac{\partial}{\partial\xi}\left(a_4 - a_1\frac{\partial a_1}{\partial\xi} - \frac{1}{2}\frac{\partial a_2}{\partial r}\right)\right. \\ &+ \left.\frac{\partial}{\partial r}\left(a_5 - a_3\frac{\partial a_3}{\partial r} - \frac{1}{2}\frac{\partial a_2}{\partial\xi}\right) - 2a_6\right\}d\xi dr\end{aligned}$$

And also we can easily find that there exists a constant c_1 such that

$$\max_{\xi\in[0,1],r\in[r_l,r_u]}\left|\frac{\partial}{\partial\xi}\left[a_4 - a_1\frac{\partial a_1}{\partial\xi} - \frac{1}{2}\frac{\partial a_2}{\partial r}\right] + \frac{\partial}{\partial r}\left[a_5 - a_3\frac{\partial a_3}{\partial r} - \frac{1}{2}\frac{\partial a_2}{\partial\xi}\right] - 2a_6\right| \leq c_1,$$

Therefore, we can get

$$\frac{1}{2}\frac{dW(\tau)}{d\tau} \leq \frac{1}{2}c_1 W(\tau),$$

then according to the Gronwall inequality, we get

$$0 \leq W(\tau) \leq e^{c_1\tau}W(0).$$

Notice that $W(0) = 0$, thus $W(\tau) \equiv 0$, which leads to the fact that $u = u_1 - u_2 \equiv 0$, or $u_1 \equiv u_2$. In another word, the solution of problem (2.22) is unique. •

This result also means we have a a unique solution for the PDE (2.15) with the final condition (2.18).

Here we give some intuitive explanations. When $w(r_l, t) = 0$, equation (2.15) at $r = r_l$ degenerates to

$$\frac{\partial B_c}{\partial t} + \frac{1}{2}\sigma^2 S^2\frac{\partial^2 B_c}{\partial S^2} + (r - D_0)S\frac{\partial B_c}{\partial S} + u\frac{\partial B_c}{\partial r} - rB_c + kZ = 0. \tag{2.31}$$

According to (2.13), $u(r_l, t) \geq 0$, $B_c(S, r_l, t)$ is determined by $B_c(S, r, t)$ in the domain $[0, \infty) \times [r_l, r_u] \times [t, T]$ because (2.31) has hyperbolic properties in the r-direction. This means that no boundary condition is needed at $r = r_l$. Similarly, no boundary condition is needed at $r = r_u$ because of (2.14).

At $S = 0$, equation (2.15) becomes

$$\frac{\partial B_c}{\partial t} + \frac{1}{2}w^2\frac{\partial^2 B_c}{\partial r^2} + (u - \lambda w)\frac{\partial B_c}{\partial r} - rB_c + kZ = 0. \tag{2.32}$$

$B_c(0, r, t)$ is determined by this bond equation and the final condition[2] at $S = 0$. There is no need to specify a condition when $S \to \infty$, just like the Black-Scholes equation.

The finial value problem

$$\begin{cases} \dfrac{\partial B_c}{\partial t} + \mathbf{L}_{S,r}B_c + kZ = 0, \quad 0 \le S, \quad r_l \le r \le r_u, \quad 0 \le t \le T, \\ B_c(S, r, T) = \max(Z, nS), \end{cases} \tag{2.33}$$

has a unique solution if two reversion conditions (2.13) and (2.14) are true. If there is no dividend, i.e., $D_0 = 0$, (2.33) gives the price of the convertible bond. However, if there is some dividend, i.e., $D_0 > 0$, then the solution of this problem might not satisfy the condition $B_c(S, r, t) \ge nS$.

[2]The final condition in this case is $B_c(0, r, T) = Z$. Because $B_c(0, r, t)$ has such a final condition and satisfies (2.32), $B_c(0, r, t)$ actually gives a bond price.

CHAPTER 3: LC PROBLEM AND FREE BOUNDARY PROBLEM

A convertible bond will be either converted into n shares of the company's stock or paid at its face vale Z at maturity. Moreover, a convertible bond can be exercised at any time before maturity (American style), thus its value must be no less than the value of n shares of stock. We call this constraint of the convertible bond the conversion constraint. In some cases, the solution of (2.33) will satisfy this conversion constraint. In these cases, (2.33) determines the value of a convertible bond. For example, if there is no dividend, i.e., $D_0 = 0$, (2.33) gives the price of the convertible bond. However, if there is some dividend, i.e., $D_0 > 0$, then the price of the convertible bond is the solution of a linear complementarity problem in the whole computational domain. In such a case, the equation (2.15) is not always true in the whole computational domain, and there exists a free boundary dividing the domain into two subdomains. The equation (2.15) is valid in one subdomain and in the other subdomain, it cannot be used to evaluate the convertible bond. Instead, the value of the convertible bond is determined by the conversion constraint.

To determine the value of the convertible bond in the whole domain, we need to determine the location of this free boundary and solve the PDE where it is valid. A free boundary problem (FBP) is reformulated from the linear complementarity (LC) problem with the location of the free boundary.

3.1 The conversion constraint of American style derivatives

At its maturity, the convertible bond is either converted into n shares of the company's stock or paid at its face value Z, whichever is more valuable. Therefore it should have a value when $t = T$:

$$B_c(S, r, T) = \max(nS, Z). \tag{3.1}$$

Since the bond can be converted into n shares of stock at any time before maturity, its price must be no less than the price of n shares of stock:

$$B_c(S, r, t) \geq nS \quad 0 \leq S, \quad 0 \leq t \leq T, \tag{3.2}$$

and this constraint is called the conversion constraint. If this is not the case, there will be an arbitrage opportunity: an instant profit can be realized by buying a convertible bond, converting it, and selling the stocks immediately at the market price.

3.2 The Linear Complementarity problem

In some cases, the solution of PDE in (2.33) will satisfy this conversion constraint. In these cases, (2.33) determines the value of a convertible bond. For example, if there is no dividend, i.e., $D_0 = 0$, (2.33) gives the price of the convertible bond. However, if there is some dividend, i.e., $D_0 > 0$, then the price of the convertible bond is the solution of a linear complementarity problem. Before we formula the convertible bond problem as LC problem, we would like to show the following theorem.

Theorem 3.1 *Let $\mathbf{L}_{S,r,t}$ be an operator in a bond problem in the form:*

$$\mathbf{L}_{S,r,t} = a(S,r,t)\frac{\partial^2}{\partial S^2} + b(S,r,t)\frac{\partial^2}{\partial S \partial r} + c(S,r,t)\frac{\partial^2}{\partial r^2} + d(s,r,t)\frac{\partial}{\partial S} + e(S,r,t)\frac{\partial}{\partial r} + f(S,r,t),$$

and $G(S,r,t)$ be the constraint function for a convertible bond. Furthermore, we assume that $\frac{\partial G}{\partial t} + \mathbf{L}_{S,r,t}G$ exists. Suppose $V(S,r,t^*) = G(S,r,t^*)$ on an open interval (A_1, B_1) on the S-axis and (C_1, D_1) on the r-axis. Let $t = t^* - \Delta t$, where Δt is a sufficiently small positive number, let (A, B) be an open interval in (A_1, B_1) and let let (C, D) be an open interval in (C_1, D_1). Show the following conclusions: If for any

$S \in (A, B)$ and $r \in (C, D)$,

$$\frac{\partial G}{\partial t}(S, r, t^*) + \mathbf{L}_{S,r,t^*} G(S, r, t^*) + g(S, r, t^*) \geq 0,$$

then the value $V(S, r, t)$ determined by the equation

$$\frac{\partial V}{\partial t}(S, r, t) + \mathbf{L}_{S,r,t} V(S, r, t) + g(S, r, t) = 0$$

satisfies the condition $V(S, r, t) - G(S, r, t) \geq 0$ on $(A, B) \times (C, D)$; and if for any $S \in (A, B)$ and $r \in (C, D)$,

$$\frac{\partial G}{\partial t}(S, r, t^*) + \mathbf{L}_{S,r,t^*} G(S, r, t^*) + g(S, r, t^*) < 0,$$

then the equation

$$\frac{\partial V}{\partial t}(S, r, t) + \mathbf{L}_{S,r,t} V(S, r, t) + +g(S, r, t) = 0$$

cannot give a solution satisfying the condition $V(S, r, t) - G(S, r, t) \geq 0$ for any $S \in (A, B)$ and $r \in (C, D)$.

Proof. Because $V(S, r, t^*) = G(S, r, t^*)$, the fact that $V(S, r, t) - G(S, r, t) \geq 0$ holds for any $t = t^* - \Delta t$, Δt being a sufficiently small positive number, is equivalent to the fact that at time t^*, $V(S, r, t) - G(S, r, t)$ is a nonincreasing function with respect to t, that is,

$$\frac{\partial V}{\partial t}(S, r, t^*) - \frac{\partial G}{\partial t}(S, r, t^*) \leq 0.$$

If

$$\frac{\partial G}{\partial t}(S, r, t^*) + \mathbf{L}_{S,r,t^*} G(S, r, t^*) + g(S, r, t^*) \geq 0$$

and

$$\frac{\partial V}{\partial t}(S, r, t^*) + \mathbf{L}_{S,r,t^*} V(S, r, t^*) + g(S, r, t^*) = \frac{\partial V}{\partial t}(S, r, t^*) + \mathbf{L}_{S,r,t^*} G(S, r, t^*) + g(S, r, t^*) = 0$$

then

$$\frac{\partial G}{\partial t}(S,r,t^*) \geq -\mathbf{L}_{S,r,t^*}G(S,r,t^*) - g(S,r,t^*) = \frac{\partial V}{\partial t}(S,r,t^*)$$

or

$$\frac{\partial V}{\partial t}(S,r,t^*) - \frac{\partial G}{\partial t}(S,r,t^*) \leq 0.$$

Therefore, in this case we can use the equation

$$\frac{\partial V}{\partial t}(S,r,t) + \mathbf{L}_{S,r,t}V(S,r,t) + g(S,r,t) = 0$$

to get a solution satisfying the condition $V(S,r,t) - G(S,r,t) \geq 0$.

If

$$\frac{\partial G}{\partial t}(S,r,t^*) + \mathbf{L}_{S,r,t^*}G(S,r,t^*) + g(S,r,t^*) < 0$$

and

$$\frac{\partial V}{\partial t}(S,r,t^*)+\mathbf{L}_{S,r,t^*}V(S,r,t^*)+g(S,r,t^*) = \frac{\partial V}{\partial t}(S,r,t^*)+\mathbf{L}_{S,r,t^*}G(S,r,t^*)+g(S,r,t^*) = 0$$

then

$$\frac{\partial G}{\partial t}(S,r,t^*) < -\mathbf{L}_{S,r,t^*}G(S,r,t^*) - g(S,r,t^*) = \frac{\partial V}{\partial t}(S,r,t^*)$$

or

$$\frac{\partial V}{\partial t}(S,r,t^*) - \frac{\partial G}{\partial t}(S,r,t^*) > 0,$$

which will cause $V(S,r,t) - G(S,r,t) < 0$ for any $t = t^* - \Delta t$. Therefore, we cannot get the solution by using the equation

$$\frac{\partial V}{\partial t}(S,r,t) + \mathbf{L}_{S,r,t}V(S,r,t) + g(S,r,t) = 0.$$

Instead we have to let $V(S,r,t) - G(S,r,t) = 0$ in order to get a solution satisfying the condition $V(S,r,t) - G(S,r,t) \geq 0$. •

By using this theorem, let $G(S,r,t) = nS$ then we can easily find the LC problem for the convertible bond problem should be

$$\begin{cases} (\dfrac{\partial B_c}{\partial t} + \mathbf{L}_{S,r}B_c + kZ)(B_c(S,r,t) - nS) &= 0, \\ \dfrac{\partial B_c}{\partial t} + \mathbf{L}_{S,r}B_c + kZ &\leq 0, \\ B_c(S,r,t) - nS &\geq 0, \\ B_c(S,r,T) &= \max(Z, nS) \geq nS, \end{cases} \tag{3.3}$$

in the domain $[0,\infty) \times [r_l, r_u] \times [0,T]$. Or equivalently,

$$\begin{cases} \min\left(-\left(\dfrac{\partial B_c}{\partial t} + \mathbf{L}_{S,r}B_c + kZ\right),\ B_c(S,r,t) - nS\right) = 0, \\ B_c(S,r,T) = \max(Z, nS) \geq nS \end{cases} \tag{3.4}$$

in the domain $[0,\infty) \times [r_l, r_u] \times [0,T]$.

3.3 The free boundary problem for two-factor convertible bond problem

3.3.1 The location of the free boundary

We will reformulate the LC problem (3.4) into a free boundary problem. Before we start, we will find the location of this free boundary by using the following procedure.

For a two-factor convertible bond problem, the free boundary has the form:

$$S = S_f(r,t).$$

Using Theorem 3.1, we can easily determine the location of the free boundary at maturity, namely, the starting points of the free boundary. Consequently, there are still two subdomains, in one of which B_c satisfies equation (2.15) while in the other of which it is determined by $B_c = nS$.

Combining these two cases, the whole domain is always divided into two subdomains, and at $t = T$, they are separated by the free boundary as seen in the Figure (3.1):

$$S_f(r,T) = \max(\frac{Z}{n}, \frac{kZ}{D_0 n}).$$

This can be proved as follows.

(i) First, let us consider the case when $\frac{k}{D_0} \leq 1$.

(a) $S > \frac{Z}{n}$

The final condition gives

$$B_c(S, r, T) = \max(nS, Z) = nS,$$

and the conversion is optimal at maturity. The equation (2.15) gives

$$\begin{aligned}\frac{\partial B_c}{\partial t} &= -(\mathbf{L}_{S,r}B_c + kZ) \\ &= -((r - D_0)nS - rnS + kZ) \\ &= D_0 ns - kZ.\end{aligned}$$

Because $S > \frac{Z}{n} \geq \frac{kZ}{D_0 z}$, we have $D_0 nS - kZ > 0$, thus

$$\frac{\partial B_c}{\partial t} > 0, \quad \text{at} \quad t = T.$$

Therefore,

$$B_c(S, r, T - \Delta t) < B_c(S, r, T) = nS$$

for a small positive Δt. This conflicts with the conversion constraint (3.2), i.e., $B_c(S, r, t) \geq nS$, for any $t \in [0, T]$. By the arbitrage argument, this situation cannot last long. In other words, equation (2.15) is not satisfied when $S > \frac{Z}{n}$ for time close to T.

(b) $S < \frac{Z}{n}$

The final condition gives

$$B_c(S, r, T) = \max(nS, Z) = Z,$$

and it is not wise to convert the bond into n shares of stock at maturity.

The conversion constraint (3.2) can be written as

$$\begin{aligned} B_c(S,r,T-\Delta t) &= B_c(S,r,T) - \frac{\partial B_c(S,r,T-\Delta t)}{\partial t}\Delta t \\ &= Z - \frac{\partial B_c}{\partial t}\Delta t \\ &\geq nS. \end{aligned} \tag{3.5}$$

If the equation (2.15) holds, then

$$\frac{\partial B_c}{\partial t} = rZ - kZ.$$

Therefore, when $r \leq k$,

$$\begin{aligned} & Z - \frac{\partial B_c}{\partial t}\Delta t \\ = \; & Z - (r-k)Z\Delta t \\ \geq \; & Z > nS. \end{aligned}$$

The conversion constraint is satisfied.

When $r > k$, we can choose Δt such that

$$\Delta t < \frac{Z - nS}{(r-k)Z}.$$

Then

$$\begin{aligned} & Z - \frac{\partial B_c}{\partial t}\Delta t \\ = \; & Z - (r-k)Z\Delta t \\ > \; & Z - (Z - nS) = nS. \end{aligned}$$

The conversion constraint is also satisfied. This means that both equation (2.15) and conversion constraint (3.2) can be fulfilled when $S < \frac{Z}{n}$.

Consequently, if $\frac{k}{D_0} \leq 1$, and t is less than but sufficiently close to T, the domain has been divided into two subdomains by the free boundary

$$S = S_f = \frac{Z}{n}.$$

When $S > \frac{Z}{n}$, the price is nS, and the equation (2.15) does not apply; when $S < \frac{Z}{n}$, the price is determined by the equation (2.15).

(ii) Then, let us consider the case when $\frac{k}{D_0} > 1$.

(a) $S > \frac{kZ}{D_0 n}$

In this case, $S > \frac{kZ}{D_0 n} > \frac{Z}{n}$, then $B_c(S, r, T) = \max(nS, Z) = nS$. The equation (2.15) does not apply for the same reason as in (i)(a).

(b) $\frac{Z}{n} < S < \frac{kZ}{D_0 n}$

This gives $B_c(S, r, T) = \max(nS, Z) = nS$. If (2.15) holds,

$$\begin{aligned}\frac{\partial B_c}{\partial t} &= -(\mathbf{L}_{S,r} B_c + kZ) \\ &= -((r - D_0)nS - rnS + kZ) \\ &= D_0 ns - kZ.\end{aligned}$$

Because $\frac{Z}{n} < S < \frac{kZ}{D_0 n}$, $D_0 ns - kZ < 0$, then

$$\frac{\partial B_c}{\partial t} = D_0 ns - kZ < 0 \quad \text{at} \quad t = T.$$

This means that $B_c(S, r, T - \Delta t) > B_c(S, r, T) = nS$ for sufficiently small positive Δt. There is no conflict with the constraint (3.2). Therefore equation (2.15) applies.

(c) $S < \frac{kZ}{D_0 n}$

The equation (2.15) can be used to evaluate the price for the same reason in (i)(b).

Consequently, if $\frac{k}{D_0} > 1$, and t is less than but sufficiently close to T, the domain has been divided into two subdomains by the free boundary

$$S_f = \frac{kZ}{D_0 n}.$$

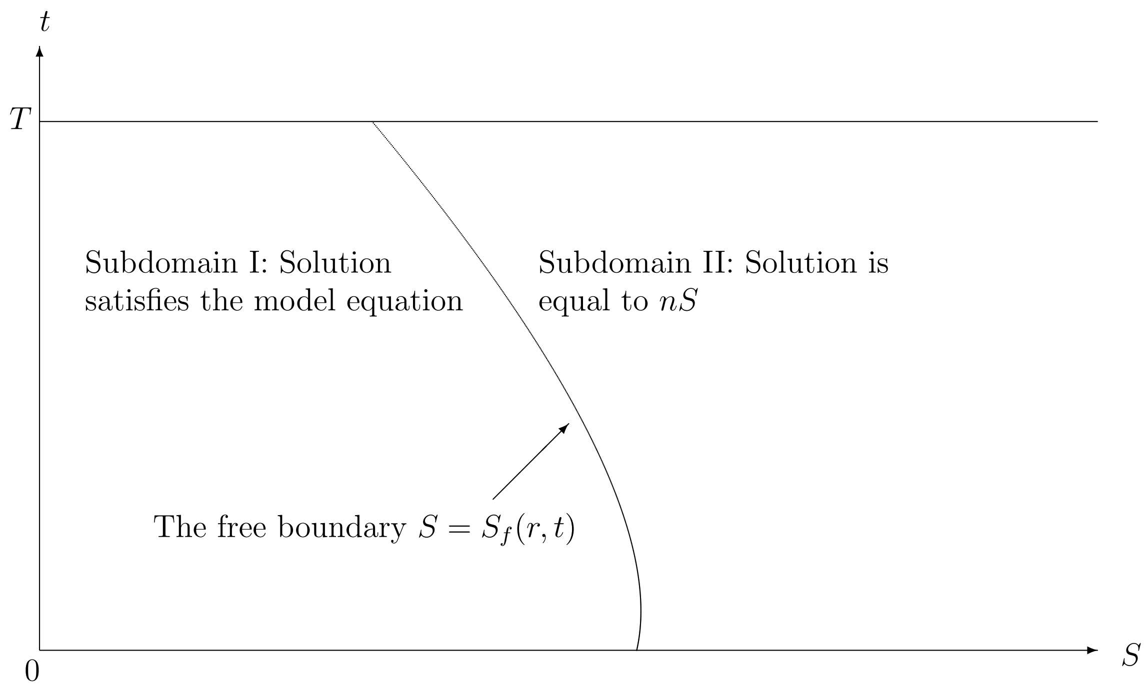

Figure 3.1: An American style derivative underlying by a dividend-stock usually has a free boundary. The starting location of this free boundary is $S_f(r,T) = \max(\frac{Z}{n}, \frac{kZ}{D_0 n})$. This free boundary divides the computational domain into two subdomains, the solution satisfies the PDE in subdomain I and the solution is equal to nS in subdomain II.

When $S > \dfrac{kZ}{D_0 n}$, the price is nS, and the equation (2.15) does not apply; when $S < \dfrac{kZ}{D_0 n}$, the price is determined by the equation (2.15).

3.3.2 The free boundary problem

When there is some dividend, say, $D_0 > 0$, the price of the convertible bond is the solution of the LC problem (3.4). Given the starting curve of the free boundary $S_f(r,T) = \max\left(\dfrac{Z}{n}, \dfrac{kZ}{D_0 n}\right)$, we will reformulate this LC problem to a FBP.

We assume that there is only one free boundary, and the computational domain $[0,\infty) \times [r_l, r_u] \times [0,T]$ has been divided into two subdomains:

$$\begin{aligned} \mathrm{I} &: [0, S_f(r,t)] \times [r_l, r_u] \times [0,T], \\ \mathrm{II} &: (S_f(r,t), \infty) \times [r_l, r_u] \times [0,T]. \end{aligned}$$

As we already know [14] that the solution in the subdomain II is

$$B_c(S,r,t) = nS, \tag{3.6}$$

which gives

$$\frac{\partial B_c}{\partial S} = n \text{ and } \frac{\partial B_c}{\partial r} = 0.$$

At the free boundary between those two subdomains, the solution and its first derivatives should be continuous, so there are three boundary conditions we need to require at the free boundary, one is for the solution and the other two are for its first derivatives:

$$B_c(S_f(r,t),r,t) = nS_f(r,t), \tag{3.7}$$

$$\frac{\partial B_c}{\partial S}(S_f(r,t),r,t) = n, \tag{3.8}$$

$$\frac{\partial B_c}{\partial r}(S_f(r,t),r,t) = 0. \tag{3.9}$$

Notice that first two equations (3.7) and (3.8) ensure that the third one (3.9) is true.

Thus boundary conditions at $S = S_f$ in subdomain I are

$$\begin{aligned} &B_c(S_f(r,t),r,t) = nS_f(r,t), \\ &\frac{\partial B_c}{\partial S}(S_f(r,t),r,t) = n. \end{aligned} \tag{3.10}$$

Thus in subdomain I $B_c(S,r,t)$ and $S_f(r,t)$ should be the solution of the problem:

$$\begin{cases} \dfrac{\partial B_c}{\partial t} + \mathbf{L}_{S,r} B_c + kZ = 0, \\ B_c(S,r,T) = \max(Z, nS), \\ B_c(S_r(r,t),r,t) = nS_f(r,t), \\ \dfrac{\partial}{\partial S} B_c(S_f(r,t),r,t) = n, \\ S_f(r,T) = \max\left(\dfrac{Z}{n}, \dfrac{kZ}{D_0 n}\right). \end{cases} \tag{3.11}$$

We will refer (3.11) the Free Boundary Problem for convertible bonds. If

$$D_0 \to 0,$$

then the starting point of the free boundary $S_f(r,T) \to \infty$, so there is no free boundary at time T if $D_0 = 0$. Also no new free boundary will appear for $t \in [0,T)$[14].

CHAPTER 4: SINGULARITY-SEPARATING METHOD

There is no analytic solution for the FBP (3.11), thus the solution must be carried out by some numerical methods. In the FBP, there exists some kind of singularity which makes the computation of highly accurate solution difficult if $t \approx T$.

There are several ways to overcome this difficulty, for example, if we increase the grid points near the singularity, this will enable us to obtain accurate results there. But this will need a greater memory requirement and slow down the computational process. A new technique called Singularity-Separating Method (SSM) is introduced to weaken the singularity.

Because the solution we need to find has a certain type of singularity caused by the final condition, we try to find an analytic expression satisfying the same final condition and a similar equation or the same equation. If both equations and final conditions are the same, then singularities for these two problems should be the same, and the difference between them should be a smooth function; if only the final condition is the same, then the singularity should be similar and the difference should be a smoother function than the solution in the original problem. If we could find such an analytic expression satisfying the final condition and a similar equation, we could solve the difference between these two and add this analytic expression back to the difference in order to get the solution of the original problem.

Of course, there is some extra work in computing the difference. However, from examples we are going to show, such a way can truly make numerical calculations more efficient. Our method will be used to the price model of the two-factor convertible bond when $D_0 \neq 0$. Indeed, the method can be used for other cases, including multi-factor derivative securities.

4.1 One-dimensional case: American vanilla call option problem

Before we apply SSM to the two-factor convertible bond problem, let us consider a one-dimensional American vanilla call option as an example, going through the details of SSM for the FBP.

4.1.1 Model problem

In $[0, S_f(t)] \times [0, T]$, the price of an American call option, $C(S,t)$, is the solution of the following FBP

$$\begin{cases} \dfrac{\partial C}{\partial t} + \dfrac{1}{2}\sigma^2 S^2 \dfrac{\partial^2 C}{\partial S^2} + (r - D_0)S\frac{\partial C}{\partial S} - rC = 0, & 0 \le S \le S_f(t), \quad 0 \le t \le T, \\ C(S,T) = \max(S - E, 0), & 0 \le S \le S_f(T), \\ C(S_f(t), t) = S_f(t) - E, & 0 \le t \le T, \\ \dfrac{\partial C}{\partial S}(S_f(t), t) = 1, & 0 \le t \le T, \\ S_f(T) = \max\left(E, \dfrac{rE}{D_0}\right); \end{cases} \tag{4.1}$$

whereas in $(S_f(t), \infty) \times [0, T]$, $C(S,t) = S - E$.

Here, we assume $D_0 \neq 0$. Therefore, as long as we have the solution of FBP we can determine $C(S,t)$ for any $S \geq 0$ and any $t \in [0,T]$. The function $C(S,T) = \max(S - E, 0)$ has a discontinuous derivative at $S = E$. Therefore, $C(S,t)$ is not very smooth in the region where $S \approx E$ and $t \approx T$.

Because the second derivative of $C(S,t)$ at $S = E$ goes to infinity, the truncation error of numerical methods near $S = E$ and $t = T$ is relatively large. To avoid such a relatively large error, we first find that numerical result of the difference between the prices of the American call option and the European call option, and then add the difference to the price of the European call option to get the price of the American call option.

4.1.2 An analytic expression

Let $c(S,t)$ represent the price of the European call option, and it is the solution of the problem

$$\begin{cases} \dfrac{\partial c}{\partial t}+\dfrac{1}{2}\sigma^2 S^2\dfrac{\partial^2 c}{\partial S^2}+(r-D_0)S\dfrac{\partial c}{\partial S}-rc=0, & 0\le S, \quad 0\le t\le T, \\ c(S,T)=\max(S-E,0), & 0\le S. \end{cases} \tag{4.2}$$

$c(S,t)$ has an analytic form of

$$c(S,t)=Se^{-D_0(T-t)}N(d_1)-Ee^{-r(T-t)}N(d_2), \tag{4.3}$$

where

$$d_1=\left[\ln\frac{Se^{(r-D_0)(T-t)}}{E}+\frac{1}{2}\sigma^2(T-t)\right]/(\sigma\sqrt{T-t}) \tag{4.4}$$

and

$$d_2=\left[\ln\frac{Se^{(r-D_0)(T-t)}}{E}-\frac{1}{2}\sigma^2(T-t)\right]/(\sigma\sqrt{T-t}). \tag{4.5}$$

4.1.3 American vanilla call option problem

Define

$$\bar{C}(S,t)=C(S,t)-c(S,t)$$

in the subdomain $[0,S_f(t)]\times[0,T]$. Since $C(S,T)=c(S,T)=\max(S-E,0)$, $\bar{C}(S,T)=0$ in the subdomain $(S_f(t),\infty)\times[0,T]$. The functions $C(S,t)$ and $c(S,t)$ satisfy the same linear homogeneous partial differential equation, the difference between them does the same. At the free boundary $S=S_f(t)$, we have

$$\bar{C}(S_f(t),t)=C(S_f(t),t)-c(S_f(t),t)=S_f(t)-E-c(S_f(t),t)$$

and

$$\frac{\partial\bar{C}}{\partial S}(S_f(t),t)=\frac{\partial C}{\partial S}(S_f(t),t)-\frac{\partial c}{\partial S}(S_f(t),t)=1-\frac{\partial c}{\partial S}(S_f(t),t).$$

Thus, $\bar{C}(S,t)$ is the solution of the FBP

$$\begin{cases} \dfrac{\partial \bar{C}}{\partial t} + \dfrac{1}{2}\sigma^2 S^2 \dfrac{\partial^2 \bar{C}}{\partial S^2} + (r - D_0) S \dfrac{\partial \bar{C}}{\partial S} - r\bar{C} = 0, & 0 \le S \le S_f(t), \quad 0 \le t \le T, \\ \bar{C}(S,T) = 0, & 0 \le S \le S_f(T), \\ \bar{C}(S_f(t),t) = S_f(t) - E - c(S_f(t),t), & 0 \le t \le T, \\ \dfrac{\partial \bar{C}}{\partial S}(S_f(t),t) = 1 - \dfrac{\partial c}{\partial S}(S_f(t),t), & 0 \le t \le T, \\ S_f(T) = \max\left(E, \dfrac{rE}{D_0}\right). \end{cases} \tag{4.6}$$

In summary, the solution of the original American call option satisfies different equations in two subdomains divided by the free boundary $S = S_f(t)$, and its solution has a discontinuous second derivative - a type of weak singularity - on the free boundary. In this method, the location of the free boundary is tracked accurately, so we can use the different equations in each subdomain exactly. Because the solution in $(S_f(t), \infty) \times [0,T]$ is given by a known function, we only need to determine the solution in $[0, S_f(t)] \times [0,T]$. In this subdomain, the second derivative near the free boundary is continuous, so the solution we want to get numerically is smoother than the original solution. Here, we also suggest to compute the difference between the American call option and the European call option numerically in $[0, S_f(t)] \times [0,T]$, instead of directly computing the price of the American call options numerically. The difference is very smooth in the domain $[0, S_f(t)] \times [0,T]$, which makes the truncation error smaller. Therefore, in the method described above, we use some techniques such that the solution is much smoother than the original solution numerically, and this makes numerical methods more efficient. We refer this as Singularity-Separating because the solution becomes smoother than the original one after some singularities have been 'separated'. Here, the singularity that has been 'separated' is the discontinuity of the derivatives of the solution, which is weak.

The same idea still works for American barrier, Asian, and lookback options. The difference will be a solution of a nonhomogeneous partial differential equation problem with a weaker singularity.

4.2 Two-Dimensional case: a two-factor convertible bond model

In the previous section (4.1.2), we have discussed the formulation of the American call option, now let us take a look at the case of the two-factor convertible bond problem.

4.2.1 An analytic expression

Consider the following problem:

$$\begin{cases} \dfrac{\partial b_c}{\partial t} + \dfrac{1}{2}\sigma^2 S^2 \dfrac{\partial^2 b_c}{\partial S^2} + (r - D_0) S \dfrac{\partial b_c}{\partial S} - r b_c + +kZ = 0, \quad 0 \le S, \quad 0 \le t \le T, \\ b_c(S,T) = \max(Z, nS), \quad 0 \le S. \end{cases} \tag{4.7}$$

where σ, r, D_0, Z, n and are $k(t)$ constants. This problem has the following analytic solution $b_c(S,r,t)$:

$$b_c(S,r,t) = nc(S,t;Z/n) + Ze^{-r(T-t)} \left[1 + \int_t^T k e^{r(T-t)} dt \right], \tag{4.8}$$

where $c(S,t;Z/n)$ is the price of a European call option with an exercise price $E = \dfrac{Z}{n}$. As we know, the expression for $c(S,t;E)$ is

$$c(S,t;E) = Se^{-D_0(T-t)} N(d_1) - Ee^{-r(T-t)} N(d_2),$$

where

$$d_1 = \left[\ln \frac{Se^{(r-D_0)(T-t)}}{E} + \frac{1}{2}\sigma^2 (T-t) \right] / (\sigma\sqrt{T-t}) \tag{4.9}$$

and

$$d_2 = \left[\ln \frac{Se^{(r-D_0)(T-t)}}{E} - \frac{1}{2}\sigma^2 (T-t) \right] / (\sigma\sqrt{T-t}). \tag{4.10}$$

4.2.2 The two-factor convertible bond problem

Define the difference between $B_c(S,r,t)$ and $b_c(S,r,t)$ as $\bar{B}_c(S,r,t)$:

$$\bar{B}_c(S,r,t) \equiv B_c(S,r,t) - b_c(S,r,t),$$

then the problem for $\bar{B}_c(S,r,t)$ is

$$\begin{cases} \dfrac{\partial \bar{B}_c}{\partial t} + \mathbf{L}_{S,r}\bar{B}_c = f(S,r,t), \\ \bar{B}_c(S,r,T) = 0, \\ \bar{B}_c(S_r(r,t),r,t) = nS_f(r,t) - b_c(S_f(r,t),r,t), \\ \dfrac{\partial}{\partial S}\bar{B}_c(S_f(r,t),r,t) = n - \dfrac{\partial}{\partial S}b_c(S_f(r,t),r,t), \\ S_f(r,T) = \max\left(\dfrac{Z}{n}, \dfrac{kZ}{D_0 n}\right), \end{cases} \tag{4.11}$$

where $\mathbf{L}_{S,r}$ is given in (2.16) and

$$\begin{aligned} & f(S,r,t) \\ = \; & -\left(\rho\sigma S w \frac{\partial^2 b_c}{\partial S \partial r} + \frac{1}{2}w^2 \frac{\partial^2 b_c}{\partial r^2} + (u - \lambda w)\frac{\partial b_c}{\partial r}\right) \\ = \; & -Z\left\{\sqrt{\frac{T-t}{2\pi}}\rho w e^{-r(T-t)-\frac{d_2^2}{2}} + \frac{1}{2}w^2\left[(T-t)^2 e^{-r(T-t)}(1-N(d_2))\right.\right. \\ & \left.\left. + \frac{1}{\sqrt{2\pi}\sigma}(T-t)^{3/2}e^{-r(T-t)-\frac{d_2^2}{2}}\right] + (u-\lambda w)(T-t)e^{-r(T-t)}[N(d_2)-1]\right\}, \\ & d_2 \text{ being } \left[\ln\frac{Se^{(r-D_0)(T-t)}}{Z/n} - \frac{1}{2}\sigma^2(T-t)\right]/(\sigma\sqrt{T-t}). \end{aligned}$$

It is obvious that if we get the solution of $\bar{B}_c(S_f(r,t),r,t)$, it is easy to obtain the solution to (3.11) by

$$B_c(S,r,t) = \bar{B}_c(S,r,t) + b_c(S,r,t),$$

where $b_c(S,r,t)$ has an analytic expression given in (4.8). This formulation (4.11) will be used only at $t \approx T$, since only in this case this formulation has the advantage.

CHAPTER 5: DETERMINATION OF u, w AND λ

Before we continue to discuss the numerical solution of the model (4.11), we take a look at the determination of u, w and the market price of risk λ. We start with determining u and w.

5.1 u and w

In the past years, many interest models had been proposed by various researchers by giving different forms of functions u,w.

5.1.1 Wilmott's model

$$w(r,t) = \sqrt{\alpha(t)r - \beta(t)}, \tag{5.1}$$

$$u(r,t) = -\gamma(t)r + \delta(t) + \lambda(r,t)w(r,t) \tag{5.2}$$

This interest rate model is suitable for solving partial differential equation with random interest rate.

This model have the following properties: The interest rate can be made mean reverting; The interest rate is bounded below by $\frac{\beta(t)}{\alpha(t)}$ so negative interest rates can be avoided. The model can be made to fit the current term structure. By adding $\lambda(r,t)w(r,t)$ into the drift, the market price of risk λ is eliminated from the convertible bond equation (2.15), (3.11) and (4.11).

The deviation is in the square root form so that the solution of the bond equation has a simple form. As being pointed out by Wilmott, existence of a simple solution is not a good reason to accept a model. He justified this by saying that the nature of fitting the current term structure makes this model useful.

Unfortunately, if we fit the parameters to match today's term structure, the simple form solution usually can not be obtained explicitly because of the complex form of the parameters. This property is reasonable and it has been kept in Sun's model.

5.1.2 Sun's model

Sun proposed an interest rate model in the sense that the interest rate is allowed to go to infinity for all interest rate models, but this is not likely to happen in a normal economy. There should be an upper bound for the interest rate. Therefore Sun proposed the deviation has the general form:

$$w(r,t) = \omega(r,t)\phi(r; r_l, r_u), \tag{5.3}$$

where $\phi(r; r_l, r_u)$ is a smooth function that is vanished at $r = r_l$ and $r = r_u$, and is close to 1 for $r \in (r_l, r_u)$. An sample ϕ is

$$\phi(r; r_l, r_u) = \left(\frac{4(r-r_l)(r_u-r)}{(r_u-r_l)^2}\right)^{\frac{1}{4}} \tag{5.4}$$

5.1.3 Our model

Let $\lambda = -u/w + \bar{\lambda}$. In this case $u - \lambda w = \bar{\lambda} w$. Thus we can always choose a form of λ so that u disappear from the PDE. Therefore we can choose

$$u = 0$$

and what is left is to find a reasonable and market data-oriented $w(r)$.

We assume that w is a function of r in the following form:

$$w(r) = (r-r_l)(r_u-r)(a_0 r^2 + b_0 r + c_0). \tag{5.5}$$

Clearly, w satisfies conditions (2.13) and (2.14). What we need to do is to determine the coefficients a_0, b_0 and c_0 from the market data and the method is as follows.

We denote the maximum and minimum interest rate in the data set as $r_{\max}$, $r_{\min}$

and we divide the interval $[r_{\min}, r_{\max}]$ into Q subintervals:

$$r_{(q)} = [r_{[q-1]}, r_{[q]}], \quad q = 1, \cdots, Q,$$

where

$$r_{[q]} = r_{\min} + q\frac{r_{\max} - r_{\min}}{Q}, \quad q = 0, 1, \cdots, Q.$$

Suppose that there are n_q interest rates $r_{q,n}, n = 1, \cdots, n_q$ in the q-th subinterval and that $r_{q,n}$ is the interest rate at time $t_{q,n}$ and at time $t_{q,n} + dt$ the interest rate is $r'_{q,n}$. In each subinterval, we can get an approximate variance of the random variable r at the midpoint of the subinterval by

$$w_q^2 = \frac{1}{(n_q - 1)dt}\left\{\sum_{n=1}^{n_q}(r'_{q,n} - r_{q,n})^2 - \frac{1}{n_q}\left[\sum_{n=1}^{n_q}(r'_{q,n} - r_{q,n})\right]^2\right\}.$$

From the Q values of w_q^2 in $[r_{\min}, r_{\max}]$, we are able to determine a_0, b_0 and c_0 in (5.5) by the least squares method with weights. The weight for the q-th interval is $n_q / \sum_{q=1}^{Q} n_q$. In this way $w(r)$ can be constructed from the historical data.

5.2 The market price of risk λ

Consider the interest rate model (2.11). For any interest rate derivative dependent on r and t, for example, a zero-coupon bond, its value $V(r,t)$ satisfies the following partial differential equation [14]:

$$\frac{\partial V}{\partial t} + \frac{1}{2}w^2\frac{\partial^2 V}{\partial r^2} + (u - \lambda w)\frac{\partial V}{\partial r} - rV + kZ = 0. \tag{5.6}$$

Like we discussed before, there is no need for boundary conditions at $r = r_l$ and $r = r_u$ if (2.13) and (2.14) hold[14], thus there is a unique solution for the final value problem:

$$\begin{cases} \dfrac{\partial V}{\partial t} + \dfrac{1}{2}w^2\dfrac{\partial^2 V}{\partial r^2} + (u - \lambda w)\dfrac{\partial V}{\partial r} - rV + kZ = 0, \\ V(r, T^*) = f(r), \quad t \le T, \quad r_l \le r \le r_u. \end{cases} \tag{5.7}$$

Once we have achieved u and w, in order to price an interest rate derivative, what needs to be done is to find λ.

Let $V(r,t;T^*)$ be the value of a zero-coupon bond with maturity T^* and face value 1. Then it is the solution of the following final value problem:

$$\begin{cases} \dfrac{\partial V}{\partial t} + \dfrac{1}{2}w^2\dfrac{\partial^2 V}{\partial r^2} + (u - \lambda w)\dfrac{\partial V}{\partial r} - rV = 0, \\ V(r,T^*;T^*) = 1, \quad t \le T^*, \quad r_l \le r \le r_u. \end{cases} \tag{5.8}$$

Let $T_{\max}$ be the longest maturity for the bonds in the market and assume that $t = 0$ means today and today's spot interest rate is r^*. If we have a function $\lambda(r,t)$ and for any $T^* \in [0, T_{\max}]$, the value $V(r^*, 0; T^*)$ obtained from (5.8) matches with the value of the zero-coupon bond with maturity T^* on the market, then (5.7) is a very good model for interest rate derivatives. The problem is how to find such a function $\lambda(r,t)$. If we assume that λ depends on t only, then $\lambda(t)$ can be determined in the following way.

5.2.1 Interpolation

On the market the historical data of 0.25-year, 0.5-year, 1-year, 2-year, 3-year, 5-year, 10-year, 15-year, 20-year and 30-year bond prices are available. However we could not get the historical data for a 4-year bond price. In order to obtain the bond price with any maturity, we construct a function defined on $[0, T_{\max}]$ by the cubic spline interpolation based on the data available. We denote the function by $\bar{V}(T^*)$.

5.2.2 Initial value of $\lambda(t)$

Although we do not know the market price of the risk so far, but it has been shown [14] that its value at $t = 0$ is given by

$$\lambda(0) = \frac{\frac{\partial^2 \bar{V}(0)}{\partial T^{*2}} - r^{*2} + u(r^*, 0)}{w(r^*, 0)}. \tag{5.9}$$

5.2.3 Solve (5.8) at different maturities

Let $\Delta t = T_{\max}/K, K$ being a big integer and let $T_k^* = k\Delta t$. Assume that we have obtained $\lambda(t)$ in $[0, T_k^* - \Delta t]$ from $\bar{V}(T^*)$ in $[0, T_k^* - \Delta t]$. We can guess $\lambda(T_k^*)$ and define $\lambda(t)$ on $[T_k^* - \Delta t, T_k^*]$. Because $\lambda(t)$ is given on $[0, T_k^*]$, we could solve (5.8) and get $V(r^*, 0; T_k^*)$.

5.2.4 Comparison of two bond prices

As soon as we find $V(r^*, 0; T_k^*)$, we can check whether this value is equal to $\bar{V}(T_k^*)$. If they are close to each other, this means we find $\lambda(T_k^*)$; if they are not, another different $\lambda(T_k^*)$ has to be guessed and we do the process again until we find $\lambda(T_k^*)$.

5.2.5 Get $\lambda(t)$

Perform the procedure above for $T_k^*, k = 1, 2, \cdots, K$ successively, we can obtain the function $\lambda(t)$ in $[0, T_{\max}]$.

In this procedure, the coefficient $\lambda(t)$ in a PDE is determined by some information on solutions of the PDE problem. Such a problem usually is called an inverse problem. To find such a $\lambda(t)$ usually is called to solve an inverse problem.

CHAPTER 6: NUMERICAL METHODS

In this chapter we deal with the formulation (3.4) and the formulation (4.11) because the formulation (3.11) is similar to and simpler than the formulation (4.11) so that we will know how to deal with (3.11) if we know how to deal with (4.11).

To solve the model problem (4.11), we start with a transformation which makes it easier to do the discretization.

6.1 A linear transformation

Consider the following transformation:

$$\begin{cases} \xi = \dfrac{S}{S_f(r,t)}, \\ \bar{r} = \dfrac{r - r_l}{r_u - r_l}, \\ \tau = T - t. \end{cases} \tag{6.1}$$

This transformation maps the domain $[0, S_f(r,t)] \times [r_l, r_u] \times [0, T]$ in the (S, r, t)-space into the domain $[0,1] \times [0,1] \times [0,T]$ in the $(\xi, \bar{r}, \tau)$-space. Three important functions, the value of the convertible bond $\bar{B}_c(S,r,t)$, the location of the free boundary $S_f(r,t)$ and the analytic expression $b_c(S,r,t)$ are also transformed in the following way:

$$\begin{cases} U(\xi, \bar{r}, \tau) = \dfrac{\bar{B}_c(S,r,t)}{Z}, \\ s_f(\bar{r}, \tau) = \dfrac{S_f(r,t)}{Z/n}, \\ v(\xi, \bar{r}, \tau) = \dfrac{b_c(S,r,t)}{Z}. \end{cases} \tag{6.2}$$

Substituting (4.8) into the third equation in (6.2) yields

$$
\begin{aligned}
v(\xi,\bar{r},\tau) &= nc(S,t;Z/n)/Z + e^{-r(T-t)} \\
&= \frac{nS}{Z}e^{-D_0(T-t)}N(d_1) - e^{-r(T-t)}N(d_2) + e^{-r(T-t)} \\
&= \xi s_f(\bar{r},\tau)e^{-D_0\tau}N(d_1) + e^{-r\tau}N(-d_2),
\end{aligned} \tag{6.3}
$$

where d_1 and d_2 are given by (4.9) and (4.10), and they can be rewritten as

$$
\begin{aligned}
d_1 &= [\ln \tfrac{Se^{(r-D_0)(T-t)}}{Z/n} + \tfrac{1}{2}\sigma^2(T-t)]/(\sigma\sqrt{T-t}) \\
&= [\ln(\xi s_f(\bar{r},\tau)e^{[r_l+\bar{r}(r_u-r_l)-D_0]\tau}) + \tfrac{1}{2}\sigma^2\tau]/(\sigma\sqrt{\tau}), \\
d_2 &= [\ln(\xi s_f(\bar{r},\tau)e^{[r_l+\bar{r}(r_u-r_l)-D_0]\tau}) - \tfrac{1}{2}\sigma^2\tau]/(\sigma\sqrt{\tau})
\end{aligned} \tag{6.4}
$$

if the new variables ξ and $\bar{r}$ are used. Because $\bar{B}_c(S,r,t) = ZU(\xi,\bar{r},\tau)$, we have

$$
\left\{
\begin{aligned}
\frac{\partial \bar{B}_c}{\partial t} &= Z\left(-\frac{\partial U}{\partial \tau} + \frac{\partial U}{\partial \xi}\frac{\xi}{s_f}\frac{\partial s_f}{\partial \tau}\right), \\
\frac{\partial \bar{B}_c}{\partial S} &= \frac{\partial U}{\partial \xi}\frac{n}{s_f}, \\
\frac{\partial \bar{B}_c}{\partial r} &= Z\left(-\frac{\partial U}{\partial \xi}\frac{\xi}{s_f}\frac{\partial s_f}{\partial \bar{r}} + \frac{\partial U}{\partial \bar{r}}\right)\frac{1}{r_u-r_l}, \\
\frac{\partial^2 \bar{B}_c}{\partial S^2} &= \frac{1}{Z}\frac{\partial^2 U}{\partial \xi^2}\left(\frac{n}{s_f}\right)^2, \\
\frac{\partial^2 \bar{B}_c}{\partial S\partial r} &= \left(-\frac{\partial^2 U}{\partial \xi^2}\frac{n\xi}{s_f^2}\frac{\partial s_f}{\partial \bar{r}} + \frac{\partial^2 U}{\partial \xi\partial \bar{r}}\frac{n}{s_f} - \frac{\partial U}{\partial \xi}\frac{n}{s_f^2}\frac{\partial s_f}{\partial \bar{r}}\right)\frac{1}{r_u-r_l}, \\
\frac{\partial^2 \bar{B}_c}{\partial r^2} &= Z\left\{\frac{\partial^2 U}{\partial \xi^2}\left(\frac{\xi}{s_f}\frac{\partial s_f}{\partial \bar{r}}\right)^2 - 2\frac{\partial^2 U}{\partial \xi\partial \bar{r}}\frac{\xi}{s_f}\frac{\partial s_f}{\partial \bar{r}}\right. \\
&\quad \left. +\frac{\partial U}{\partial \xi}\left[2\frac{\xi}{s_f^2}\left(\frac{\partial s_f}{\partial \bar{r}}\right)^2 - \frac{\xi}{s_f}\frac{\partial^2 s_f}{\partial \bar{r}^2}\right] + \frac{\partial^2 U}{\partial \bar{r}^2}\right\}\left(\frac{1}{r_u-r_l}\right)^2.
\end{aligned}
\right. \tag{6.5}
$$

Substituting these relations into (4.11), we have the following free-boundary problem in the domain $[0,1]\times[0,1]\times[0,T]$ in $(\xi,\bar{r},\tau)$-space:

$$\begin{cases} \dfrac{\partial U}{\partial \tau} = \mathbf{L}_{\xi,\bar{r}}U + a_7, \\ U(\xi,\bar{r},0) = 0, \\ U(1,\bar{r},\tau) = s_f(\bar{r},\tau) - v(1,\bar{r},\tau), \\ \dfrac{\partial U}{\partial \xi}(1,\bar{r},\tau) = s_f(\bar{r},\tau) - \dfrac{\partial v}{\partial \xi}(1,\bar{r},\tau), \\ s_f(\bar{r},0) = \max(1, k/D_0). \end{cases} \tag{6.6}$$

Here the operator $\mathbf{L}_{\xi,\bar{r}}$ is defined as follows:

$$\mathbf{L}_{\xi,\bar{r}} = a_1\xi^2\frac{\partial^2}{\partial\xi^2} + a_2\xi w\frac{\partial^2}{\partial\xi\partial\bar{r}} + a_3\frac{\partial^2}{\partial\bar{r}^2} + (a_4 + \frac{1}{s_f}\frac{\partial s_f}{\partial\tau})\xi\frac{\partial}{\partial\xi} + a_5\frac{\partial}{\partial\bar{r}} + a_6$$

and the expressions for a_i, $i=1,\cdots,7$ are

$$\begin{aligned}
a_1 &= \tfrac{1}{2}\sigma^2 - \rho\sigma w\tfrac{1}{s_f(r_u-r_l)}\tfrac{\partial s_f}{\partial\bar{r}} + \tfrac{1}{2}w^2[\tfrac{1}{s_f(r_u-r_l)}\tfrac{\partial s_f}{\partial\bar{r}}]^2, \\
a_2 &= \tfrac{1}{r_u-r_l}[\rho\sigma - \tfrac{w}{s_f(r_u-r_l)}\tfrac{\partial s_f}{\partial\bar{r}}], \\
a_3 &= \tfrac{1}{2(r_u-r_l)^2}w^2, \\
a_4 &= r - D_0 - \tfrac{1}{s_f(r_u-r_l)}\tfrac{\partial s_f}{\partial\bar{r}}(\rho\sigma w + u - \lambda w) \\
&\quad + \tfrac{1}{2}w^2\left\{2\left[\tfrac{1}{s_f(r_u-r_l)}\tfrac{\partial s_f}{\partial\bar{r}}\right]^2 - \tfrac{1}{s_f(r_u-r_l)^2}\tfrac{\partial^2 s_f}{\partial\bar{r}^2}\right\}, \\
a_5 &= \tfrac{u-\lambda w}{r_u-r_l}, \\
a_6 &= -r, \\
a_7 &= k + \rho\sigma Sw\tfrac{\partial^2 v}{\partial S\partial r} + \tfrac{1}{2}w^2\tfrac{\partial^2 v}{\partial r^2} + (u-\lambda w)\tfrac{\partial v}{\partial r} \\
&= k + \sqrt{\tfrac{T-t}{2\pi}}\rho w e^{-r(T-t)-\frac{d_2^2}{2}} + \tfrac{1}{2}w^2\Bigg[(T-t)^2e^{-r(T-t)}(1-N(d_2)) \\
&\quad + \sqrt{\tfrac{T-t}{2\pi}}\tfrac{T-t}{\sigma}e^{-r(T-t)-\frac{d_2^2}{2}}\Bigg] + (u-\lambda w)(T-t)e^{-r(T-t)}[N(d_2)-1].
\end{aligned}$$

6.2 Discretization

In this section, we will discretize the formulation (6.6). Assume that

$$\begin{aligned}
\xi_m &= m\Delta\xi, & m &= 0,\cdots,M,\\
\bar{r}_i &= i\Delta\bar{r}, & i &= 0,\cdots,I,\\
\tau^n &= n\Delta\tau, & n &= 0,\cdots,N,
\end{aligned}$$

where M, N, I are given integers, $\Delta\xi = 1/M$, $\Delta\bar{r} = 1/I$ and $\Delta\tau = T/N$. Let $U^n_{m,i}$ denote the approximate value of $U(\xi_m, \bar{r}_i, \tau^n)$.

6.2.1 Interior points

For each interior point, the PDE in (6.6) can be discretized at the point $(\xi_m, \bar{r}_i, \tau^{n+1/2})$ as

$$\begin{aligned}
\tfrac{U^{n+1}_{m,i}-U^n_{m,i}}{\Delta\tau} = {} & \tfrac{\tilde{a}_1 m^2}{2}(U^{n+1}_{m+1,i} - 2U^{n+1}_{m,i} + U^{n+1}_{m-1,i} + U^n_{m+1,i} - 2U^n_{m,i} + U^n_{m-1,i})\\
& + \tfrac{\tilde{a}_2 wm}{8\Delta\bar{r}}(U^{n+1}_{m+1,i+1} - U^{n+1}_{m+1,i-1} - U^{n+1}_{m-1,i+1} + U^{n+1}_{m-1,i-1}\\
& + U^n_{m+1,i+1} - U^n_{m+1,i-1} - U^n_{m-1,i+1} + U^n_{m-1,i-1})\\
& + \tfrac{\tilde{a}_3}{2\Delta\bar{r}^2}(U^{n+1}_{m,i+1} - 2U^{n+1}_{m,i} + U^{n+1}_{m,i-1} + U^n_{m,i+1} - 2U^n_{m,i} + U^n_{m,i-1})\\
& + \tfrac{m}{4}\left(\tilde{a}_4 + \tfrac{2(s^{n+1}_{f,i} - s^n_{f,i})}{\Delta\tau}\tfrac{1}{s^{n+1}_{f,i} + s^n_{f,i}}\right)\\
& \qquad \cdot(U^{n+1}_{m+1,i} - U^{n+1}_{m-1,i} + U^n_{m+1,i} - U^n_{m-1,i})\\
& + \tfrac{\tilde{a}_5}{4\Delta\bar{r}}(U^{n+1}_{m,i+1} - U^{n+1}_{m,i-1} + U^n_{m,i+1} - U^n_{m,i-1})\\
& + \tfrac{\tilde{a}_6}{2}(U^{n+1}_{m,i} + U^n_{m,i})\\
& + \tilde{a}_7,\\
& \quad m = 0,\cdots,M-1, \quad i = 1,\cdots,I-1.
\end{aligned} \tag{6.7}$$

In order to achieve a second-order accuracy, $\tilde{a}_1, \cdots, \tilde{a}_7$ should take the value at the point $(\xi_m, \bar{r}_i, \tau^{n+1/2})$, i.e., their expressions are:

$$
\begin{aligned}
\tilde{a}_1 \equiv a_{1,m,i}^{n+\frac{1}{2}} &= \tfrac{1}{2}(\sigma_{m,i}^{n+\frac{1}{2}})^2 - \frac{\rho_{m,i}^{n+\frac{1}{2}}\sigma_{m,i}^{n+\frac{1}{2}}w_{m,i}^{n+\frac{1}{2}}}{2\Delta\bar{r}(r_u-r_l)}\frac{s_{f,i+1}^{n+1}-s_{f,i-1}^{n+1}+s_{f,i+1}^{n}-s_{f,i-1}^{n}}{s_{f,i}^{n+1}+s_{f,i}^{n}} \\
&\quad +\tfrac{1}{8}\left(\frac{w_{m,i}^{n+\frac{1}{2}}}{\Delta\bar{r}(r_u-r_l)}\right)^2\left(\frac{s_{f,i+1}^{n+1}-s_{f,i-1}^{n+1}+s_{f,i+1}^{n}-s_{f,i-1}^{n}}{s_{f,i}^{n+1}+s_{f,i}^{n}}\right)^2, \\
\tilde{a}_2 \equiv a_{2,m,i}^{n+\frac{1}{2}} &= \frac{1}{r_u-r_l}\rho_{m,i}^{n+\frac{1}{2}}\sigma_{m,i}^{n+\frac{1}{2}} - \frac{w_{m,i}^{n+\frac{1}{2}}}{2(r_u-r_l)^2\Delta\bar{r}}\frac{s_{f,i+1}^{n+1}-s_{f,i-1}^{n+1}+s_{f,i+1}^{n}-s_{f,i-1}^{n}}{s_{f,i}^{n+1}+s_{f,i}^{n}}, \\
\tilde{a}_3 \equiv a_{3,m,i}^{n+\frac{1}{2}} &= \frac{1}{2(r_u-r_l)^2}w_{m,i}^{n+\frac{1}{2}}, \\
\tilde{a}_4 \equiv a_{4,m,i}^{n+\frac{1}{2}} &= (r_l + i\Delta\bar{r}(r_u-r_l) - D_0) \\
&\quad -\frac{\rho_{m,i}^{n+\frac{1}{2}}\sigma_{m,i}^{n+\frac{1}{2}}w_{m,i}^{n+\frac{1}{2}}+u_{m,i}^{n+\frac{1}{2}}-\lambda_{m,i}^{n+\frac{1}{2}}w_{m,i}^{n+\frac{1}{2}}}{2\Delta\bar{r}(r_u-r_l)}\frac{s_{f,i+1}^{n+1}-s_{f,i-1}^{n+1}+s_{f,i+1}^{n}-s_{f,i-1}^{n}}{s_{f,i}^{n+1}+s_{f,i}^{n}} \\
&\quad +\left(\frac{w_{m,i}^{n+\frac{1}{2}}}{2\Delta\bar{r}(r_u-r_l)}\frac{s_{f,i+1}^{n+1}-s_{f,i-1}^{n+1}+s_{f,i+1}^{n}-s_{f,i-1}^{n}}{s_{f,i}^{n+1}+s_{f,i}^{n}}\right)^2 \\
&\quad -\tfrac{1}{2}\left(\frac{w_{m,i}^{n+\frac{1}{2}}}{\Delta\bar{r}(r_u-r_l)}\right)^2\frac{s_{f,i+1}^{n+1}-2s_{f,i}^{n+1}+s_{f,i-1}^{n+1}+s_{f,i+1}^{n}-2s_{f,i}^{n}+s_{f,i-1}^{n}}{s_{f,i}^{n+1}+s_{f,i}^{n}}, \\
\tilde{a}_5 \equiv a_{5,m,i}^{n+\frac{1}{2}} &= \frac{u_{m,i}^{n+\frac{1}{2}}-\lambda_{m,i}^{n+\frac{1}{2}}w_{m,i}^{n+\frac{1}{2}}}{r_u-r_l}, \\
\tilde{a}_6 \equiv a_{6,m,i}^{n+\frac{1}{2}} &= -[r_l + i\Delta\bar{r}(r_u-r_l)], \\
\tilde{a}_7 \equiv a_{7,m,i}^{n+\frac{1}{2}} &= k - \tfrac{1}{Z}f\left(\frac{Zm\Delta\xi(s_{f,i}^{n+1}+s_{f,i}^{n})}{2n}, r_l + i\Delta\bar{r}(r_u-r_l), (n+\tfrac{1}{2})\Delta\tau\right).
\end{aligned}
\tag{6.8}
$$

At $\xi = 0$, $U_{m-1,i-1}$, $U_{m-1,i}$, $U_{m-1,i+1}$ actually does not appear in (6.7) because their coefficients are zero due to $m = 0$. Thus the discretization method used for interior points can also be used at the points with $\xi = 0$ and $\bar{r} \neq 0, 1$. Thus in (6.7) we also put $m = 0$. However special attentions need to be paid to the boundary points with $\bar{r} = 0, 1$ and $\xi \in [0, 1)$, and the points with $\xi = 1$ and $\bar{r} \in [0, 1]$.

6.2.2 $\bar{r} = 0$ and $\bar{r} = 1$ with $\xi \in [0, 1)$

At the boundaries $\bar{r} = 0$ and $\bar{r} = 1$, due to $w = 0$, the partial differential equations in (6.6) becomes

$$
\frac{\partial U}{\partial \tau} = a_1\xi^2\frac{\partial^2 U}{\partial \xi^2} + \left(a_4 + \frac{1}{s_f}\frac{\partial s_f}{\partial \tau}\right)\xi\frac{\partial U}{\partial \xi} + a_5\frac{\partial U}{\partial \bar{r}} + a_6 U + a_7. \tag{6.9}
$$

Thus we can approximate the partial differential equation in (6.6) at the boundaries $\bar{r}=0$ and $\bar{r}=1$ by

$$\begin{aligned}
\frac{U^{n+1}_{m,0}-U^n_{m,0}}{\Delta\tau} &= \tfrac{\tilde{a}_1 m^2}{2}(U^{n+1}_{m+1,0}-2U^{n+1}_{m,0}+U^{n+1}_{m-1,0}+U^n_{m+1,0}-2U^n_{m,0}+U^n_{m-1,0})\\
&+\tfrac{m}{4}(\tilde{a}_4+\tfrac{2}{\Delta\tau}\tfrac{s^{n+1}_{f,0}-s^n_{f,0}}{s^{n+1}_{f,0}+s^n_{f,0}})(U^{n+1}_{m+1,0}-U^{n+1}_{m-1,0}+U^n_{m+1,0}-U^n_{m-1,0})\\
&+\tfrac{\tilde{a}_5}{4\Delta\bar{r}}(-U^{n+1}_{m,2}+4U^{n+1}_{m,1}-3U^{n+1}_{m,0}-U^n_{m,2}+4U^n_{m,1}-3U^n_{m,0})\\
&+\tfrac{\tilde{a}_6}{2}(U^{n+1}_{m,0}+U^n_{m,0})\\
&+\tilde{a}_7,\\
&\qquad m=0,\cdots,M-1
\end{aligned} \tag{6.10}$$

and

$$\begin{aligned}
\frac{U^{n+1}_{m,I}-U^n_{m,I}}{\Delta\tau} &= \tfrac{\tilde{a}_1 m^2}{2}(U^{n+1}_{m+1,I}-2U^{n+1}_{m,I}+U^{n+1}_{m-1,I}+U^n_{m+1,I}-2U^n_{m,I}+U^n_{m-1,I})\\
&+\tfrac{m}{4}(\tilde{a}_4+\tfrac{2}{\Delta\tau}\tfrac{s^{n+1}_{f,I}-s^n_{f,I}}{s^{n+1}_{f,I}+s^n_{f,I}})(U^{n+1}_{m+1,I}-U^{n+1}_{m-1,I}+U^n_{m+1,I}-U^n_{m-1,I})\\
&+\tfrac{\tilde{a}_5}{4\Delta\bar{r}}(3U^{n+1}_{m,I}-4U^{n+1}_{m,I-1}+U^{n+1}_{m,I-2}\\
&\qquad +3U^n_{m,I}-4U^n_{m,I-1}+U^n_{m,I-2})\\
&+\tfrac{\tilde{a}_6}{2}(U^{n+1}_{m,I}+U^n_{m,I})\\
&+\tilde{a}_7,\\
&\qquad m=0,\cdots,M-1.
\end{aligned} \tag{6.11}$$

Here $\tilde{a}_1$ and $\tilde{a}_4,\cdots,\tilde{a}_7$ are also evaluated at $(\xi_m,\bar{r}_i,\tau^{n+\frac{1}{2}})$, the formulae for $\tilde{a}_1$ and $\tilde{a}_4,\cdots,\tilde{a}_7$ are almost the same as these given above except that the partial derivative $\dfrac{\partial s_f}{\partial\bar{r}}$ is approximated by

$$\frac{\partial s_f}{\partial\bar{r}}=\frac{-{s_f}^{n+1}_2+4{s_f}^{n+1}_1-3{s_f}^{n+1}_0-{s_f}^n_2+4{s_f}^n_1-3{s_f}^n_0}{4\Delta\bar{r}}$$

and

$$\frac{\partial s_f}{\partial\bar{r}}=\frac{3{s_f}^{n+1}_I-4{s_f}^{n+1}_{I-1}+{s_f}^{n+1}_{I-2}+3{s_f}^n_I-4{s_f}^n_{I-1}+{s_f}^n_{I-2}}{4\Delta\bar{r}}$$

at $\bar{r}=0$ and $\bar{r}=1$ respectively. From the expression for a_4, because $w=0$ at $\bar{r}=0$ and $\bar{r}=1$, the one-sided second-order finite difference schemes for $\dfrac{\partial^2 s_f}{\partial\bar{r}^2}$ are not needed. In this case the way of discretizing the partial derivative with respect to r jumps from $i = 0$ to $i = 1$ and from $i = I - 1$ to $i = I$. This cause small problems on the results. In order to overcome this problem, at $i = 1, 2, 3$ and $i = I-3, I-2, I-1$, the partial derivative with respect to r actually is discretized by a mixture of central difference and one-sided difference. For example, for $i = 0, 1, 2, 3$,

$$\begin{aligned}\left(\frac{\partial U}{\partial \bar{r}}\right)_{m,i}^{n+1/2} \quad &= \quad \frac{3-i}{3} \cdot \frac{-U_{i+2}^{n+1} + 4U_{i+1}^{n+1} - 3U_i^{n+1} - U_{i+2}^n + 4U_{i+1}^n - 3U_i^n}{4\Delta\bar{r}} \\ &\quad + \frac{i}{3} \cdot \frac{U_{i+1}^{n+1} - U_{i-1}^{n+1} + U_{i+1}^n - 3U_{i-1}^n}{4\Delta\bar{r}}.\end{aligned}$$

6.2.3 $\xi = 1$ with $\bar{r} \in [0, 1]$

At the boundary $\xi = 1$, we have

$$U_{M,i}^{n+1} = g(s_{f,i}^{n+1}, \bar{r}_i, \tau^{n+1}), \quad i = 0, \cdots, I,$$

where

$$g(s_f, \bar{r}, \tau) = s_f(\bar{r}, \tau)[1 - e^{-D_0\tau}N(d_1)] - e^{-[r_l+\bar{r}(r_u-r_l)]\tau}N(-d_2)$$

and

$$\frac{3U_{M,i}^{n+1} - 4U_{M-1,i}^{n+1} + U_{M-2,i}^{n+1}}{2\Delta\xi} = h(s_{f,i}^{n+1}, \bar{r}_i, \tau^{n+1}), \quad i = 0, \cdots, I,$$

where $h(s_f, \bar{r}, \tau) = s_f(\bar{r}, \tau)[1 - e^{-D_0\tau}N(d_1)]$.

6.2.4 $\tau = 0$

At $\tau = 0$, we have

$$U_{m,i}^0 = 0, \quad m = 0, \cdots, M, \quad i = 0, \cdots, I \tag{6.12}$$

and

$$s_{f,i}^0 = \max(1, K/D_0), \quad i = 0, \cdots, I. \tag{6.13}$$

6.3 A Gauss-Seidel-type iteration method

Now we have a nonlinear system for $U_{m,i}^{n+1}$ and $s_{f,i}^{n+1}$, $m = 0, \cdots, M$ and $i = 0, \cdots, I$. To solve this system, we use a Gauss-Seidel-type iteration method.

Let $U_{m,i}^{(j)}$ and $s_{f,i}^{(j)}$ be the j-th iteration value of $U_{m,i}^{n+1}$ and $s_{f,i}^{n+1}$ respectively. Using this notation we write the nonlinear system given in the above subsection in the following form:

$$
\begin{aligned}
U_{m,i}^{(j)} \;=\; & \{\tfrac{\tilde{a}_1 m^2}{2}(U_{m+1,i}^{(j-1)} + U_{m-1,i}^{(j)} + U_{m+1,i}^{n} - 2U_{m,i}^{n} + U_{m-1,i}^{n}) \\
& + \tfrac{\tilde{a}_2 w m}{8\Delta\bar{r}}(U_{m+1,i+1}^{(j-1)} - U_{m+1,i-1}^{(j-1)} - U_{m-1,i+1}^{(j)} + U_{m-1,i-1}^{(j)} \\
& + U_{m+1,i+1}^{n} - U_{m+1,i-1}^{n} - U_{m-1,i+1}^{n} + U_{m-1,i-1}^{n}) \\
& + \tfrac{\tilde{a}_3}{2\Delta\bar{r}^2}(U_{m,i+1}^{(j-1)} + U_{m,i-1}^{(j)} + U_{m,i+1}^{n} - 2U_{m,i}^{n} + U_{m,i-1}^{n}) \\
& + \tfrac{m}{4}\left(\tilde{a}_4 + \tfrac{2(s_{f,i}^{(j-1)} - s_{f,i}^{n})}{\Delta\tau}\tfrac{1}{s_{f,i}^{(j-1)} + s_{f,i}^{n}}\right) \\
& \qquad \cdot(U_{m+1,i}^{(j-1)} - U_{m-1,i}^{(j)} + U_{m+1,i}^{n} - U_{m-1,i}^{n}) \\
& + \tfrac{\tilde{a}_5}{4\Delta\bar{r}}(U_{m,i+1}^{(j-1)} - U_{m,i-1}^{(j)} + U_{m,i+1}^{n} - U_{m,i-1}^{n}) \\
& + \tfrac{\tilde{a}_6}{2}U_{m,i}^{n} + \tilde{a}_7 \\
& + \tfrac{U_{m,i}^{n}}{\Delta\tau}\} \times \tfrac{1}{\tilde{a}_1 m^2 + \frac{\tilde{a}_3 w^2}{\Delta\bar{r}^2} + \frac{1}{\Delta\tau} - \frac{\tilde{a}_6}{2}}, \\
& \qquad m = 1, \cdots, M, \quad i = 1, \cdots, I-1;
\end{aligned}
\tag{6.14}
$$

at $m = 0$,

$$
\begin{aligned}
U_{0,i}^{(j)} \;=\; & \{\tfrac{\tilde{a}_3 w^2}{2\Delta\bar{r}^2}(U_{0,i+1}^{(j-1)} + U_{0,i-1}^{(j)} + U_{0,i+1}^{n} - 2U_{0,i}^{n} + U_{0,i-1}^{n}) \\
& + \tfrac{\tilde{a}_5}{4\Delta\bar{r}}(U_{0,i+1}^{(j-1)} - U_{0,i-1}^{(j)} + U_{0,i+1}^{n} - U_{0,i-1}^{n}) \\
& + \tfrac{\tilde{a}_6}{2}U_{0,i}^{n} + \tilde{a}_7 \\
& + \tfrac{U_{0,i}^{n}}{\Delta\tau}\} \times \tfrac{1}{\frac{\tilde{a}_3 w^2}{\Delta\bar{r}^2} + \frac{1}{\Delta\tau} - \frac{\tilde{a}_6}{2}}, \\
& \qquad i = 1, \cdots, I-1;
\end{aligned}
\tag{6.15}
$$

at $i = 0$,

$$\begin{aligned} U^{(j)}_{m,0} &= \{\tfrac{\tilde{a}_1 m^2}{2}(U^{(j-1)}_{m+1,0} + U^{(j)}_{m-1,0} + U^n_{m+1,0} - 2U^n_{m,0} + U^n_{m-1,0}) \\ &\quad + \tfrac{m}{4}\left(\tilde{a}_4 + \tfrac{2}{\Delta\tau}\tfrac{s^{(j-1)}_{f,0} - s^n_{f,0}}{s^{(j-1)}_{f,0} + s^n_{f,0}}\right)(U^{(j-1)}_{m+1,0} - U^{(j)}_{m-1,0} + U^n_{m+1,0} - U^n_{m-1,0}) \\ &\quad + \tfrac{\tilde{a}_5}{4\Delta\bar{r}}(-U^{(j-1)}_{m,2} + 4U^{(j-1)}_{m,1} - U^n_{m,2} + 4U^n_{m,1} - 3U^n_{m,0}) \\ &\quad + \tfrac{\tilde{a}_6}{2}U^n_{m,0} + \tilde{a}_7 \\ &\quad + \tfrac{U^n_{m,0}}{\Delta\tau}\} \times \tfrac{1}{\tilde{a}_1 m^2 + \frac{1}{\Delta\tau} + \frac{3\tilde{a}_5}{4\Delta\bar{r}} - \frac{\tilde{a}_6}{2}}, \\ &\qquad m = 0, \cdots, M-1; \end{aligned} \tag{6.16}$$

at $i = I$,

$$\begin{aligned} U^{(j)}_{m,I} &= \{\tfrac{\tilde{a}_1 m^2}{2}(U^{(j-1)}_{m+1,I} + U^{(j)}_{m-1,I} + U^n_{m+1,I} - 2U^n_{m,I} + U^n_{m-1,I}) \\ &\quad + \tfrac{m}{4}\left(\tilde{a}_4 + \tfrac{2}{\Delta\tau}\tfrac{s^{(j-1)}_{f,I} - s^n_{f,I}}{s^{(j-1)}_{f,I} + s^n_{f,I}}\right)(U^{(j-1)}_{m+1,I} - U^{(j)}_{m-1,I} + U^n_{m+1,I} - U^n_{m-1,I}) \\ &\quad + \tfrac{\tilde{a}_5}{4\Delta\bar{r}}(-4U^{(j)}_{m,I-1} + U^{(j)}_{m,I-2} + 3U^n_{m,I} - 4U^n_{m,I-1} + U^n_{m,I-2}) \\ &\quad + \tfrac{\tilde{a}_6}{2}U^n_{m,I} + \tilde{a}_7 \\ &\quad + \tfrac{U^n_{m,I}}{\Delta\tau}\} \times \tfrac{1}{\tilde{a}_1 m^2 + \frac{1}{\Delta\tau} - \frac{\tilde{a}_6}{2} - \frac{3\tilde{a}_5}{4\Delta\bar{r}}}, \\ &\qquad m = 0, \cdots, M-1; \end{aligned} \tag{6.17}$$

at $m = M$,

$$\begin{aligned} U^{(j)}_{M,i} &= s^{(j)}_{f,i}[1 - e^{-D_0\tau^{n+1}}N(d_1)] - e^{-[r_l + \bar{r}_i(r_u - r_l)]\tau^{n+1}}N(-d_2), \\ &\qquad i = 0, \cdots, I \end{aligned} \tag{6.18}$$

and

$$\begin{aligned} \frac{3U^{(j)}_{M,i} - 4U^{(j)}_{M-1,i} + U^{(j)}_{M-2,i}}{2\Delta\xi} &= s^{(j)}_{f,i}[1 - e^{-D_0\tau^{n+1}}N(d_1)], \\ &\qquad i = 0, \cdots, I, \end{aligned} \tag{6.19}$$

where

$$d_1 = \left[\ln(s^{(j-1)}_{f,i} e^{[r_l + \bar{r}_i(r_u - r_l) - D_0]\tau^{n+1}}) + \frac{1}{2}\sigma^2\tau^{n+1}\right] / (\sigma\sqrt{\tau^{n+1}})$$

and

$$d_2 = d_1 - \sigma\sqrt{\tau^{n+1}}.$$

In (6.14)-(6.17), $\tilde{a}_l$ stands for $\frac{1}{2}(a_{l,m,i}^{(j-1)} + a_{l,m,i}^n)$, $l = 1, \cdots, 7$.

Thus when we know $U_{m,i}^{(j-1)}$ and $s_{f,i}^{(j-1)}$, $m = 0, 1, \cdots, M$, $i = 0, 1, \cdots, I$, we can have $U_{m,i}^{(j)}$ and $s_{f,i}^{(j)}$, $m = 0, 1, \cdots, M$, $i = 0, 1, \cdots, I$, in the following way. First we can use (6.16) with $m = 0$, (6.15), and (6.17) with $m = 0$ to obtain $U_{0,i}^{(j)}$, $i = 0, 1, \cdots, I$, successively. When $U_{m-1,i}^{(j)}$, $i = 0, 1, \cdots, I$, are known, we can use (6.16), (6.14), and (6.17) to obtain $U_{m,i}^{(j)}$, $i = 0, 1, \cdots, I$. This procedure can be done for $m = 1, 2, \cdots, M-1$ successively. After that, find $U_{M,i}^{(j)}$ and $s_{f,i}^{(j)}$ by using (6.18) and (6.19), $i = 0, 1, \cdots, I$. We call this procedure a Gauss-Seidel-type iteration. Therefore, let $U_{m,i}^{(0)} = U_{m,i}^n$ and $s_{f,i}^{(0)} = s_{f,i}^n$, we can have the Gauss-Seidel-type iteration for $j = 0, 1, \cdots$, successively. When we have $U_{m,i}^{(j)}$ and $s_{f,i}^{(j)}$ from $U_{m,i}^{(j-1)}$ and $s_{f,i}^{(j-1)}$, we can check if the following inequality holds:

$$\sum_{m=0}^{M}\sum_{i=0}^{I}(U_{m,i}^{(j)} - U_{m,i}^{(j-1)})^2 + \sum_{i=0}^{I}(s_{f,i}^{(j)} - s_{f,i}^{(j-1)})^2 \le \epsilon^2,$$

where ϵ^2 is a small number given according to the required accuracy. If it holds, we can stop the iteration and have gotten $U_{m,i}^{n+1}$ and $s_{f,i}^{n+1}$ from $U_{m,i}^n$ and $s_{f,i}^n$.

Starting from $U_{m,i}^0$ and $s_{f,i}^0$ given by (6.12) and (6.13), we can perform the procedure of finding $U_{m,i}^{n+1}$ and $s_{f,i}^{n+1}$ from $U_{m,i}^n$ and $s_{f,i}^n$ for $n = 0, 1, \cdots, N-1$ successively and finally obtain $U_{m,i}^N$ and $s_{f,i}^N$.

For some cases, for example, convertible bonds with call/put provision, it will be easier to solve the model problem(3.4). To solve the model problem (3.4), we consider the following transformation:

$$\begin{cases} \xi = \dfrac{S}{S + Z/n}, \\ \tau = T - t. \end{cases} \tag{6.20}$$

and

$$U(\xi, r, \tau) = \frac{B(S, r, t)}{nS + Z}. \tag{6.21}$$

we have

$$\left\{\begin{array}{rcl}\dfrac{\partial B}{\partial t} & = & -(nS+Z)\dfrac{\partial U}{\partial \tau},\\ \dfrac{\partial B}{\partial S} & = & nU+(nS+Z)\dfrac{\partial U}{\partial \xi}\dfrac{\partial \xi}{\partial S},\\ \dfrac{\partial B}{\partial r} & = & (nS+Z)\dfrac{\partial U}{\partial r},\\ \dfrac{\partial^2 B}{\partial S^2} & = & (nS+Z)\dfrac{\partial^2 U}{\partial \xi^2}\left(\dfrac{\partial \xi}{\partial S}\right)^2,\\ \dfrac{\partial^2 B}{\partial S\partial r} & = & \left((nS+Z)\dfrac{\partial^2 U}{\partial \xi\partial r}\dfrac{\partial \xi}{\partial S}+n\dfrac{\partial U}{\partial r}\right),\\ \dfrac{\partial^2 B}{\partial r^2} & = & (nS+Z)\dfrac{\partial^2 U}{\partial r^2}.\end{array}\right. \tag{6.22}$$

Substituting these relations into (3.4), we have the following problem in the domain $[0,1]\times[r_l,r_u]\times[0,T]$ in (ξ,r,τ)-space:

$$\left\{\begin{array}{l}\min\left(-\left(\dfrac{\partial U}{\partial t}-\mathbf{L}_{\xi,r}U-a_7\right),\ U(\xi,r,\tau)-\xi\right)=0,\\ \\ U(\xi,r,0)=\max(1-\xi,\xi).\end{array}\right. \tag{6.23}$$

Here the operator $\mathbf{L}_{\xi,r}$ is defined as follows:

$$\mathbf{L}_{\xi,r} = a_1\xi^2(1-\xi)^2\frac{\partial^2}{\partial \xi^2}+a_2\xi(1-\xi)\frac{\partial^2}{\partial\xi\partial r}+a_3\frac{\partial^2}{\partial r^2}+a_4\xi(1-\xi)\frac{\partial}{\partial \xi}+a_5\frac{\partial}{\partial r}+a_6$$

and the expressions for a_i, $i=1,\cdots,7$ are

$$\begin{array}{rcl} a_1 & = & \frac{1}{2}\sigma^2,\\ a_2 & = & \rho\sigma w,\\ a_3 & = & \frac{1}{2}w^2,\\ a_4 & = & r-D_0,\\ a_5 & = & (\rho\sigma w\xi+u-\lambda w),\\ a_6 & = & -(r(1-\xi)+D_0\xi),\\ a_7 & = & k(1-\xi).\end{array}$$

When the coupon payments are discrete, i.e., there are n individual payments k_i paid at time $\tau_i \leq T, i = 1, \cdots, n,$, then $a_7$becomes

$$a_7 = \sum_{i=1}^{n} k_i \delta(\tau_i)(1-\xi)/Z. \tag{6.24}$$

Then we can use similar method introduced in Section 6.2 to discretize the formulation (6.23) and solve it by using the Gauss-Seidel-type projected iteration method. Assume that

$$\begin{aligned}
\xi_m &= m\Delta\xi, & m &= 0, \cdots, M, \\
r_i &= i\Delta r, & i &= 0, \cdots, I, \\
\tau^n &= n\Delta\tau, & n &= 0, \cdots, N,
\end{aligned}$$

where M, N, I are given integers, $\Delta\xi = 1/M$, $\Delta r = (r_u - r_l)/I$ and $\Delta\tau = T/N$. Let $U^n_{m,i}$ denote the approximate value of $U(\xi_m, r_i, \tau^n)$.

For each interior point, the PDE in (6.23) can be discretized at the point $(\xi_m, r_i, \tau^{n+1/2})$ as

$$\begin{aligned}
\tfrac{U^{n+1}_{m,i} - U^n_{m,i}}{\Delta\tau} = \; & \tfrac{\tilde{a}_1 m^2(1-m\Delta\xi)^2}{2}(U^{n+1}_{m+1,i} - 2U^{n+1}_{m,i} + U^{n+1}_{m-1,i} + U^n_{m+1,i} - 2U^n_{m,i} + U^n_{m-1,i}) \\
& + \tfrac{\tilde{a}_2 m(1-m\Delta\xi)}{8\Delta r}(U^{n+1}_{m+1,i+1} - U^{n+1}_{m+1,i-1} - U^{n+1}_{m-1,i+1} + U^{n+1}_{m-1,i-1} \\
& + U^n_{m+1,i+1} - U^n_{m+1,i-1} - U^n_{m-1,i+1} + U^n_{m-1,i-1}) \\
& + \tfrac{\tilde{a}_3}{2\Delta r^2}(U^{n+1}_{m,i+1} - 2U^{n+1}_{m,i} + U^{n+1}_{m,i-1} + U^n_{m,i+1} - 2U^n_{m,i} + U^n_{m,i-1}) \\
& + \tfrac{\tilde{a}_4 m(1-m\Delta\xi)}{4}(U^{n+1}_{m+1,i} - U^{n+1}_{m-1,i} + U^n_{m+1,i} - U^n_{m-1,i}) \\
& + \tfrac{\tilde{a}_5}{4\Delta r}(U^{n+1}_{m,i+1} - U^{n+1}_{m,i-1} + U^n_{m,i+1} - U^n_{m,i-1}) \\
& + \tfrac{\tilde{a}_6}{2}(U^{n+1}_{m,i} + U^n_{m,i}) \\
& + \tilde{a}_7, \\
& \quad m = 0, \cdots, M, \quad i = 1, \cdots, I-1.
\end{aligned} \tag{6.25}$$

$\tilde{a}_1, \cdots, \tilde{a}_7$ should take the value at the point $(\xi_m, r_i, \tau^{n+1/2})$ and their expressions are:

$$\begin{aligned}
\tilde{a}_1 \equiv a_{1,m,i}^{n+\frac{1}{2}} &= \tfrac{1}{2}(\sigma_{m,i}^{n+\frac{1}{2}})^2, \\
\tilde{a}_2 \equiv a_{2,m,i}^{n+\frac{1}{2}} &= \rho_{m,i}^{n+\frac{1}{2}} \sigma_{m,i}^{n+\frac{1}{2}} w_{m,i}^{n+\frac{1}{2}}, \\
\tilde{a}_3 \equiv a_{3,m,i}^{n+\frac{1}{2}} &= \tfrac{1}{2} w_{m,i}^{n+\frac{1}{2}}, \\
\tilde{a}_4 \equiv a_{4,m,i}^{n+\frac{1}{2}} &= (r_l + i\Delta r - D_0), \\
\tilde{a}_5 \equiv a_{5,m,i}^{n+\frac{1}{2}} &= \rho_{m,i}^{n+\frac{1}{2}} \sigma_{m,i}^{n+\frac{1}{2}} w_{m,i}^{n+\frac{1}{2}} m\Delta\xi + u_{m,i}^{n+\frac{1}{2}} - \lambda_{m,i}^{n+\frac{1}{2}} w_{m,i}^{n+\frac{1}{2}}, \\
\tilde{a}_6 \equiv a_{6,m,i}^{n+\frac{1}{2}} &= -[(r_l + i\Delta r)(1 - m\Delta\xi) + D_0 m\Delta\xi], \\
\tilde{a}_7 \equiv a_{7,m,i}^{n+\frac{1}{2}} &= k(1 - m\Delta\xi).
\end{aligned} \tag{6.26}$$

When the coupon payments are discrete, then $a_7$becomes

$$\tilde{a}_7 = \sum_{i=1}^{n} k_i \delta(\tau_i)(1 - m\Delta\xi)/Z. \tag{6.27}$$

At the boundaries $r = r_l$ and $r = r_u$, due to $w = 0$, the partial differential equations in (6.23) becomes

$$\frac{\partial U}{\partial \tau} = a_1 \xi^2 (1-\xi)^2 \frac{\partial^2 U}{\partial \xi^2} + a_4 \xi(1-\xi) \frac{\partial U}{\partial \xi} + a_5 \frac{\partial U}{\partial r} + a_6 U + a_7. \tag{6.28}$$

Thus we can approximate the partial differential equation in (6.23) at the boundaries $r = r_l$ and $r = r_u$ by

$$\begin{aligned}
\frac{U_{m,0}^{n+1} - U_{m,0}^{n}}{\Delta\tau} = {} & \frac{\tilde{a}_1 m^2 (1-m\Delta\xi)^2}{2} (U_{m+1,0}^{n+1} - 2U_{m,0}^{n+1} + U_{m-1,0}^{n+1} + U_{m+1,0}^{n} - 2U_{m,0}^{n} + U_{m-1,0}^{n}) \\
& + \frac{\tilde{a}_4 m(1-m\Delta\xi)}{4} (U_{m+1,0}^{n+1} - U_{m-1,0}^{n+1} + U_{m+1,0}^{n} - U_{m-1,0}^{n}) \\
& + \frac{\tilde{a}_5}{4\Delta r} (-U_{m,2}^{n+1} + 4U_{m,1}^{n+1} - 3U_{m,0}^{n+1} - U_{m,2}^{n} + 4U_{m,1}^{n} - 3U_{m,0}^{n}) \\
& + \frac{\tilde{a}_6}{2} (U_{m,0}^{n+1} + U_{m,0}^{n}) \\
& + \tilde{a}_7, \\
& \quad m = 0, \cdots, M
\end{aligned} \tag{6.29}$$

and

$$\begin{aligned}
\frac{U^{n+1}_{m,I}-U^{n}_{m,I}}{\Delta\tau} \quad = \quad & \frac{\tilde{a}_1 m^2(1-m\Delta\xi)^2}{2}(U^{n+1}_{m+1,I}-2U^{n+1}_{m,I}+U^{n+1}_{m-1,I}+U^{n}_{m+1,I}-2U^{n}_{m,I}+U^{n}_{m-1,I}) \\
& +\frac{\tilde{a}_4 m(1-m\Delta\xi)}{4}(U^{n+1}_{m+1,I}-U^{n+1}_{m-1,I}+U^{n}_{m+1,I}-U^{n}_{m-1,I}) \\
& +\frac{\tilde{a}_5}{4\Delta r}(3U^{n+1}_{m,I}-4U^{n+1}_{m,I-1}+U^{n+1}_{m,I-2} \\
& \qquad +3U^{n}_{m,I}-4U^{n}_{m,I-1}+U^{n}_{m,I-2}) \\
& +\frac{\tilde{a}_6}{2}(U^{n+1}_{m,I}+U^{n}_{m,I}) \\
& +\tilde{a}_7, \\
& \qquad m=0,\cdots,M.
\end{aligned} \tag{6.30}$$

At $\tau = 0$, we consider that the issuer can only pay part of the bond value when ξ_m is close to 0, so we adjust the final condition as:

$$U^0_{m,i} = \max(f(\xi_m), m\Delta\xi), \quad m=0,\cdots,M, \quad i=0,\cdots,I. \tag{6.31}$$

where

$$f(\xi_m) = \begin{cases} a\xi_m^2 + b\xi_m + c, & if \quad \xi_m \le \xi_1, \\ 1-\xi_m, & if \quad \xi_m > \xi_1. \end{cases} \tag{6.32}$$

and $a = (c-1)/(\xi_1)^2, b = -1-2a\xi_1, c = \xi_2, \xi_1 = 0.1$ and $\xi_2 = 0.6$.

Now we have a system for $U^{n+1}_{m,i}$, $m = 0,\cdots,M$ and $i = 0,\cdots,I$. To solve this system, we also use a Gauss-Seidel-type iteration method.

Let $U^{(j)}_{m,i}$ be the j-th iteration value of $U^{n+1}_{m,i}$ respectively. Using this notation we write the system in the following form:

$$\begin{aligned}
U_{m,i}^{(j)} \;=\;& \{\tfrac{\tilde{a}_1 m^2(1-m\Delta\xi)^2}{2}(U_{m+1,i}^{(j-1)}+U_{m-1,i}^{(j)}+U_{m+1,i}^{n}-2U_{m,i}^{n}+U_{m-1,i}^{n})\\
&+\tfrac{\tilde{a}_2 m(1-m\Delta\xi)}{8\Delta r}(U_{m+1,i+1}^{(j-1)}-U_{m+1,i-1}^{(j-1)}-U_{m-1,i+1}^{(j)}+U_{m-1,i-1}^{(j)}\\
&+U_{m+1,i+1}^{n}-U_{m+1,i-1}^{n}-U_{m-1,i+1}^{n}+U_{m-1,i-1}^{n})\\
&+\tfrac{\tilde{a}_3 w^2}{2\Delta r^2}(U_{m,i+1}^{(j-1)}+U_{m,i-1}^{(j)}+U_{m,i+1}^{n}-2U_{m,i}^{n}+U_{m,i-1}^{n})\\
&+\tfrac{\tilde{a}_4 m(1-m\Delta\xi)}{4}(U_{m+1,i}^{(j-1)}-U_{m-1,i}^{(j)}+U_{m+1,i}^{n}-U_{m-1,i}^{n})\\
&+\tfrac{\tilde{a}_5}{4\Delta r}(U_{m,i+1}^{(j-1)}-U_{m,i-1}^{(j)}+U_{m,i+1}^{n}-U_{m,i-1}^{n})\\
&+\tfrac{\tilde{a}_6}{2}U_{m,i}^{n}+\tilde{a}_7\\
&+\tfrac{U_{m,i}^{n}}{\Delta\tau}\}\times\frac{1}{\tilde{a}_1 m^2(1-m\Delta\xi)^2+\frac{\tilde{a}_3 w^2}{\Delta r^2}+\frac{1}{\Delta\tau}-\frac{\tilde{a}_6}{2}},\\
&\quad m=0,\cdots,M,\quad i=1,\cdots,I-1;
\end{aligned}\tag{6.33}$$

at $i=0$,

$$\begin{aligned}
U_{m,0}^{(j)} \;=\;& \{\tfrac{\tilde{a}_1 m^2(1-m\Delta\xi)^2}{2}(U_{m+1,0}^{(j-1)}+U_{m-1,0}^{(j)}+U_{m+1,0}^{n}-2U_{m,0}^{n}+U_{m-1,0}^{n})\\
&+\tfrac{\tilde{a}_4 m(1-m\Delta\xi)}{4}(U_{m+1,0}^{(j-1)}-U_{m-1,0}^{(j)}+U_{m+1,0}^{n}-U_{m-1,0}^{n})\\
&+\tfrac{\tilde{a}_5}{4\Delta r}(-U_{m,2}^{(j-1)}+4U_{m,1}^{(j-1)}-U_{m,2}^{n}+4U_{m,1}^{n}-3U_{m,0}^{n})\\
&+\tfrac{\tilde{a}_6}{2}U_{m,0}^{n}+\tilde{a}_7\\
&+\tfrac{U_{m,0}^{n}}{\Delta\tau}\}\times\frac{1}{\tilde{a}_1 m^2(1-m\Delta\xi)^2+\frac{1}{\Delta\tau}+\frac{3\tilde{a}_5}{4\Delta r}-\frac{\tilde{a}_6}{2}},\\
&\quad m=0,\cdots,M;
\end{aligned}\tag{6.34}$$

at $i=I$,

$$\begin{aligned}
U_{m,I}^{(j)} \;=\;& \{\tfrac{\tilde{a}_1 m^2(1-m\Delta\xi)^2}{2}(U_{m+1,I}^{(j-1)}+U_{m-1,I}^{(j)}+U_{m+1,I}^{n}-2U_{m,I}^{n}+U_{m-1,I}^{n})\\
&+\tfrac{\tilde{a}_4 m(1-m\Delta\xi)}{4}(U_{m+1,I}^{(j-1)}-U_{m-1,I}^{(j)}+U_{m+1,I}^{n}-U_{m-1,I}^{n})\\
&+\tfrac{\tilde{a}_5}{4\Delta r}(-4U_{m,I-1}^{(j)}+U_{m,I-2}^{(j)}+3U_{m,I}^{n}-4U_{m,I-1}^{n}+U_{m,I-2}^{n})\\
&+\tfrac{\tilde{a}_6}{2}U_{m,I}^{n}+\tilde{a}_7\\
&+\tfrac{U_{m,I}^{n}}{\Delta\tau}\}\times\frac{1}{\tilde{a}_1 m^2(1-m\Delta\xi)^2+\frac{1}{\Delta\tau}-\frac{\tilde{a}_6}{2}-\frac{3\tilde{a}_5}{4\Delta r}},\\
&\quad m=0,\cdots,M;
\end{aligned}\tag{6.35}$$

Thus when we know $U_{m,i}^{(j-1)}$, $m=0,1,\cdots,M$, $i=0,1,\cdots,I$, we can have $U_{m,i}^{(j)}$,

$m = 0, 1, \cdots, M$, $i = 0, 1, \cdots, I$. Obviously, $U_{m,i}^{(j)}$ should be not less than $m\Delta\xi$. Therefore, for convertible bonds without call/put provision,

$$U_{m,i}^{(j)} = \max(U_{m,i}^{(j)}, \xi_m).$$

When we use the formulation (6.23) to get the theoretical price of a callable convertible, suppose that the bond has a call provision with a call value C and $U_{m,i}^{n+1}$ got by the iteration at each step, we also consider the call price C should be adjusted when ξ_m is close to zero. Therefor we need add one more step

$$U_{m,i}^{(j)} = \max(\min(f(\xi_m)C/Z, U_{m,i}^{(j)}), \xi_m), \tag{6.36}$$

in order to get the callable price.

When we use the formulation (6.23) to get the theoretical price of a puttable convertible, suppose that the bond has a put provision with a put value P and $U_{m,i}^{n+1}$ got by the iteration at each step, we need add one more step

$$U_{m,i}^{(j)} = \max(f(\xi_m)P/Z, U_{m,i}^{(j)}, \xi_m). \tag{6.37}$$

in order to get the puttable price.

When we use the formulation (6.23) to get the theoretical price of a callable/puttable convertible, suppose that the bond has a call provision with a call value C and a put provision with a put value P and $U_{m,i}^{n+1}$ got by the iteration at each step, we need add one more step

$$U_{m,i}^{(j)} = \max(f(\xi_m)P/Z, \max(\min(f(\xi_m)C/Z, U_{m,i}^{(j)}), \xi_m)). \tag{6.38}$$

in order to get the callable/puttable price.

When we have $U_{m,i}^{(j)}$ from $U_{m,i}^{(j-1)}$, we can check if the following inequality holds:

$$\max_{m=0,\cdots,M} \max_{i=0,\cdots,I} (U_{m,i}^{(j)} - U_{m,i}^{(j-1)})^2 \leq \epsilon^2,$$

where ϵ^2 is a small number given according to the required accuracy. If it holds, we can stop the iteration and have gotten $U_{m,i}^{n+1}$ from $U_{m,i}^{n}$.

CHAPTER 7: NUMERICAL RESULTS

The LC problem after using linear transform, the free-boundary problem after using SSM and linear transform are given in the equation (6.6) and in the equation (6.23), and their discretization have been discussed in chapter 6.

In this chapter, we will evaluate convertible bonds by our models and go over some numerical results. We will evaluate the two-factor convertible bond in a way that the market price of risk model is proposed by Sun [8] as a test example of our singularity separation technique. Later on, we will implement our inverse problem model to determine the market price of risk with data on markets: the volatility of the interest rates $w(r)$ is calculated by the least squares method with different interest rates; the market price of risk $\lambda(t)$ is calculated by solving the inverse problem; and the mathematical model of evaluating the prices of convertible bond and convertible bond with call and put features are solved numerically with Gaussian Seidel iterations. The comparisions between two different models (one with SSM and the other without it) are discussed. Finally, the numerical results are tested by market prices and our results have achieved a very good agreement with the market prices.

7.1 Convertible bond model without a specified market price of risk

Brennan and Schwartz [2] proposed the following model for the spot interest rate:

$$w(r,t) = \alpha r$$

and

$$u - \lambda(t)w = -\gamma r + \delta,$$

Table 7.1: Values of a convertible bond with $T = 30$ at $S = 1$ and $r = 0.05$. $w(r,t) = 0.26r\phi(r; r_l, r_u)$ and $u - \lambda(t)w = -0.13r + 0.008$, where $\phi(r; r_l, r_u) = 1$ if $r_l \le r \le \frac{r_u + r_l}{2}$ and $\phi(r; r_l, r_u) = \left[\frac{4(r-r_l)(r_u-r)}{(r_u-r_l)^2}\right]^{1/4}$ if $\frac{r_u+r_l}{2} < r \le r_u$, r_l being 0 and r_u being 0.30. The other parameters are $k = 0.06$, $\sigma = 0.2$, $\rho = -0.01$, $D_0 = 0.05$, $Z = 1$, and $n = 1$.

Meshes	Results	|Errors|	CPU (sec.)
$10 \times 10 \times 10$	1.310535732	0.001144	0.08
$20 \times 20 \times 20$	1.311431494	0.000252	0.29
$40 \times 40 \times 40$	1.311588189	0.000095	9.02
$80 \times 80 \times 80$	1.311651155	0.000032	74.06

where $\alpha = 0.26$, $\gamma = 0.13$, and $\delta = 0.008$. This model was modified by Sun [8] in his thesis. Instead of $w(r,t) = \alpha r$, the expression for $w(r,t)$ is

$$w(r,t) = \alpha r \phi(r; r_l, r_u),$$

where

$$\phi(r; r_l, r_u) = \begin{cases} 1 & \text{if} \quad r_l \le r \le \dfrac{r_u + r_l}{2}, \\ \left[\dfrac{4(r - r_l)(r_u - r)}{(r_u - r_l)^2}\right]^{1/4} & \text{if} \quad \dfrac{r_u + r_l}{2} < r \le r_u. \end{cases}$$

In this case $u(r,t)$ and $w(r,t)$ satisfy the reversion conditions and no boundary condition is needed at $r = r_u$ and $r = r_l$. In the thesis by Sun [8] the other parameters used are

$$\begin{aligned} k = 0.06, \quad \sigma = 0.2, \quad \rho = -0.01, \quad D_0 = 0.05, \\ Z = 1, \quad n = 1, \quad r_l = 0.0, \quad r_u = 0.3, \quad T = 30, \end{aligned} \tag{7.1}$$

and the results for this problem were given.

We tried the same problem and our values at $S = 1$ and $r = 0.05$ are given in Table 7.1. There $a \times b \times c$ means a mesh with $M = a, I = b, N = c$. Our results match with the results in the thesis by Sun [8]. By the way the errors and CPU times in seconds needed are also given in Table 7.1. When an error with six decimals is calculated, we need an "exact" result with seven decimals. In this case $B_c = 1.3116835$, which was given in the thesis by Sun [8] and is confirmed by us. In this model the credit rating of a company is not considered, so that the results obtained in this subsection are hard to be used in practice.

7.2 Convertible bond model with a specified market price of risk

7.2.1 $w(r)$ and $\lambda(t)$

As we stated before, the market data-oriented $w(r)$ is determined by statistics and the least squares method. In our numerical experiment, one month LIBORs on U.S. dollar during January 1977–August 2010 are used. In the data the minimum interest rate $r_{\min}$ is 0.0022906 and the maximum interest rate $r_{\max}$ is 0.23562. Thus we let the lower bound r_l be 0.0 and the upper bound r_u be 0.24 in (5.5). Taking $Q = 40$ and using the method described in chapter 5, for a_0, b_0 and c_0 we have

$$a_0 = 4.1, \quad b_0 = -0.51, \quad c_0 = 0.0224.$$

Thus, $w(r)$ has an analytic form of

$$w(r) = (r - r_l)(r_u - r)(4.1r^2 - 0.51r + 0.0224). \tag{7.2}$$

Figure 7.1 shows the plot of $w(r)$. By the way, the data from statistics are also shown as "$\circ$" in Figure 7.1.

Using the function $w(r)$ obtained and choosing $u(r) = 0$, we can obtain the market price of risk $\lambda(t)$ through solving an inverse problem based on a zero-coupon bond curve. Such a curve is generated by the cubic spline interpolation based on the interest rates for 3-month, 6-month, 1-year, 2-year, 3-year, 5-year, 10-year, 15-year, 20-year, 30-year bonds. When the interest rates for bonds of companies with credit rating AAA on August 7, 2009 are used, the curve is shown in Figure 7.2. The corresponding market price of risk $\lambda(t)$ is plotted in Figure 7.3.

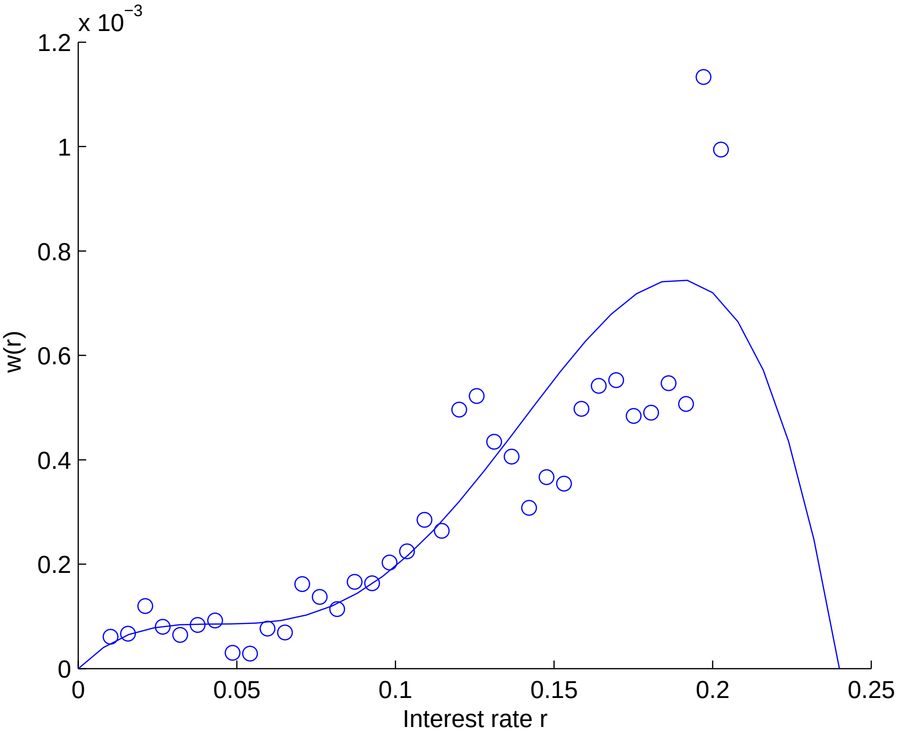

Figure 7.1: $w(r)$ in the equation (7.2) with $r_l = 0.0$ and $r_u = 0.24$, where $a_0 = 4.1$, $b_0 = -0.51$, $c_0 = 0.0224$. The line is the numerical result, and the circle is the market data.

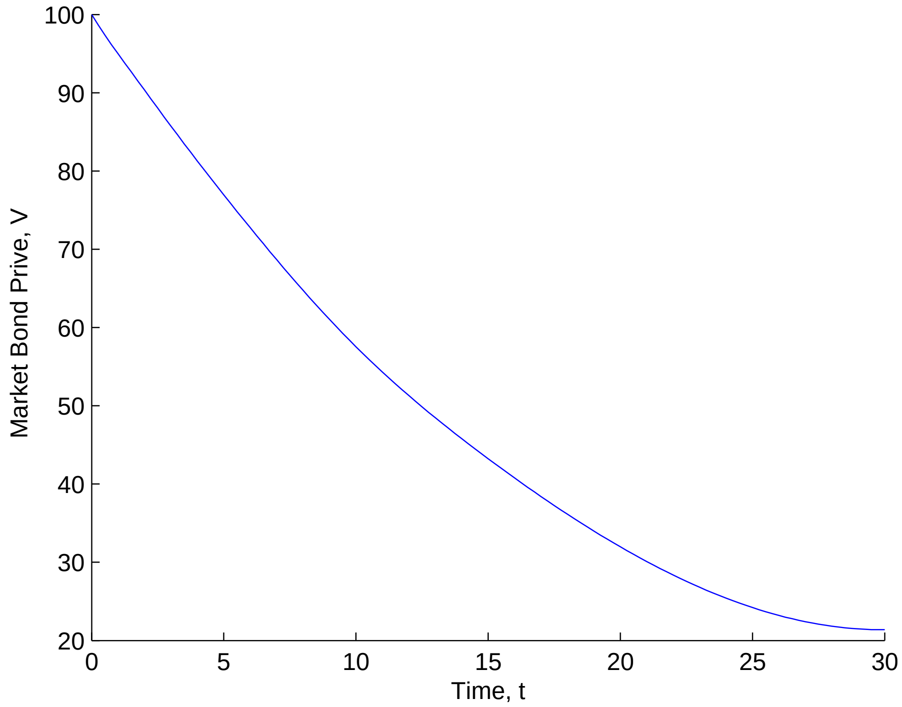

Figure 7.2: A cubic-spline interpolation curve of bonds with credit rating AAA based on the value of: 3-month, 6-month, 1-year, 2-year, 3-year, 5-year, 10-year, 15-year, 20-year, 30-year zero coupon bonds.

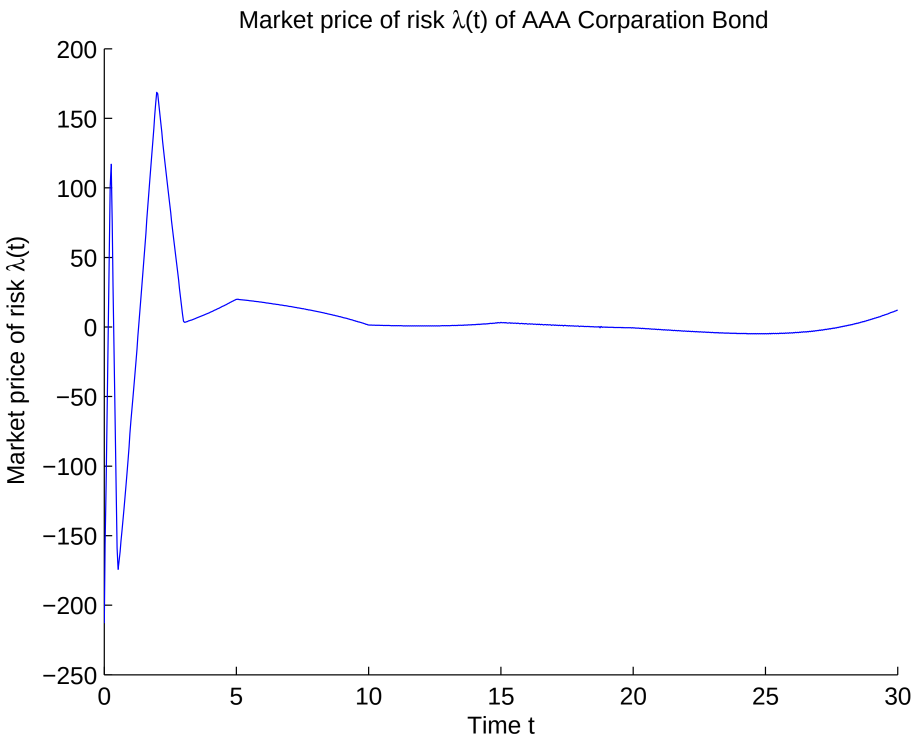

Figure 7.3: The numerical result of $\lambda(t)$ for companies with credit rating AAA by solving an inverse problem with different interest rates bonds data: 3-month, 6-month, 1-year, 2-year, 3-year, 5-year, 10-year, 15-year, 20-year, 30-year zero coupon bonds. The lower and upper bounds of the interest rates r are $r_l = 0.0$ and $r_u = 0.24$.

7.2.2 Different credit rating convertible bonds

In order to evaluating convertible bonds issued by companies with different credit ratings, what we need to do is to input different sets of interest rates for bonds and to generate different $\lambda(t)$. Besides this, other parts of computation are the same. The other parameters in (6.6) are constants and we set them to be:

$$\begin{gathered} \sigma = 0.2, \quad \rho = -0.01, \quad D_0 = 0.05, \\ Z = 1, \quad n = 1. \end{gathered} \tag{7.3}$$

We consider convertible bonds issued by companies with credit rating AAA, AA, BBB, or BB. Two sets of the convertible bond values are given in Tables 7.2 and 7.3. By the way, the interest rates of three-month bonds for companies with different credit ratings, the interest rates r^* for different companies, are also given in the two tables.

Table 7.2: Values of convertible bonds with different credit ratings at $S = 1$. $w(r) = (r - r_l)(r_u - r)(4.1r^2 - 0.51r + 0.0224)$, r_l being 0 and r_u being 0.24, $u = 0$, and $\lambda(t)$ is determined by solving an inverse problem. The other parameters are $k = 0.06, \sigma = 0.2, \rho = -0.01, D_0 = 0.05, Z = 1$, and $n = 1$.

Convertible Bonds	$T = 5$ years	$T = 10$ years	$T = 30$ years	r^*
AAA	1.175584806	1.228230226	1.298332213	0.003244
AA	1.169335272	1.216584933	1.263766584	0.010358
BBB	1.161230073	1.200970074	1.233405172	0.028973
BB	1.130830708	1.148463551	1.150208262	0.050115

The interest rates used for bonds with various maturities are the market data on August 7, 2009. In Table 7.2, for all the companies the coupon rates k are equal to 0.06 and in Table 7.3 the coupon rates k are functions of t and different for companies with different credit ratings. The functions $k(t)$ are chosen so that the values of bonds for r^*, $B_c(0, r^*, t)$, are almost equal to one for any t. In this case, $B_c(S, r^*, t) - 1$ represents the value of "convertible". In Tables 7.2 and 7.3 only the values at a specified point

Table 7.3: Values of convertible bonds with different credit ratings at $S = 1$. $w(r) = (r - r_l)(r_u - r)(4.1r^2 - 0.51r + 0.0224)$, r_l being 0 and r_u being 0.24, $u = 0$, and $\lambda(t)$ is determined by solving an inverse problem. The other parameters are $k = k(t)$, $k(t)$ being a given function, $\sigma = 0.2, \rho = -0.01, D_0 = 0.05, Z = 1$, and $n = 1$.

Convertible Bonds	$T = 5$ years	$T = 10$ years	$T = 30$ years	r^*
AAA	1.143485461	1.177327741	1.224142083	0.003244
AA	1.142743542	1.174748883	1.212606631	0.010358
BBB	1.146878726	1.18381062	1.228147746	0.028973
BB	1.168840727	1.220962979	1.286091767	0.050115

are given. In order to show how the entire solution looks like, in Figure 7.4 the value of a 30-year convertible bond for a company with credit rating AAA is plotted. In this figure the free boundary is also given as a bold solid line.

7.3 Comparison between formulations (3.11) and (4.11)

As we pointed out, we can price a convertible bond by using the formulation (3.11) or (4.11) and the formulation (4.11) should be used for $t \approx T$ because it is more efficient for this case. Here we give examples to support such a conclusion. Consider a convertible bond with $T = 0.5$ and we price such a convertible bond by using the formulations (3.11) and (4.11). The results, including the error and CPU time used, are given in Tables 7.4 and 7.5. From the two tables, we can see that for the same mesh, the error of the results obtained by using (4.11) is much smaller than that obtained by using (3.11). Even though the CPU time needed for the former is much greater than that for the latter, the former is still more efficient than the latter.

Here we give two examples. The first one is: in order to have a result with an error of 0.003 or so, the CPU time for the former is about 0.01 second, but the CPU time for the latter is more than 0.13 second. The second one is: in order to have a result with an error of 0.0006 or so, the CPU time for the former is about 0.05 second, but the CPU time for the latter is more than 1.23 second. From the two examples, we can see that in order to have the same accuracy, the CPU time needed for the latter

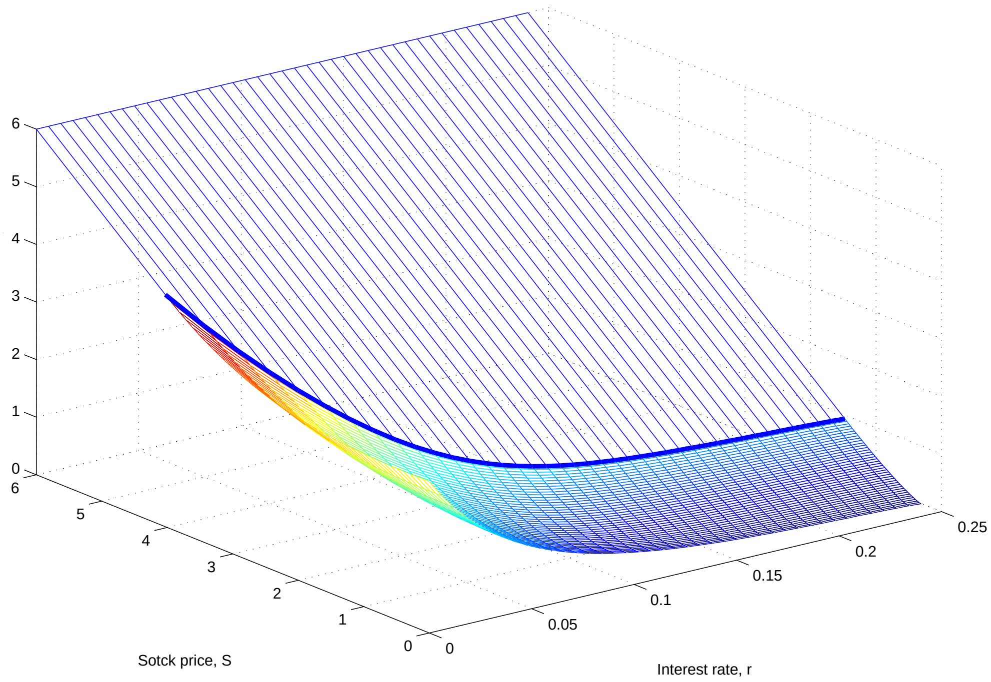

Figure 7.4: Numerical result of a two-factor convertible bond problem. The model is based on the model problem described in (2.15) and (2.16). The singularity separation technique is used to deal with the free boundary (chapter 4) and the market price of risk $\lambda(t)$ is determined by solving an inverse problem (chapter 5). The solid line shows the location of the free boundary and 30-year convertible bond price of companies with credit rating AAA is shown.

Table 7.4: Values of a convertible bond with $T = 0.5$ at $S = 1$ and $r = 0.05$. $w(r,t) = 0.26r\phi(r; r_l, r_u)$ and $u - \lambda(t)w = -0.13r + 0.008$, where $\phi(r; r_l, r_u) = 1$ if $r_l \le r \le \frac{r_u + r_l}{2}$ and $\phi(r; r_l, r_u) = \left[\frac{4(r-r_l)(r_u-r)}{(r_u-r_l)^2}\right]^{1/4}$ if $\frac{r_u+r_l}{2} < r \le r_u$, r_l being 0 and r_u being 0.30. The other parameters are $k = 0.06$, $\sigma = 0.2$, $\rho = -0.01$, $D_0 = 0.05$, $Z = 1$, and $n = 1$. The formulation (4.11) is used. A highly-accurate result for this point is 1.05985146. The code is running on an Intel(R) Core(TM) 2 CPU T7200 @2.00 GHz computer.

Meshes	Results	|Errors|	CPU (sec.)
$10 \times 10 \times 10$	1.057320	0.002531	.008
$20 \times 20 \times 20$	1.059253	0.000599	0.05

Table 7.5: Values of a convertible bond with $T = 0.5$ at $S = 1$ and $r = 0.05$. $w(r,t) = 0.26r\phi(r; r_l, r_u)$ and $u - \lambda(t)w = -0.13r + 0.008$, where $\phi(r; r_l, r_u) = 1$ if $r_l \le r \le \frac{r_u+r_l}{2}$ and $\phi(r; r_l, r_u) = \left[\frac{4(r-r_l)(r_u-r)}{(r_u-r_l)^2}\right]^{1/4}$ if $\frac{r_u+r_l}{2} < r \le r_u$, r_l being 0 and r_u being 0.30. The other parameters are $k = 0.06$, $\sigma = 0.2, \rho = -0.01, D_0 = 0.05, Z = 1$, and $n = 1$. The formulation (3.11) is used. A highly-accurate result for this point is 1.05985146. The code is running on an Intel(R) Core(TM) 2 CPU T7200 @2.00 GHz computer.

Meshes	Results	|Errors|	CPU (sec.)
$10 \times 10 \times 10$	1.027959	0.031893	.001
$20 \times 20 \times 20$	1.043968	0.015883	0.01
$40 \times 40 \times 40$	1.055042	0.003909	0.13
$80 \times 80 \times 80$	1.058987	0.000864	1.23

is 10–20 times the CPU time needed for the former or more when T is met big.

CHAPTER 8: The Code and Theoretical Prices of CB

Based on the methods described in this thesis, a user-friendly code has been written. And the code has been written into the CD attached to the thesis. This code of Calculating Convertible Bond is named as CCB. It consists of three processes: Process I — Collecting Data from the Market, Process II — Preparing the Data File, and Process III — Getting Prices of Convertible Bonds. Process I collects the raw data from market (i.e. Bloomberg) and generates a set of data files that are needed by Process II. Process II prepares the input file "dataForCCB.txt" for Process III, which contains all parameters (the ticker of the interested company, expired date, interest rate, etc) and a flag (whether a convertible bond with or without Call/Put features is calculated). By the way, those computational parameters are also added into that file. Process III reads all the information from "dataForCCB.txt", carries out all calculations, and saves the calculated prices in "outputOfCCB.txt". The market prices of convertible bonds are also listed in this file. Detailed information on the convertible bond is given in the file as well. All codes are implemented in C/C++ with a batch file named "Pricing", which simplifies all required running command in Unix/Linux system.

At the end of this Chapter theoretical prices of some convertible bonds are shown.

8.1 Collecting Data from the Market

In Process I the raw data are downloaded from Bloomberg and six sets of Excel files containing the needed data by Process II are generated. The six sets of Excel files are

1. **"CV.xls"** This file contains information on convertible bonds without call/put

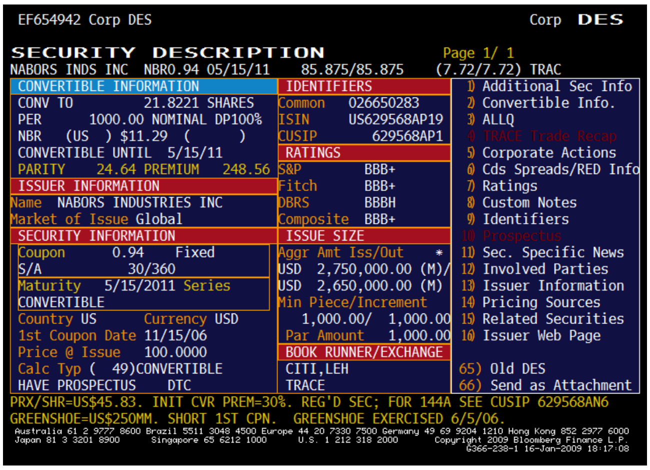

Figure 8.1: A snapshot of the Bloomberg security description for NBR. The CUSIP number is $629568AP1$.

feature and is automatically updated when we are connecting with Bloomberg service. There are four sheets in the file. We need save each of them to "CV1.csv", "CV2.csv", "CV3.csv" and "CV4.csv", then we can use the data even when we do not connect with Bloomberg.

- For each company, there is an identification number, which is called the CUSIP number because it is given by the Committee on Uniform Security Identification Procedures. For example, the CUSIP number of Nabors Industries Ltd. (NBR) is $629568AP1$. See Figure 8.1.

- In the file there are many Bloomberg functions. In the file the Bloomberg function

  BDH("Bloomberg Ticker";"PX_LAST"; "StartDate";"EndDate")

  is included. In order to download history convertible bond and stock prices, this function is needed. For example,

  BDH("629568AP1 Corp";"PX_LAST"; "2000-1-3 ";" ")

  downloads history *convertible bond prices* from Bloomberg for the company with CUSIP number $629568AP1$ (NBR) from the starting date of 2000-1-3 to the last available date. Leaving blank for the last input in BDH function means the ending date is the last available date.

BDH("NBR Equity";"PX_LAST"; "2000-1-3 ";"   ")

downloads history *stock prices* from Bloomberg for the company with CUSIP number $629568AP1$ (NBR) from the starting date of 2000-1-3 to the last available date. Leaving blank for the last input in BDH function means the ending date is the last available date.

- The Bloomberg function

  BDP("Ticker market sector ";"data item ")

  can be used to download ratio, maturity, rating (S&P) and coupon information.

The above process can be repeated for other interested companies.

2. **"CV_CP.xls"** This file contains information on convertible bonds with call/put feature. The way of getting history convertible bond and stock prices for the convertible bond with call and put feature is similar to that for the convertible bond without call and put feature, but the call and put schedules and prices are listed in a separate file called "CallPutSchedule.xls". This file is also automatically updated when we are connecting with Bloomberg service. There are four sheets in the file. We need save each of them to "CV_CP1.csv", "CV_CP2.csv", "CV_CP3.csv" and "CV_CP4.csv", then we can use the data even when we do not connect with Bloomberg;

3. **"CallPutSchedule.csv"** The call/put prices and schedules for convertible bonds with call/put feature are provided here.

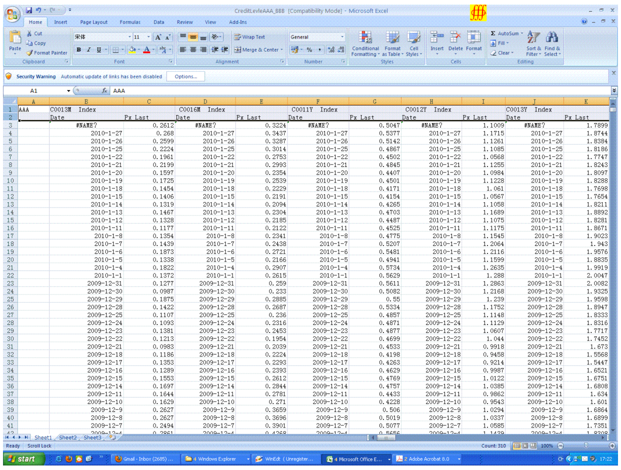

Figure 8.2: A snapshot of the file "CreditLevelAAA_BBB.csv".

4. **"DetailInformation.csv"** Supplementary information for a convertible bond either with or without call/put feature is provided here. It has the information such as the maturity date, dividend, and first date for the coupon payment.

   https://dividendinvestor.com/

   is used for getting the dividend information.

5. **"CreditLevelAAA_BBB.csv"** and **"CreditLevelBB_B.csv"** The interest rates for different credit level from AAA to B with durations of 3-month, 6-month, 1-year, 2-year, 3-year, 5-year, 10-year, 20-year and 30-year are given here (see Figure 8.2). They are gotten by saving "CreditLevleAAA_BBB.xls" and "CreditLevleBB_B.xls" that are automatically updated when we are connecting with Bloomberg service. The Bloomberg function

   BLPH("security ";"PX_LAST"; "StartDate";"EndDate")

   is used to achieve those interest rates. For example,

   BLPH("C0013M Index ";"PX_LAST";"2000-1-1 ";" ")

downloads the 3-month interest rates for credit level AAA (C0013M). Others can be downloaded via the same way.

6. **"LIBOR1M.csv"** One month LIBOR data are given here. It is gotten by saving "LIBOR1M.xls". The Bloomberg function

   BDH("US0001M INDEX ";"PX_LAST"; "1970-1-1 ";"   ")

   downloads the one month LIBOR data for US dollars.

8.2 Preparing the Data File

The code for preparing the data file consists of one main source file "data_process.cpp" and one head file "data_process.h" written in C/C++. After we have preprocessed the raw data from the market, we run the executable file "data_process.exe" as
$ ./data_process.exe
→→→ (in Unix/Linux) After reading information from "dataFile.csv" and other supplementary data files, as a result, a file "dataForCCB.txt" is generated which will be the input file of the process of getting prices of convertible bonds.

In what follows, we give some explanation on the input and output files:

1. **INPUT: "dataFile.csv" and "computation.txt"** The interested company ticker, CUSIP number and date of the the convertible bondsshould be put into "dataFile.csv" as an inputs for the executable file "data_process.exe". Besides that, some information on the method, the mesh etc. is needed to be given in "computation.txt" and the default information is shown in Table 8.1. If the default information is used, that part is not necessary to be given. the illustrative files are given as Table 8.1 and Table 8.2.

   In Table 8.1, at the beginning of the table the following are given from left to right: "modelSwitch = 1" means that one-factor model is used and a mesh size in (S, t)-space should be given and "modelSwitch = 2" means that two-factor model is used and a mesh size in (S, r, t)-space should be given. $showTime = 0$ means that the CPU time should not be printed in the final output file after computing, otherwise will be printed, and $showInformation = 0$ means that no information besides the results will be printed in the output file, otherwise the mesh size and the detailed information will be printed. Then the mesh size for determining the function $\lambda(t)$ should be given. Those information are default setup, you do not need to input and if you want to change some information, then you need to input the new information.

2. **OUTPUT: "dataForCCB.txt".** This file is the output file, and it is prepared as the input file of the process of getting prices of convertible bonds. Two illustrative files are given in Table 8.3 where the bond is a convertible bond without call/put feature and the one-factor model should be used and in Table 8.4 where the bond is a convertible bond with call/put feature and the two-factor model should be used.

Table 8.1: An illustrative file "computation.txt" which is the one part input of the process of preparing the data.

modelSwitch(1 or 2)	Mesh size ($S \times r \times t$)	showTime	showInformation
2	$40 \times 80 \times 200$	0	0
Mesh size for lambda			
200×200			

Table 8.2: An illustrative file "dataFile.xls" which is the another input of the process of preparing the data. The information on each convertible bond that should be priced should be given. The information is the company ticker, CUSIP number and date on that the bond should be priced.

COMPANY TICKER	CUSIP	DATE
NBR	629568AP1	1/27/2010
MDT	585055AK2	1/27/2010
JNJ	02261WAB5	1/27/2010

Table 8.3: An illustrative file "dataForCCB.txt" which is the output of the process of preparing the data and the input of the process of getting prices of convertible bonds.

1 80 200 1 1	B1
200 200	B2
NBR 629568AP1 BBB+	L1
1 5/15/2011 0.0 21.8221 9.4 11/15/2006 1000.00	L2
1/27/2010 23.59 0.23063 0.4849 -0.0916 987.50	L3
1.6632 1.6956 1.8423 2.3582 3.1267 4.1465 5.3158 5.7532 6.0045 6.1004	L4

8.3 Getting Prices of Convertible Bonds

The code for numerical calculation consists of one main source file "CCB.cpp " and five head files written in **C/C++**. They are:

- "CCB.cpp" accomplishes calculating of convertible bonds in three steps: (a) reading data from "dataForCCB.txt "; (b) computing λ from the yield of the bonds; (c) calculating the convertible bond prices.
- "subFunc.h" contains a group of functions, such as numerical interpolation.
- "global.h" gives a set of parameters and functions defined globally.
- "calculate.h" defines the function *OriginalPDEO2Center( )* that computes the bond prices, being called in step (b).

Table 8.4: An illustrative file "dataForCCB.txt" which is the output of the process of preparing the data and the input of the process of getting prices of convertible bonds.

2 40 80 200 0 1	B1
200 200	B2
JNJ 02261WAB5 AAA	L1
2 7/28/2020 0.031 13.7465 0.0 7/28/2020 1000.00	L2
1/27/2010 63.44 0.23063 0.2226 -0.0388 922.50	L3
0.2680 0.3437 0.5377 1.1715 1.8744 2.8308 4.3367 4.7122 5.0813 5.5220	L4
17	L5
07/28/2003 602.77	L6
07/28/2004 620.99	L7
07/28/2005 639.76	L8
07/28/2006 651.10	L9
07/28/2007 679.02	L10
07/28/2008 699.54	L11
07/28/2009 720.69	L12
07/28/2010 742.47	L13
07/28/2011 764.91	L14
07/28/2012 788.03	L15
07/28/2013 811.15	L16
07/28/2014 836.39	L17
07/28/2015 861.67	L18
07/28/2016 887.71	L19
07/28/2017 914.54	L20
07/28/2018 942.18	L21
07/28/2019 970.66	L22
3	L23
07/28/2003 602.77	L24
07/28/2008 699.54	L25
07/28/2013 811.85	L26

- "fileProcess.h" contains the function *transformfile( )* that computes the bond prices, being called in step (b).
- "convBond.h" defines the function *twoDimConvBond( )* that computes the convertible bond prices by using two-factor model and the function *Onefactor-CBIm( )* that computes the convertible bond prices by using one-factor model, being called in step (c).

Before using the code, an executable file "CCB.exe" should be generated by using the makefill. The command is

$ make

→→→ (in Unix/Linux) The "makefile" is used to compile the code for Unix/Linux system and an executable file "CCB.exe" is generated. As long as "CCB.exe" exists, in order to do the calculation, use the following command:

$ ./CCB.exe

→→→ (in Unix/Linux) reads information from "dataForCCB.txt", calculates the convertible bonds, and stores the results in "outputOfCCB.txt".

Some details on input and output are given here.

1. **INPUT** "dataForCCB.txt " is achieved from Process II containing information needed for Process III. The parameters in this file are shown in Table 8.3 for type CONV and Table 8.4 for type CONV/C/P. For each computation the first two lines give the computational parameters. If the user thinks that some computational parameters need to be changed, it should be done before doing this part.

   In Table 8.3, in the line labeled as B1 the following information is given: **model-Switch** [1] (Here "**modelSwitch** [1]" means that in this example $modelSwitch =$

Table 8.5: A result for JNJ (type CONV/C/P). The last row gives the convertible bond price and the market value for this convertible bond. On the top detailed information on the convertible bond is also shown.

The mesh (S x t) for the calculating convertible bond:									
40x80 x200									
The mesh (r x t) for solving the inverse problem:									
200 x 200									
TICKER	CUSIP	Credit Rating							
JNJ	02261WAB5	AAA							
	Expiry Date	D0	n	K	F_D_K	Z			
2	7/28/2020	0.031	13.7465	0.0	7/28/2020	1000.00			
	Pricing Date	S	r(%)	sigma	rho	Market Value			
	1/27/2010	63.44	0.23063	0.4849	-0.0916	922.50			
Bond Interest Rates with Various Maturities Used in Computation									
3m	6m	1y	2y	3y	5y	10y	15y	20y	30y
0.2680	0.3437	0.5377	1.1715	1.8744	2.8308	4.3367	4.7122	5.0813	5.5220
Convertible Bond Price					Market Value for Convertible Bond				
872.078					922.5				

Table 8.6: A result for JNJ (type CONV/C/P). The last row gives the convertible bond price and the market value for this convertible bond.

TICKER	DATE	CUSIP	CV Price	Market Price	(CV-MV)/MV(%)
JNJ	1/27/2010	02261WAB5	872.078	922.5	-5.46

1. In what follows a similar notation is used. $modelSwitch = 1$ means we choose one-factor model), the **mesh** $S{\times}t$ [40×400] (if $modelSwitch = 1$, a mesh on the (S, t)-space should be given), **showTime** [1] ($showTime = 1$ means the CPU time needs to be printed after computing, otherwise not printed), and **showInformation** [1] ($showInformation = 1$ means the mesh size and the detailed information need to be printed) from left to right; In the line labeled as B2 lists the **mesh** $r{\times}t$ [200×200] for the inverse problem; The **ticker**, **CUSIP** and **credit rating** of the company [NBR] are in the line labeled as L1; L2 lists the **Type of the bond** [1] ($Type = 1$ means CONV and $Type = 2$ means CONV/C/P), **Expire Date** [5/15/2011], **dividend** D_0 [0.0], **convert ratio** n [21.8221], **coupon** K [9.4], 1**st coupon date F_D_K** [11/15/2006], **face value** Z [1000.00] from left to right; L3 lists **date of pricing** [1/27/2010], **stock price** S [23.59], **interest rate** r [0.23063%], **sigma** σ [0.4849], **rho** ρ [-0.0916], **market value** [987.50](If we do not have the market value data, we will put -10 for market value.) from left to right; L4 lists **the ten bond interest rates** with maturities of 3 months [1.6632], 6 months [1.6956], 1 year [1.8423], 2 years [2.3582], 3 years [3.1267], 5 years [4.1465], 10 years [5.3158], 15 years [5.7532], 20 years [6.0045] and 30 years [6.1004] from left to right.

In Table 8.4, in line labeled as B1 the following information is given: **modelSwitch** [2] ($modelSwitch = 2$ means we choose two-factor model), the **mesh** $S{\times}r{\times}t$ [40×40×400], (if $modelSwitch = 2$, a mesh on the (S, r, t)-space should be given), **showTime** [0] ($showTime = 0$ means the CPU time does not need to be printed after computing), and **showInformation** [1] ($showInformaiton = 1$ means the mesh size and the detailed information need to be printed) from left to right; B2 lists the **mesh** $r{\times}t$ [200× 200] for the inverse problem; The ticker, CUSIP and rating of the company [JNJ] are in L1; L2 lists the **Type of the bond** [2] ($Type = 1$ means CONV and $Type = 2$ means CONV/C/P), **Expire**

Date [7/28/2020], **dividend** D_0 [0.031] **convert ratio** n [13.7465], **coupon** K [0.0], **1st coupon date F_D_K** [7/28/2020], **face value** Z [1000.0] from left to right; L3 lists **Date of Pricing** [01/27/2010], **stock price** S [63.44], **interest rate** r [0.23063%], **sigma** σ [0.2226], **rho** ρ [-0.0388], **market value** [922.5](If we do not have the market value data, we will put -10 for market value.); L4 lists the **ten bond interest rates** with maturities of 3 months [0.2680], 6 months [0.3437], 1 year [0.5377], 2 years [1.1715], 3 years [1.8744], 5 years [2.8308], 10 years [4.3367], 15 years [4.7122], 20 years [5.0813] and 30 years [5.5220] from left to right; L5 lists the **total number of calls**; in the lines below L5 which are L6–L22 in this case, each line lists a **call date and call price**; The next line which is L23 in this case lists the **total number of puts**; and the last few lines which are L24–L26 in this case list **put dates and put prices**.

2. **OUTPUT** "outputOfCCB.txt" is the final output file of the program. The price of the convertible bonds of each corporation in "dataForCCB.txt "is listed here. Table 8.5 shows the result for JNJ (for its input, see Table 8.4). If in Table 8.4 $showInformation = 0$, the output will like what is given in Table 8.6.

8.4 The batch file

As we discussed in Section 8.1, 8.2 and 8.3, after collecting the raw data from the market(i.e. Bloomberg), the theoretical prices of convertible bonds on the market are achieved though two process: a) Preparing the data file and b) Getting prices of convertible bonds. There is one batch file named "Pricing" in the attached CD which simplifies all required running commands in Unix/Linux system.

8.5 Theoretical Prices of Convertible Bonds on the Market

8.5.1 Market data of convertible bonds

In this section, we collect some market data to test our model. Most of the data are quoted from Bloomberg. The data about D_0 is getting from https://dividendinvestor.com/. And the volatility of the stock price σ and correlation ρ are calculated by using historical data from the pricing date to the previous six years. We collected forty-one convertible bonds with different ratings issued by S&P at August 13, 2010 as our objects. In the thesis we list seventeen of them to show and eight of them are convertible bonds with call and put features, others are just convertible bonds. Table 8.7 gives the information about the bonds we choose. And Table 8.8 shows the detail of the data needed. When we compute the convertible bond with call and put features(CONV/C/P), we also need the call prices, the callable dates, the put prices and the puttable dates. The bonds are callable on and any time after date(s) with the price(s). And the put schedule is discrete put. Those data are shown in Table 8.9.

8.5.2 Numerical evaluated prices of convertible bonds

The market price of risk $\lambda(t)$ of different credit ratings has been calculated by the model problem introduced in chapter 5, and we could get the numerical evaluated prices of convertible bonds of those twenty companies using our model. We also modify the credit rating depending on how the stock price changing in the previous 30 days. If the stock price decreases more than 20% , then the rating will be moved down by 2 levels. If the stock price decreases more than 15% and less than 20%, then the rating will be moved down by 1 level. Otherwise the rating will not be changed. The Table 8.10 gives overviews of comparisons between market prices, two-factor model numerical evaluated prices and the comparison between the theoretical price and the market price at August 13, 2010. The average absolute error for the total 41 objects collected is 6.92%. If we choose the one-factor model, then the average

Table 8.7: The information about the ticker name of the company who issued the convertible bond, credit rating, and the type of bond.

Ticker	CUSIP	*Rating*	Type
MDT	$585055AK2$	$AA-$	$CONV$
DDR	$251591AR4$	BB	$CONV$
DRE	$26441YAN7$	$BBB-$	$CONV$
NBR	$629568AP1$	$BBB+$	$CONV$
TAP	$60871RAA8$	$BBB-$	$CONV$
ATK	$018804AM6$	$BB-$	$CONV$
ADM	$039483AW2$	A	$CONV$
$AMGN$	$031162ANO$	$A+$	$CONV$
IGT	$459902AQ5$	BBB	$CONV$
BEC	$075811AD1$	$BB+$	$CONV/C/P$
AMG	$008252AL2$	$BBB-$	$CONV/C/P$
CAM	$13342BAB1$	BBB	$CONV/C/P$
HCN	$42217KAP1$	$BBB-$	$CONV/C/P$
LLL	$502413AW7$	$BB+$	$CONV/C/P$
PLD	$743410AQ5$	$BBB-$	$CONV/C/P$
$TECD$	$878237AE6$	$BBB-$	$CONV/C/P$
VNO	$929043AC1$	BBB	$CONV/C/P$

Table 8.8: For eighteen bonds, the ticker name of the issuing company, the stock price S, convert ratio n, dividend D_0, correlation ρ, volatility of the stock price σ, semi-annual coupon rate k, and maturity are listed. The face value is \$1000 for all bonds and interested rate r is 0.27188% at August 13, 2010.

Ticker	S	n	D_0	ρ	σ	k	$Maturity$
MDT	35.57	18.2508	0.019	−0.0792	0.2578	16.25	4/15/2011
DDR	10.59	13.3783	0.09	0.0227	0.8269	30	3/15/2012
DRE	11.16	20.4298	0.059	−0.0901	0.6569	37.5	12/1/2011
NBR	16.46	21.8221	0.0	−0.0986	0.4960	9.4	5/15/2011
TAP	45.62	18.5154	0.023	−0.0982	0.2796	25	7/30/2013
ATK	69.41	10.3617	0.0	−0.1027	0.2302	27.5	9/15/2011
ADM	30.05	22.8343	0.018	−0.0933	0.4006	8.75	2/14/2014
$AMGN$	55.02	12.5247	0.0002	−0.07066	0.2946	1.25	2/1/2011
IGT	15.22	50.0808	0.013	−0.1080	0.4479	32.5	5/1/2014
BEC	45.55	13.4748	0.011	−0.0476	0.2976	25	10/15/2036
AMG	68.25	7.9586	0.0	−0.0807	0.4772	39.5	8/15/2038
CAM	25.93	28.2656	0.0	−0.0556	0.4618	25	6/15/2026
HCN	44.72	20.9132	0.064	−0.0373	0.3131	47.5	12/1/2026
LLL	70.26	10.1074	0.016	−0.0328	0.3318	30	8/1/2035
PLD	10.33	13.1614	0.046	−0.0319	0.6192	22.5	4/1/2037
$TECD$	36.75	18.431	0.0	−0.0634	0.4170	27.5	12/15/2026
VNO	81	11.4718	0.04	−0.0693	0.4011	38.75	4/15/2025

Table 8.9: Call and put schedules for the convertible bonds with call and put features.

Ticker	*Call − Date*	*Price*	*Put − Date*	*Price*
BEC	10/20/2013	1000.00	12/15/2013	1000.00
			12/15/2016	1000.00
			12/15/2021	1000.00
			12/16/2026	1000.00
			12/16/2031	1000.00
AMG	8/15/2013	1000	8/15/2013	1000.00
			8/15/2018	1000.00
			8/15/2028	1000.00
			8/15/2033	1000.00
CAM	06/20/2011	1000.00	06/11/2011	1000.00
			06/11/2016	1000.00
			06/11/2021	1000.00
HCN	12/1/2011	1000	12/1/2011	1000.00
			12/1/2016	1000.00
			12/1/2021	1000.00
LLL	2/1/2011	1000.00	2/1/2011	1000.00
			2/1/2016	1000.00
			2/1/2021	1000.00
			2/1/2026	1000.00
			2/1/2031	1000.00
PLD	4/5/2012	1000.00	4/1/2012	1000.00
			4/1/2017	1000.00
			4/1/2022	1000.00
			4/1/2027	1000.00
			4/1/2032	1000.00
TECD	4/5/2012	1000.00	4/1/2012	1000.00
			4/1/2017	1000.00
			4/1/2022	1000.00
VNO	4/18/2012	1000.00	4/15/2012	1000.00
			4/15/2015	1000.00
			4/15/2020	1000.00

Table 8.10: Market convertible bond prices and two-factor model numerical evaluated prices at August 13, 2010.

Ticker	Numerical evaluated price	Market price	$(CV - MV)/MV(\%)$
MDT	1014.6903	1003.59	1.10
DDR	916.7834	961.365	-4.63
DRE	1029.6842	1002.76	2.68
NBR	1001.7980	1010	-0.81
TAP	1082.7114	1090	-0.66
ATK	1024.0789	995.525	2.86
ADM	1110.7087	1010	9.97
$AMGN$	1001.5086	996.75	0.47
IGT	1208.8406	1108.23	9.07
BEC	991.4719	1020	-2.79
AMG	1203.4769	995	20.95
CAM	1235.2088	1215	1.663
HCN	1144.0753	1050	8.95
LLL	1002.2489	1002.5	-0.02
PLD	964.1913	960	0.43
$TECD$	1064.4536	1013.75	5.0
VNO	1230.5950	1125	9.38

absolute error is 12.82%. We can find that the theoretical prices of the two-factor model are better than the one-factor model.

For convertible bonds without call/put provision with $T > 5$ and a very small spot interest rate, the results have some oscillations in the two-factor model. For this case, a further research is needed in order to improve the results.

CHAPTER 9: CONCLUSIONS

In this paper, a two-factor model for pricing convertible bonds has been provided, a numerical method for evaluating convertible bond price by using this model is described and some results on convertible bonds are also shown.

The two factors are the stock price of the company and the interest rate. In the model the market price of risk for the interest rate is determined by the market data through solving an inverse problem. Thus the interest rate model always gives the correct price of any zero-coupon bond. When evaluating the convertible bonds issued by companies with different credit ratings, different zero-coupon bond curves are used. In this way, the credit of the company is considered in this model. Because of these facts, the price of convertible bonds obtained by this model is more reasonable.

In order to price the convertible bond by this model, we need to solve a linear complementarity(LC) problem, in which there is a free boundary. Thus usually a closed-form solution does not exist and numerical methods are adopted to price the convertible bonds by the model mentioned. On the free boundary the second derivative of the solution is discontinuous. The first derivative of the payoff function at the point $\frac{Z}{n}$, Z being the face value and n being the conversion ratio, is also discontinuous. These problems cause some relatively large errors. To solve these problems, we formulate this LC problem as a free-boundary problem. In this case, because the location of the free boundary is obtained, we can easily know in which case the convertible bond should be exercised. (This is important in practice.) To weaken the singularity of the solution at $t = T$ and $\frac{Z}{n}$, we construct an analytic solution for a similar equation and the same final condition. Because the price of the convertible bond and the analytic solution have similar singularities, the difference between the two solutions has a weaker singularity. As the singularity of the solution obtained by numerical method is weaker, we can find quite good results even by using a coarse mesh.

A user-friendly code has been written for such a method. Taking the market data as input, we can easily, quickly, and reasonably give the price of a convertible bond. Many results of convertible bonds are shown in this paper. This code can be used in practice to evaluate the price of convertible bonds issued by companies with different credit ratings except the convertible bonds with $T > 5$ and with a spot interest rate close to zero.

REFERENCES

[1] F. Black and M. Scholes. The pricing of options and corporate liabilities. *Journal of Political Economy*, 81:637–654, 1973.

[2] M. Brennan and E. Schwartz. Analyzing convertible bonds. *Journal of Financial and Quantitative Analysis*, 15(4):907–929, 1980.

[3] V. Druskin, L. Knizhnerman, T. Tamarchenko, and S. Kostek. Krylov subspace reduction and its extensions for option pricing. *Journal of Computational Finance*, 1:63–79, 1997.

[4] I. Gihman and A. Skorohod. *Stochastic differential equations*. Springer-Verlag, 1972.

[5] J. C. Hull. *Options, Futures and Other Derivatives*. Prentice Hall, 2006.

[6] F. Longstaff and E. Schwartz. Valuing risky debt: a new approach. *Journal of Finance*, 50:789–819, 1995.

[7] R. Merton. Theory of rational option pricing. *Bell Journal of Finance and Management Science*, 4(1):141–184, 1973.

[8] Y. Sun. *High Order Methods for Evaluateing Convertible Bonds*. PhD thesis, University of North Carolina at Charlotte, 1999.

[9] O. Vasicek. An equilibrium characterisation of the term structure. *Journal of Financial Economics*, 5:177–188, 1977.

[10] J. O. Wever, P. P. Smid, and R. H. Koning. Pricing of convertible bonds with hard call features. *SOM-theme E: Financial Markets and Institutions*.

[11] P. Wilmott, J. Dewynne, and S. Howison. *Option pricing, mathematical models and computation*. Oxford Financial Press, 1993.

[12] Y.-L. Zhu and J. Li. Multi-factor financial derivatives on finite domian. communications in mathematical sciences. *Communications in Mathematical Sciences*, 1:343–359, 2003.

[13] Y.-L. Zhu and Y. Sun. The singularity-separating method for two-factor convertible bonds. *Journal of Computational Finance*, 3(1):91–110, 1999.

[14] Y.-L. Zhu, X. Wu, and I.-L. Chern. *Derivative Securities and Difference Methods*. Springer, 2004.