Pricing Convertible Bonds with Dividend Protection subject to Credit Risk Using a Numerical PDE Approach

by

Qingkai Mo

A thesis submitted in conformity with the requirements
for the degree of Master of Science
Graduate Department of Computer Science
University of Toronto

Abstract

Pricing Convertible Bonds with Dividend Protection subject to Credit Risk Using a Numerical PDE Approach

Qingkai Mo

Master of Science

Graduate Department of Computer Science

University of Toronto

2006

We develop a pricing model for convertible bonds with dividend protection subject to credit risk by extending the models developed by Tsiveriotis and Fernandes (TF), and by Ayache, Forsyth and Vetzal (AFV). We consider two techniques to incorporate the dividend protection feature: *Conversion Ratio Adjustment* and *Dividend Pass-Thru*. We apply finite difference methods to discretize the PDEs associated with our models, and study the Projected Successive Over-Relaxation and penalty methods to handle the free boundaries. We compare these two methods in terms of convergence rate, number of iterations per time step and computation time for pricing convertible bonds without dividends. Finally we apply the penalty method, the better of the two methods, to solve the systems arising from our models for convertible bonds with dividend protection. We examine the convergence rates and discuss the difference between the results from the extended TF and AFV models, with both dividend protection techniques.

Acknowledgements

This research was supported in part by the Department of Computer Science at the University of Toronto and by OGS (Ontario Graduate Scholarship). I would like to thank Dr. Izzy Nelken for bringing this interesting research topic in Computational Finance to our attention. I would also like to thank my supervisors, Professor Christina Christara and Professor Ken Jackson, for their guidance and helpful comments throughout the research. Finally, I would like to express my deep gratitude to my wife for her unconditional support and endless love.

Contents

List of Tables

List of Figures

Chapter 1

Introduction and Literature Survey

A convertible bond is a standard corporate bond that usually pays regular coupons and principal, but has an additional conversion feature. The issuer of the convertible bond grants the holder the option to convert the bond into a predetermined amount of the issuing company's stock at certain times in the future. Because of this feature, the convertible bond incorporates elements of both debt and equity. A convertible bond often has some additional features such as callability (*i.e.*, the issuer has the right to buy the bond back at certain times at some predetermined price) and puttability (*i.e.*, the holder has the right to sell the bond back to the issuer at certain times at a particular price). Besides these features, a convertible bond may also have some exotic and complicated features, such as trigger prices and "soft call" provisions. All of these features make it complicated to determine a fair price for a convertible bond.

The convertible bond market is expanding rapidly. In the U.S., in 2000, just over $60 billion of new convertible bonds were issued, whereas over $105 billion were issued in 2001. The global market for convertible bonds exceeded $500 billion in 2002 [1]. There are two frequently cited rationales on why firms issue convertible bonds. First, issuing convertible bonds is an indirect means to add equity to the capital structure, since convertible bonds can be regarded as a form of delayed equity. Second, convertible

bonds allow a firm to borrow more cheaply relative to straight bonds in the sense that the coupons on convertible bonds are normally lower than those on comparable straight bonds [21].

In recent years, a new convertible bond feature, dividend protection, has emerged. If the issuing company increases its dividend on the common stock, the stock price will decrease and consequently the conversion value of the convertible bond will decrease. To keep the convertible bonds attractive in this case, a so-called dividend protection provision is added to the contract, giving convertible bond holders insurance against the decline of the issuing company's stock caused by an increase in the dividend: if the issuing company increases its dividend on the common stock, the convertible bond will be adjusted to become convertible to more shares or the convertible bond holders will receive a cash compensation proportional to the increase in dividend amount. Over the past several years, convertible bonds with dividend protection have grown in popularity. As stated in [13], in the U.S., in 2004, 90% of the proceeds and 87% of new convertible securities had some form of dividend protection, compared to 45% and 43%, respectively, in 2003 and 14% and 11%, respectively, in 2002.

As a convertible bond is a hybrid security of equity and debt, it is exposed to three sources of risk: equity risk, interest rate risk and credit risk. The modeling of the inherent credit risk has been one of the major focuses of convertible bond pricing. The modern academic literature on convertible bond valuation was pioneered by the papers of Ingersoll [14] and Brennan and Schwartz [3] [4]. These models are based on Merton's structural approach to modeling default. In this approach, the basic underlying variable is the value of the issuing firm. The firm's debt and equity are claims contingent on the firm's value, and options on its debt and equity are compound options on this variable. The asset value of the firm is modeled using geometric Brownian motion. Default occurs when the firm's value becomes sufficiently low that it is unable to meet its financial obligations. Under the assumption of no dividends and no coupons, Ingersoll developed closed form

solutions for the convertible bond price. Brennan and Schwartz [3] extended Ingersoll's model by including discrete dividends, and applied finite difference methods to solve the partial differential equation (PDE) for the price of the convertible bond. Later, Brennan and Schwartz [4] extended their work to a two-factor stochastic model describing both the value of the firm and the interest rate. However, they found that a stochastic term structure had little impact on the convertible bond price.

Since the firm value is not directly observable in the market, recent research on convertible bond pricing has moved away from modeling the firm value and towards the modeling of equity. The equity value approach prices convertible bonds based on the equity value instead of on the firm value, and models equity as a geometric Brownian motion. The first equity-based models were faced with the great difficulty of modeling credit risk. Early papers with an *ad hoc* approach to discounting cash flows generated by convertible bonds include those of McConnell and Schwartz [18], Cheung and Nelken [5], and Ho and Pfeffer [11]. Many of these models do not explicitly model bankruptcy, but instead uniformly apply a somewhat arbitrary risky spread to the risk-free discount rate to model credit risk. More recent papers recognize that equity and debt components of convertible bonds are subject to different default risk. Tsiveriotis and Fernandes (TF) [25] effectively split the convertible bond into two separate components: a cash-only component and an equity component. The cash-only component receives bond cash flows and is discounted using a risky rate. The equity component receives equity flows and is discounted using a risk-free rate. The evolution of each component is described by a partial differential equation. These equations are coupled through their respective boundary conditions. A related approach was promoted by Goldman Sachs [23] and involves careful weighting of risky and risk-free discounting in a binomial lattice.

Along with the development of convertible bond models, there have been significant developments in credit risk modeling. In particular, Jarrow and Turnbull [15] introduced the reduced form approach to modeling credit risk. This approach models default as

the first jump in a Poisson process. Upon default, the stock price jumps to zero (or a very low value) and a certain percentage of the bond is assumed to be recovered. In recent work, Ayache, Forsyth and Vetzal (AFV) [1] extend the reduced form approach to model credit risk associated with convertible bonds, and detail exactly what happens on default. AFV argue that many of the existing models, such as the TF model [25], are incomplete because they do not explicitly specify what happens in the event of a default by the issuing company, and this can result in internal inconsistencies. AFV also solve the PDEs associated with their model by finite difference methods rather than using slow binomial or trinomial trees.

The objective of this thesis is to valuate convertible bonds with dividend protection subject to credit risk. Our task involves two main parts: modeling and implementation. First, we will consider two existing models, namely the TF and AFV models, and adjust them to incorporate the dividend protection feature. The reason for choosing these two models is that they are the two most commonly used models in practice and they model credit risk in different ways. Second, we will apply finite difference methods to discretize the PDEs associated with our models and find a fast and robust numerical algorithm to solve the resulting systems.

The organization of this thesis is as follows. Chapter 2 introduces the TF and AFV models for pricing convertible bonds, and describes our proposed approach to valuating convertible bonds with dividend protection based on these two models. Chapter 3 presents finite difference discretization of the PDEs associated with our proposed approach. Chapter 4 first explores two numerical methods for handling the free boundary associated with a convertible bond without dividends under the TF model: the Projected Successive Over-Relaxation (PSOR) and the penalty methods. Then the better of the two methods is used to solve the systems arising from our models of a convertible bond with dividend protection derived from the TF and AFV models, respectively. Chapter 5 presents the numerical results from the better method determined in Chapter 4,

and compares the results from our extension of the TF and AFV models for the following cases: a convertible bond without dividends, a convertible bond with dividends but without dividend protection, and a convertible bond with dividends and with dividend protection. Chapter 6 contains our conclusions and a discussion of some possible future work.

Chapter 2

Modeling

Convertible bond modeling is complicated for several reasons. One reason is that convertibles have many possible features, such as conversion, callability and puttability, which must be taken into account for pricing. Correct modeling of these features increases the complexity of the model. Another reason is that convertible bonds are exposed to three sources of risk: equity risk, credit risk and interest rate risk. The modeling of credit risk has been a major source of difficulty in convertible bond modeling. In the literature, many approaches to pricing convertible bonds subject to credit risk have been proposed. In this research thesis, we consider two commonly used models: the one developed by Tsiveriotis and Fernandes (TF) [25], and the one developed by Ayache, Forsyth, and Vetzal (AFV) [1]. We adjust these models to incorporate the dividend protection feature.

2.1 Models for Pricing Convertible Bonds with Credit Risk

2.1.1 TF Model

The core idea behind the TF model is that the convertible bond (CB) consists of an equity and a bond component, and these components have different default risks, so credit risk is incorporated differently in each component when pricing CBs. The underlying equity has zero default risk because it is assumed that the issuing company can always deliver its own stock. Whereas, coupon and principal payments, and any put provisions which allow the holder to sell the CB back to the issuer, depend on the issuer's timely access to the required cash amounts and thus bear credit risk. This credit risk is modeled by the use of the market-observed credit spread between a bond from the same issuing company that is not convertible and a risk-free government bond. Therefore, in the TF model, the value of the CB is split into two parts: the "Cash-Only" part of the CB (COCB) and the equity part. The holder of a COCB is entitled to all cash flows, and no equity flows, that an optimally behaving holder of the corresponding CB would receive. The COCB is discounted at a risky rate. On the other hand, the equity part represents the value of the CB related to payments in equity and is discounted at the risk-free rate. The above splitting results in a system of two coupled Black-Scholes-like PDEs. For simplicity, TF use a single-factor model, *i.e.*, a CB is viewed as an equity-only derivative. Moreover, the COCB itself is considered to be a derivative of the underlying equity. The two coupled Black-Scholes-like equations are

$$\frac{\partial U}{\partial t} + \frac{\sigma^2 S^2}{2}\frac{\partial^2 U}{\partial S^2} + r_g S\frac{\partial U}{\partial S} - r(U - B) - (r + r_c)B + f(t) = 0 \tag{2.1}$$

and

$$\frac{\partial B}{\partial t} + \frac{\sigma^2 S^2}{2}\frac{\partial^2 B}{\partial S^2} + r_g S\frac{\partial B}{\partial S} - (r + r_c)B + f(t) = 0, \tag{2.2}$$

where U is the value of the CB, B the value of the COCB, S the price of the underlying stock, r the risk-free interest rate, r_g the growth rate of the stock, which, according to the discussion of the TF model in [12], can be regarded as the risk free rate, r_c the observed credit spread implied by the straight bonds of the same issuer for similar maturities as the CB, and $f(t)$ the external flows (in cash or equity) to the derivative. For example, for a bond paying a coupon of c_j at time t_j, we have

$$f(t) = \sum c_j \delta(t - t_j),$$

where δ is the Dirac function. We will ignore $f(t)$ for now; the handling of external flows is discussed in Section 2.2.3.

Equations (2.1) and (2.2) are coupled through appropriate final and boundary conditions reflecting the terms of each individual CB, and need to be solved simultaneously. For example, assume that the CB has the face value of F, pays a coupon payment of $c_j = K$ semiannually and matures at time T. The CB can be converted into κ shares of the company's stock at any time during the life of the CB, is callable by the issuer at a price B_c at any time after T_c, and is puttable by the holder at a price B_p at any time after T_p. Then the corresponding final and boundary conditions are as follows:

- Final conditions at maturity (assuming put and call will not occur at maturity):

$$U(S,T) = \begin{cases} F + K & \text{if } F + K \geq \kappa S \\ \kappa S & \text{otherwise,} \end{cases}$$

$$B(S,T) = \begin{cases} F + K & \text{if } F + K \geq \kappa S \\ 0 & \text{otherwise.} \end{cases} \tag{2.3}$$

- Upside constraints due to conversion for $t \in [0,T]$:

$$\begin{aligned} &U(S,t) \geq \kappa S, \\ &B(S,t) = 0, \quad \text{if } \hat{U} \leq \kappa S, \end{aligned} \tag{2.4}$$

where $\hat{U}$ is the "continuous value" of the CB when none of the events of conversion, puttability, and callability occurs. We use $\hat{U}$ in a similar way in the following constraints.

- Upside constraints due to callability by the CB issuer for $t \in [T_c, T]$:

$$\begin{aligned} U(S,t) &\leq \max(B_c, \kappa S), \\ B(S,t) &= 0, \quad \text{if } \hat{U} \geq B_c, \end{aligned} \tag{2.5}$$

  where it is assumed that the holder has the right to convert if the issuer calls the CB.

- Downside constraints due to puttability by the CB holder for $t \in [T_p, T]$:

$$\begin{aligned} U(S,t) &\geq B_p, \\ B(S,t) &= B_p, \quad \text{if } \hat{U} \leq B_p. \end{aligned} \tag{2.6}$$

Notice that in the TF model, the handling of the COCB value (B) when a call occurs is different from when a put occurs. The rationale behind this is that when a call occurs, the issuing company must have the required amount of money on hand to buy back the CB, so this payment does not have default risk and thus B should be equal to 0. On the other hand, when a put occurs, *i.e.*, the holder of the CB sells the CB back to the issuer, the issuer may face financial difficulties and may not have timely access to the required cash amounts, so this payment bears default risk and thus B should be set to the put price B_p.

2.1.2 AFV Model

Ayache, Forsyth and Vetzal [1] extend the reduced-form approach and derive a different model, based on a hedging portfolio, where the risk due to the stochastic process followed by the stock price is eliminated, and assuming a Poisson default process. They use a deterministic hazard rate $p(S,t)$ to model the credit risk: the probability of default in

$[t, t+dt]$ conditional on no-default in $[0,t]$ is $p(S,t)dt$. The AFV model provides a general single-factor framework for valuing risky convertible bonds. It accommodates many different assumptions about the behavior of the stock price in the case of default and recovery after default.

The AFV model assumes that, upon default, the stock price jumps according to

$$S^+ = S^-(1-\eta),$$

where S^+ is the stock price immediately after default, S^- is the stock price right before default, and $0 \leq \eta \leq 1$. Further, upon default, the holder of the convertible bond can choose to receive either the amount RX (where $0 \leq R \leq 1$ is the recovery factor, and there are many possible assumptions for X, e.g. the face value of bond, discounted bond cash flows, or the pre-default value of the bond portion of the convertible bond) or shares worth $\kappa S^-(1-\eta)$ (where κ is the conversion ratio).

Under these assumptions and defining

$$\mathcal{M}U \equiv -\frac{\partial U}{\partial t} - (\frac{\sigma^2 S^2}{2}\frac{\partial^2 U}{\partial S^2} + (r+p\eta)S\frac{\partial U}{\partial S} - (r+p)U), \tag{2.7}$$

the value of the convertible bond U satisfies:

- if $B_c > \kappa S$:

$$\begin{pmatrix} \mathcal{M}U - p\ \max(\kappa S(1-\eta), RX) = 0 \\ U - \max(B_p, \kappa S) \geq 0 \\ U - B_c \leq 0 \end{pmatrix}$$

$$\vee \begin{pmatrix} \mathcal{M}U - p\ \max(\kappa S(1-\eta), RX) \geq 0 \\ U - \max(B_p, \kappa S) = 0 \\ U - B_c \leq 0 \end{pmatrix}$$

$$\vee \begin{pmatrix} \mathcal{M}U - p\ \max(\kappa S(1-\eta), RX) \leq 0 \\ U - \max(B_p, \kappa S) \geq 0 \\ U - B_c = 0 \end{pmatrix}, \tag{2.8}$$

where the notation $(x = 0) \vee (y = 0) \vee (z = 0)$ means that at least one of $x = 0, y = 0, z = 0$ holds at each point in the solution domain, and p is the hazard rate;

- if $B_c \leq \kappa S$:

$$U = \kappa S. \tag{2.9}$$

Several possible assumptions can be made about X. Assuming X to be B, the pre-default bond component of the convertible bond, the AFV model provides a new decomposition of the convertible bond into bond and equity components so that $U = B + C$, where B is the value of the bond component and C is the value of the equity component. According to [1], the PDEs and free boundary constraints associated with this decomposition are

$$\begin{pmatrix} \mathcal{M}C - p\ \max(\kappa S(1-\eta) - RB, 0) = 0 \\ C - (\max(B_c, \kappa S) - B) \leq 0 \\ C - (\kappa S - B) \geq 0 \end{pmatrix}$$

$$\vee \begin{pmatrix} \mathcal{M}C - p\ \max(\kappa S(1-\eta) - RB, 0) \leq 0 \\ C = \max(B_c, \kappa S) - B \end{pmatrix}$$

$$\vee \begin{pmatrix} \mathcal{M}C - p\ \max(\kappa S(1-\eta) - RB, 0) \geq 0 \\ C = \kappa S - B \end{pmatrix}, \tag{2.10}$$

and

$$\begin{pmatrix} \mathcal{M}B - RpB = 0 \\ B - B_c \leq 0 \\ B - (B_p - C) \geq 0 \end{pmatrix} \vee \begin{pmatrix} \mathcal{M}B - RpB \leq 0 \\ B = B_c \end{pmatrix} \vee \begin{pmatrix} \mathcal{M}B - RpB \geq 0 \\ B = B_p - C \end{pmatrix}. \tag{2.11}$$

The final conditions at maturity for U, C and B, respectively, are:

$$U(S,T) = \max(\kappa S, F + K), \tag{2.12}$$

$$C(S,T) = \max(\kappa S - F - K, 0), \tag{2.13}$$

and

$$B(S,T) = F + K. \tag{2.14}$$

Notice that the difference between the AFV and the TF splittings is that, for the former, only $B \leq B_c$ is required, while for the latter, $B = 0$ when $U = \kappa S$, and $C = 0$ when $U = B_p$ are required.

2.1.3 Connection between the TF and AFV Models

Although the TF and AFV models are different, there is some connection between them. As stated in [1], we can view the TF model as a partial default model (the stock price does not jump upon default, and thus $\eta = 0$) since the equity part of the convertible bond is discounted at the risk-free rate (see Equation (2.1)). More specifically, setting $\eta = 0$ and $X = B$, the PDEs governing U and B under the AFV model become

$$\frac{\partial U}{\partial t} + \frac{\sigma^2 S^2}{2}\frac{\partial^2 U}{\partial S^2} + rS\frac{\partial U}{\partial S} - (r+p)U + p\max(\kappa S, RB) = 0, \tag{2.15}$$

$$\frac{\partial B}{\partial t} + \frac{\sigma^2 S^2}{2}\frac{\partial^2 B}{\partial S^2} + rS\frac{\partial B}{\partial S} - (r + p(1-R))B = 0. \tag{2.16}$$

Comparing Equations (2.2) and (2.16) (ignoring $f(t)$), we can see that they are exactly the same if $r_c = p(1-R)$. However, this does not mean that if $r_c = p(1-R)$, the TF model becomes the partial default AFV model. Setting $r_c = p(1-R)$, the PDE governing the value of U under the TF model in Equation (2.1) becomes (ignoring $f(t)$)

$$\frac{\partial U}{\partial t} + \frac{\sigma^2 S^2}{2}\frac{\partial^2 U}{\partial S^2} + rS\frac{\partial U}{\partial S} - rU - p(1-R)B = 0. \tag{2.17}$$

It is easy to see that, in general, (2.17) is not equivalent to (2.15). In Chapter 5, we will use the connection between the TF and AFV models outlined above to compare the numerical results obtained from these two models.

2.2 Our Proposed Approach for Pricing Convertible Bonds with Dividend Protection

We know of two types of dividend protection [13]: *Conversion Ratio Adjustment*, which promises to adjust the conversion ratio when the dividend on the underlying common stock is changed, and *Dividend Pass-Thru*, which promises to pass on increases in dividend amount to the convertible bond holders. *Conversion Ratio Adjustment* constitutes the overwhelming majority (over 99%) of convertible bond dividend protection cases [13]. Our proposed approach extends either the TF model or the AFV model described above to incorporate either of these two types of dividend protection. In our approach, we assume the interest rate is deterministic and the stock price is the only underlying variable, because, according to [10], the random nature of the spot interest rate is of second-order importance. Moreover, we assume that dividend payments on the common stock are paid discretely, which is the standard practice. The handling of discrete dividend payments is discussed in detail in Section 2.2.3. In the following sections, we discuss our models for *Conversion Ratio Adjustment* and *Dividend Pass-Thru* in detail.

2.2.1 *Conversion Ratio Adjustment*

Assume the stock price at $t = t_0 = 0$ is S_{int}, the underlying stock pays dividend D_i for the time period $[t_i, t_{i+1}]$, $i = 0, 1, \ldots, d$, where $t_{d+1} = T$. The conversion ratio at the time of issue ($t = 0$) is κ_0. For our extensions of the TF and AVF models, the *Conversion Ratio Adjustment* comes into play only at the final and boundary conditions for the associated PDEs. Motivated by the idea presented in [13], we adjust the conversion ratio as follows: when deciding whether or not the convertible bond should be (voluntarily or forcibly) converted into stock, the conversion ratio for the time period $[t_i, t_{i+1}]$ (for $i = 1, \ldots, d$) is changed, according to

$$\kappa_i = \kappa_0 \frac{S_{int}}{S_{int} - (D_i - D_0)_+} \quad , \tag{2.18}$$

where the notation $(\cdot)_+$ means

$$(x)_+ = \begin{cases} x & \text{if } x > 0 \\ 0 & \text{otherwise.} \end{cases}$$

More specifically, we adjust the TF and AFV models respectively to incorporate the *Conversion Ratio Adjustment* as follows.

- For the TF model:

  Apply Equations (2.1) and (2.2) to govern the evolution of the value of the convertible bond (U) and the value of COCB (B), respectively, for $t \in [0, T]$. Apply the final/boundary conditions (Equations (2.3) – (2.6)) using appropriate conversion ratios for different time periods, i.e., κ_0 for $[0, t_1]$ and κ_i for $[t_i, t_{i+1}]$ computed by Equation (2.18) for $i = 1, \ldots, d$.

- For the AFV model:

  Apply Equations (2.7) – (2.14) to the convertible bond for $t \in [0, T]$. Apply the final/boundary conditions using appropriate conversion ratios for different time periods, i.e., κ_0 for $[0, t_1]$ and κ_i for $[t_i, t_{i+1}]$ computed by Equation (2.18) for $i = 1, \ldots, d$.

2.2.2 *Dividend Pass-Thru*

For the *Dividend Pass-Thru*, the issuing company passes the excess dividend payouts onto the convertible bond holder. The excess dividend payouts to the convertible bond holder can be viewed as coupon payments which are paid at each dividend payment date for which the new dividend is greater than the initial dividend.

Assume the same dividend scheme as the one in Section 2.2.1. The excess dividend payout K_i' to the holder of one convertible bond at the dividend payment date t_i (for $i = 1, \ldots, d$) is determined by

$$K_i' = \kappa(D_i - D_0)_+ \quad , \tag{2.19}$$

where κ is the conversion ratio. Therefore, to incorporate the *Dividend Pass-Thru* into the TF and AFV models, we just add the excess dividend payouts at dividend payment dates as computed by Equation (2.19) to the coupon payments of the convertible bond, with the PDEs and final/boundary conditions for each model being unchanged. The treatment of coupon payments will be discussed in the next section.

2.2.3 Dividend, Coupon and Accrued Interest Payments

In this section, we discuss in detail the handling of dividend, coupon and accrued interest payments. The treatment described below applies to both the TF and AFV models in the same way.

In our approach, we assume a discrete dividend model, *i.e.*, dividend payments on the common stock are paid discretely (*e.g.*, quarterly, semi-annually or annually). If a discrete dividend D is paid at time t_d, according to the no-arbitrage argument and following the idea of discrete dividend handling for American options presented in [27], we must have

$$S(t_d^+) = S(t_d^-) - D, \tag{2.20}$$

$$U(S(t_d^-), t_d^-) = U(S(t_d^+), t_d^+), \tag{2.21}$$

$$B(S(t_d^-), t_d^-) = B(S(t_d^+), t_d^+), \tag{2.22}$$

and

$$C(S(t_d^-), t_d^-) = C(S(t_d^+), t_d^+), \tag{2.23}$$

where t_d^- is the time immediately before the dividend payment, t_d^+ the time immediately after the dividend payment, $S(t_d^-)$ the stock price immediately before the dividend payment, $S(t_d^+)$ the stock price immediately after the dividend payment, and $U(S,t)$, $B(S,t)$ and $C(S,t)$ denote the value of the convertible bond, the value of the bond component of the CB and the value of the equity component of the CB, respectively, when the underlying stock price is S at time t.

The rationale for Equation (2.20) is clear; see, for example, [27]. The rationale behind Equations (2.21) to (2.23) is that, since the holder of the convertible bond (and also the holder of the bond component and the equity component) does not receive the dividend on the stock, to avoid arbitrage possibility, the value of the convertible bond (also the bond component and the equity component) must be the same immediately before and immediately after the dividend date.

Consider a coupon payment K_i paid at time t_i. Let t_i^+ be the time immediately after the coupon payment, and t_i^- the time immediately before the coupon payment. By a similar no-arbitrage argument, the discrete coupon payments are handled by setting

$$\begin{aligned} U(S, t_i^-) &= U(S, t_i^+) + K_i, \\ B(S, t_i^-) &= B(S, t_i^+) + K_i, \\ C(S, t_i^-) &= C(S, t_i^+), \end{aligned} \tag{2.24}$$

where U is the value of the convertible bond, B the value of the bond component and C the value of the equity component.

The call price B_c and the put price B_p in the boundary conditions for the PDEs described earlier are dirty prices which include accrued interest, whereas in practice the quoted call price and quoted put price are clean prices. Therefore, in our implementation, we need to compute the accrued interest to get the actual call price and put price. Let t be the current time in the forward direction. If the last coupon payment was paid at t_{i-1} and the next coupon payment in the amount of K_i is paid at t_i, then the accrued interest on the pending coupon payment at time t is normally taken to be

$$AccI(t) = K_i \frac{t - t_{i-1}}{t_i - t_{i-1}}. \tag{2.25}$$

The dirty call price B_c and the dirty put price B_p are computed by

$$\begin{aligned} B_c(t) &= B_c^{cl} + AccI(t), \\ B_p(t) &= B_p^{cl} + AccI(t), \end{aligned} \tag{2.26}$$

where B_c^{cl} and B_p^{cl} are the respective clean prices.

According to Hull [12], we assume that the conversion, call and put would happen immediately after the coupon payment; and in our numerical procedure, we apply the convesion, callability and puttability constraints first, and then add the coupon payment (because we are proceeding backwards in time).

Chapter 3

Finite Difference Discretization

From Chapter 2, we can see that incorporating the dividend protection (either *Conversion Ratio Adjustment* or *Dividend Pass-Thru*) into the TF and AFV models does not change the original PDEs associated with these models. However, the PDEs in the TF and AFV models are parabolic PDEs for which an analytical solution is not known for the associated conversion, callability and puttability constraints. Therefore, we must solve the PDEs numerically. In this chapter, we use the Finite Difference Method (FDM) for both the spatial and temporal discretization for the PDEs arising from the TF and AFV models, respectively.

3.1 Variable Transformations for PDEs

To simplify the problem, we consider some variable transformations for the PDEs associated with the TF and AFV models.

3.1.1 Transformed PDEs Associated with the TF Model

We apply two variable transformations for the coupled PDEs (2.1) and (2.2). Recall that U is the value of the CB, and B the value of COCB.

- Set $\tau = T - t$ to transform the PDEs from forward to backward time.

  Equations (2.1) and (2.2) become

$$\frac{\partial U}{\partial \tau} = \frac{\sigma^2 S^2}{2}\frac{\partial^2 U}{\partial S^2} + rS\frac{\partial U}{\partial S} - r(U-B) - (r+r_c)B, \tag{3.1}$$

$$\frac{\partial B}{\partial \tau} = \frac{\sigma^2 S^2}{2}\frac{\partial^2 B}{\partial S^2} + rS\frac{\partial B}{\partial S} - (r+r_c)B. \tag{3.2}$$

  We ignore $f(t)$ for now and handle coupon payments as described in Section 2.2.3. The same approach applies to the PDEs associated with the AFV model as well. We refer to these two transformed PDEs as *τ-transformed* PDEs.

- Set $\tau = T - t$ and $x = \ln(\frac{S}{S_{int}})$ (where S_{int} is the stock price at $t = 0$).

  Then we have

$$\frac{\partial U}{\partial S} = \frac{\partial U}{\partial x}\frac{dx}{dS} = \frac{\partial U}{\partial x}\frac{1}{S},$$

$$\frac{\partial B}{\partial S} = \frac{\partial B}{\partial x}\frac{1}{S},$$

$$\begin{aligned}\frac{\partial^2 U}{\partial S^2} &= \frac{\partial(\frac{\partial U}{\partial S})}{\partial S} = \frac{\partial(\frac{\partial U}{\partial x}\frac{1}{S})}{\partial S} \\ &= \frac{1}{S}\frac{\partial(\frac{\partial U}{\partial x})}{\partial S} - \frac{1}{S^2}\frac{\partial U}{\partial x} \\ &= \frac{1}{S^2}\frac{\partial^2 U}{\partial x^2} - \frac{1}{S^2}\frac{\partial U}{\partial x} \\ &= \frac{1}{S^2}(\frac{\partial^2 U}{\partial x^2} - \frac{\partial U}{\partial x}),\end{aligned}$$

$$\frac{\partial^2 B}{\partial S^2} = \frac{1}{S^2}(\frac{\partial^2 B}{\partial x^2} - \frac{\partial B}{\partial x}).$$

  Therefore, Equations (2.1) and (2.2) become

$$\frac{\partial U}{\partial \tau} = \frac{\sigma^2}{2}\frac{\partial^2 U}{\partial x^2} + (r - \frac{\sigma^2}{2})\frac{\partial U}{\partial x} - r(U-B) - (r+r_c)B, \tag{3.3}$$

$$\frac{\partial B}{\partial \tau} = \frac{\sigma^2}{2}\frac{\partial^2 B}{\partial x^2} + (r - \frac{\sigma^2}{2})\frac{\partial B}{\partial x} - (r+r_c)B. \tag{3.4}$$

  We refer to these two transformed PDEs as *x-transformed* PDEs.

In Chapter 4, we explore two numerical methods, namely the PSOR method and the penalty method, to solve the PDEs arising from a convertible bond without dividends under the TF model. Our preliminary investigation showed that *x-transformed* PDEs work better for the PSOR method, while *τ-transformed* PDEs work better for the penalty method. Therefore, we use *x-transformed* PDEs for the PSOR method, and *τ-transformed* PDEs for the penalty method throughout this research thesis.

3.1.2 Transformed PDEs Associated with the AFV Model

We set $\tau = T - t$ and the PDEs in Section 2.1.2 become (setting $X = B$)

$$\frac{\partial U}{\partial \tau} = \frac{\sigma^2 S^2}{2}\frac{\partial^2 U}{\partial S^2} + (r + p\eta)S\frac{\partial U}{\partial S} - (r+p)U + p\max(\kappa S(1-\eta), RB), \quad (3.5)$$

$$\frac{\partial B}{\partial \tau} = \frac{\sigma^2 S^2}{2}\frac{\partial^2 B}{\partial S^2} + (r + p\eta)S\frac{\partial B}{\partial S} - (r+p)B + RpB, \quad (3.6)$$

$$\frac{\partial C}{\partial \tau} = \frac{\sigma^2 S^2}{2}\frac{\partial^2 C}{\partial S^2} + (r + p\eta)S\frac{\partial C}{\partial S} - (r+p)C + p\max(\kappa S(1-\eta) - RB, 0), \quad (3.7)$$

where U is the value of the convertible bond, B the value of the bond component, and C the value of the equity component.

3.2 Formation of Finite Difference Discretization

In this section, we discuss how to apply the Finite Difference Method to discretise the transformed PDEs described in the previous section in both the spatial and temporal dimensions.

3.2.1 The Finite Difference Grid

It is clear that the domain of S is $[0, \infty)$. In the implementation, we need to approximate the upper limit ∞ with a sufficiently large positive number S_{max}. For the *τ-transformed* PDEs and the *x-transformed* PDEs, respectively, we consider the region enclosed by $[S_{min}, S_{max}] \times [0, T]$ and $[x_{min}, x_{max}] \times [0, T]$, with uniform step-sizes in the spatial and

temporal dimensions. Clearly, $S_{min} = 0$. The choice of S_{max} is more complicated. On one hand, S_{max} has to be sufficiently large in order to avoid excessive error due to the truncation of the infinite domain. On the other hand, an unnecessarily large value of S_{max} increases the computational cost unnecessarily. Li [17] has shown that, for American options with exercise price equal to 100 and for convertible bonds with stock price at $t = 0$ equal to 100, $S_{max} = 500$ is a reasonable upper limit on S in the finite difference method. Since our numerical examples described later use similar parameters for the convertible bond as those in [17], we choose $S_{max} = 500$ as the upper limit on S in our case too. For the implementation in the x variables, we use $x_{min} = -16$ and $x_{max} = 2$, which corresponds to $S_{min} = 1.12535 \times 10^{-5}$ and $S_{max} = 738.906$, respectively. (Recall that $S = S_{int} \exp(x)$, and $S_{int} = 100$ in our examples.)

Letting $\Delta S = (S_{max} - S_{min})/N$ and $\Delta x = (x_{max} - x_{min})/N$, we divide the S-axis (or x-axis) into N equally spaced intervals. Then the spatial grid points are

$$S_n = S_{min} + n\Delta S, \quad n = 0, 1, \ldots, N.$$

or

$$x_n = x_{min} + n\Delta x, \quad n = 0, 1, \ldots, N.$$

In order to avoid interpolation errors when computing the convertible bond's value at $t = 0$ for the spot stock price, we choose N such that the spot stock price S_{int} or the corresponding spot x value x_{int} is a grid point in the spatial dimension.

Letting $\Delta\tau = T/M$, we divide the τ-axis into M equally spaced intervals. Then the temporal grid points are

$$\tau_m = m\Delta\tau, \quad m = 0, 1, \ldots, M.$$

We use the notation U_n^m to denote our numerical approximation to $U(S_n, \tau_m)$ (or $U(x_n, \tau_m)$), the value of U at the grid point (S_n, τ_m) (or (x_n, τ_m)). Similar notation applies to B and C as well.

3.2.2 Discretization of the τ-*Transformed* PDEs Associated with the TF Model

We use central differences to approximate the first and second derivatives in the spatial dimension, and use the θ-method to approximate the first derivative in the temporal dimension. We derive the finite difference discretization for the τ-*transformed* PDEs associated with the TF model in detail; the discretization for the PDEs associated with the AFV model follows in a similar way.

Using central differences in the spatial dimension, we have

$$\frac{\partial U}{\partial S}(S_n, \tau_{m+1}) \approx \frac{U_{n+1}^{m+1} - U_{n-1}^{m+1}}{2\Delta S}$$

and

$$\frac{\partial^2 U}{\partial S^2}(S_n, \tau_{m+1}) \approx \frac{U_{n+1}^{m+1} - 2U_n^{m+1} + U_{n-1}^{m+1}}{\Delta S^2}.$$

Using the θ-method in the temporal dimension, PDE (3.1) can be approximated by the discretized equation

$$\begin{aligned}
\frac{U_n^{m+1} - U_n^m}{\Delta\tau} &= (1-\theta)\left(\frac{\sigma^2 S_n^2}{2}\frac{U_{n+1}^m - 2U_n^m + U_{n-1}^m}{\Delta S^2} + rS_n\frac{U_{n+1}^m - U_{n-1}^m}{2\Delta S} - rU_n^m - r_c B_n^m\right) \\
&\quad +\theta\left(\frac{\sigma^2 S_n^2}{2}\frac{U_{n+1}^{m+1} - 2U_n^{m+1} + U_{n-1}^{m+1}}{\Delta S^2} + rS_n\frac{U_{n+1}^{m+1} - U_{n-1}^{m+1}}{2\Delta S}\right. \\
&\quad \left. -rU_n^{m+1} - r_c B_n^{m+1}\right), \qquad (3.8)
\end{aligned}$$

for $n = 1, \ldots, N-1$ and $m = 0, \ldots, M-1$.

For $n = 1, \ldots, N-1$, define

$$\begin{aligned}
\alpha_n &= \left(\frac{\sigma^2 S_n^2}{2\Delta S^2} - \frac{rS_n}{2\Delta S}\right)\Delta\tau, \\
\beta_n &= \left(\frac{\sigma^2 S_n^2}{2\Delta S^2} + \frac{rS_n}{2\Delta S}\right)\Delta\tau. \qquad (3.9)
\end{aligned}$$

Using (3.9), Equation (3.8) becomes (after simplification)

$$U_n^{m+1} - \theta\left(\alpha_n U_{n-1}^{m+1} - (r\Delta\tau + \alpha_n + \beta_n)U_n^{m+1} + \beta_n U_{n+1}^{m+1}\right)$$

$$= U_n^m + (1-\theta)\left(\alpha_n U_{n-1}^m - (r\Delta\tau + \alpha_n + \beta_n)U_n^m + \beta_n U_{n+1}^m\right) - r_c\left((1-\theta)B_n^m + \theta B_n^{m+1}\right)\Delta\tau. \quad (3.10)$$

Similarly, PDE (3.2) can be approximated by the discretized equation

$$\begin{aligned}\frac{B_n^{m+1} - B_n^m}{\Delta\tau} &= (1-\theta)\left(\frac{\sigma^2 S_n^2}{2}\frac{B_{n+1}^m - 2B_n^m + B_{n-1}^m}{\Delta S^2} + rS_n\frac{B_{n+1}^m - B_{n-1}^m}{2\Delta S} - (r+r_c)B_n^m\right)\\ &\quad +\theta\left(\frac{\sigma^2 S_n^2}{2}\frac{B_{n+1}^{m+1} - 2B_n^{m+1} + B_{n-1}^{m+1}}{\Delta S^2} + rS_n\frac{B_{n+1}^{m+1} - B_{n-1}^{m+1}}{2\Delta S}\right.\\ &\quad \left.-(r+r_c)B_n^{m+1}\right), \quad (3.11)\end{aligned}$$

for $n = 1, \ldots, N-1$ and $m = 0, \ldots, M-1$.

Substituting (3.9) into (3.11), we get

$$\begin{aligned}&B_n^{m+1} - \theta\left(\alpha_n B_{n-1}^{m+1} - ((r+r_c)\Delta\tau + \alpha_n + \beta_n)B_n^{m+1} + \beta_n B_{n+1}^{m+1}\right)\\ =\ &B_n^m + (1-\theta)\left(\alpha_n B_{n-1}^m - ((r+r_c)\Delta\tau + \alpha_n + \beta_n)B_n^m + \beta_n B_{n+1}^m\right). \quad (3.12)\end{aligned}$$

Note that for the θ-method, we have the following:

for	$\theta = 0$,	we get the explicit scheme;
for	$\theta = 1/2$,	we get the Crank-Nicolson scheme;
for	$\theta = 1$,	we get the fully implicit scheme.

We discuss the choice of θ in Chapter 4.

3.2.3 Discretization of the x-*Transformed* PDEs Associated with the TF Model

Using central differences in the spatial dimension and the θ-method in the temporal dimension, PDEs (3.3) and (3.4) can be approximated by the discretized equations

$$\begin{aligned}\frac{U_n^{m+1} - U_n^m}{\Delta\tau} &= (1-\theta)\left(\frac{\sigma^2}{2}\frac{U_{n+1}^m - 2U_n^m + U_{n-1}^m}{\Delta x^2} + (r - \frac{\sigma^2}{2})\frac{U_{n+1}^m - U_{n-1}^m}{2\Delta x} - rU_n^m - r_c B_n^m\right)\\ &\quad +\theta\left(\frac{\sigma^2}{2}\frac{U_{n+1}^{m+1} - 2U_n^{m+1} + U_{n-1}^{m+1}}{\Delta x^2} + (r - \frac{\sigma^2}{2})\frac{U_{n+1}^{m+1} - U_{n-1}^{m+1}}{2\Delta x}\right.\\ &\quad \left.-rU_n^{m+1} - r_c B_n^{m+1}\right) \quad (3.13)\end{aligned}$$

and

$$\begin{aligned}\frac{B_n^{m+1}-B_n^m}{\Delta\tau} &= (1-\theta)\left(\frac{\sigma^2}{2}\frac{B_{n+1}^m-2B_n^m+B_{n-1}^m}{\Delta x^2}+(r-\frac{\sigma^2}{2})\frac{B_{n+1}^m-B_{n-1}^m}{2\Delta x}-(r+r_c)B_n^m\right)\\ &\quad+\theta\left(\frac{\sigma^2}{2}\frac{B_{n+1}^{m+1}-2B_n^{m+1}+B_{n-1}^{m+1}}{\Delta x^2}+(r-\frac{\sigma^2}{2})\frac{B_{n+1}^{m+1}-B_{n-1}^{m+1}}{2\Delta x}\right.\\ &\quad\left.-(r+r_c)B_n^{m+1}\right), \qquad (3.14)\end{aligned}$$

for $n=1,\ldots,N-1$ and $m=0,\ldots,M-1$.

Define

$$\begin{aligned}\alpha &= \left(\frac{\sigma^2}{2\Delta x^2}-\frac{r-\frac{\sigma^2}{2}}{2\Delta x}\right)\Delta\tau,\\ \beta &= \left(\frac{\sigma^2}{2\Delta x^2}+\frac{r-\frac{\sigma^2}{2}}{2\Delta x}\right)\Delta\tau. \qquad (3.15)\end{aligned}$$

Using (3.15), Equations (3.13) and (3.14) become (after simplification)

$$\begin{aligned}&U_n^{m+1}-\theta\left(\alpha U_{n-1}^{m+1}-(r\Delta\tau+\alpha+\beta)U_n^{m+1}+\beta U_{n+1}^{m+1}\right)\\ =\;&U_n^m+(1-\theta)\left(\alpha U_{n-1}^m-(r\Delta\tau+\alpha+\beta)U_n^m+\beta U_{n+1}^m\right)\\ &-r_c\left((1-\theta)B_n^m+\theta B_n^{m+1}\right)\Delta\tau \qquad (3.16)\end{aligned}$$

and

$$\begin{aligned}&B_n^{m+1}-\theta\left(\alpha B_{n-1}^{m+1}-((r+r_c)\Delta\tau+\alpha+\beta)B_n^{m+1}+\beta B_{n+1}^{m+1}\right)\\ =\;&B_n^m+(1-\theta)\left(\alpha B_{n-1}^m-((r+r_c)\Delta\tau+\alpha+\beta)B_n^m+\beta B_{n+1}^m\right), \qquad (3.17)\end{aligned}$$

respectively.

3.2.4 Discretization of the Transformed PDEs Associated with the AFV Model

Using similar discretizations as in Section 3.2.3, PDEs (3.5), (3.6) and (3.7) can be approximated by the discretized equations

$$\frac{U_n^{m+1}-U_n^m}{\Delta\tau} = (1-\theta)\left(\frac{\sigma^2S_n^2}{2}\frac{U_{n+1}^m-2U_n^m+U_{n-1}^m}{\Delta S^2}+(r+p\eta)S_n\frac{U_{n+1}^m-U_{n-1}^m}{2\Delta S}\right.$$

$$
\begin{aligned}
&-(r+p)U_n^m + p\max\Big(\kappa S_n(1-\eta), RB_n^m\Big)\Big) \\
&+\theta\left(\frac{\sigma^2 S_n^2}{2}\frac{U_{n+1}^{m+1}-2U_n^{m+1}+U_{n-1}^{m+1}}{\Delta S^2} + (r+p\eta)S_n\frac{U_{n+1}^{m+1}-U_{n-1}^{m+1}}{2\Delta S}\right. \\
&\left.-(r+p)U_n^{m+1} + p\max\Big(\kappa S_n(1-\eta), RB_n^{m+1}\Big)\Big)\right), \qquad (3.18)
\end{aligned}
$$

$$
\begin{aligned}
\frac{B_n^{m+1}-B_n^m}{\Delta\tau} &= (1-\theta)\left(\frac{\sigma^2 S_n^2}{2}\frac{B_{n+1}^m-2B_n^m+B_{n-1}^m}{\Delta S^2} + (r+p\eta)S_n\frac{B_{n+1}^m-B_{n-1}^m}{2\Delta S}\right. \\
&\quad\left.-\Big(r+(1-R)p\Big)B_n^m\right) \\
&\quad+\theta\left(\frac{\sigma^2 S_n^2}{2}\frac{B_{n+1}^{m+1}-2B_n^{m+1}+B_{n-1}^{m+1}}{\Delta S^2} + (r+p\eta)S_n\frac{B_{n+1}^{m+1}-B_{n-1}^{m+1}}{2\Delta S}\right. \\
&\quad\left.-\Big(r+(1-R)p\Big)B_n^{m+1}\right), \qquad (3.19)
\end{aligned}
$$

and

$$
\begin{aligned}
\frac{C_n^{m+1}-C_n^m}{\Delta\tau} &= (1-\theta)\left(\frac{\sigma^2 S_n^2}{2}\frac{C_{n+1}^m-2C_n^m+C_{n-1}^m}{\Delta S^2} + (r+p\eta)S_n\frac{C_{n+1}^m-C_{n-1}^m}{2\Delta S}\right. \\
&\quad\left.-(r+p)C_n^m + p\max\Big(\kappa S_n(1-\eta) - RB_n^m, 0\Big)\right) \\
&\quad+\theta\left(\frac{\sigma^2 S_n^2}{2}\frac{C_{n+1}^{m+1}-2C_n^{m+1}+C_{n-1}^{m+1}}{\Delta S^2} + (r+p\eta)S_n\frac{C_{n+1}^{m+1}-C_{n-1}^{m+1}}{2\Delta S}\right. \\
&\quad\left.-(r+p)C_n^{m+1} + p\max\Big(\kappa S_n(1-\eta) - RB_n^{m+1}, 0\Big)\right), \qquad (3.20)
\end{aligned}
$$

for $n = 1, \ldots, N-1$ and $m = 0, \ldots, M-1$.

For $n = 1, \ldots, N-1$, define

$$
\begin{aligned}
\alpha_n^{AFV} &= \left(\frac{\sigma^2 S_n^2}{2\Delta S^2} - \frac{(r+P\eta)S_n}{2\Delta S}\right)\Delta\tau, \\
\beta_n^{AFV} &= \left(\frac{\sigma^2 S_n^2}{2\Delta S^2} + \frac{(r+P\eta)S_n}{2\Delta S}\right)\Delta\tau. \qquad (3.21)
\end{aligned}
$$

Using (3.21), Equations (3.18), (3.19) and (3.20) become (after simplification)

$$
\begin{aligned}
&U_n^{m+1} - \theta\Big(\alpha_n^{AFV}U_{n-1}^{m+1} - \Big((r+p)\Delta\tau + \alpha_n^{AFV} + \beta_n^{AFV}\Big)U_n^{m+1} + \beta_n^{AFV}U_{n+1}^{m+1}\Big) \\
= \;& U_n^m + (1-\theta)\Big(\alpha_n^{AFV}U_{n-1}^m - \Big((r+p)\Delta\tau + \alpha_n^{AFV} + \beta_n^{AFV}\Big)U_n^m + \beta_n^{AFV}U_{n+1}^m\Big) \\
&+p\Big((1-\theta)\max\Big(\kappa S_n(1-\eta), RB_n^m\Big) \\
&+\theta\max\Big(\kappa S_n(1-\eta), RB_n^{m+1}\Big)\Big)\Delta\tau, \qquad (3.22)
\end{aligned}
$$

$$
\begin{aligned}
& B_n^{m+1} - \theta\Big(\alpha_n^{AFV} B_{n-1}^{m+1} - \Big(\big(r + (1-R)p\big)\Delta\tau + \alpha_n^{AFV} + \beta_n^{AFV}\Big) B_n^{m+1} + \beta_n^{AFV} B_{n+1}^{m+1}\Big) \\
= \; & B_n^m + (1-\theta)\Big(\alpha_n^{AFV} B_{n-1}^m - \Big(\big(r + (1-R)p\big)\Delta\tau + \alpha_n^{AFV} + \beta_n^{AFV}\Big) B_n^m \\
& + \beta_n^{AFV} B_{n+1}^m\Big), \qquad (3.23)
\end{aligned}
$$

and

$$
\begin{aligned}
& C_n^{m+1} - \theta\Big(\alpha_n^{AFV} C_{n-1}^{m+1} - \big((r+p)\Delta\tau + \alpha_n^{AFV} + \beta_n^{AFV}\big) C_n^{m+1} + \beta_n^{AFV} C_{n+1}^{m+1}\Big) \\
= \; & C_n^m + (1-\theta)\Big(\alpha_n^{AFV} C_{n-1}^m - \big((r+p)\Delta\tau + \alpha_n^{AFV} + \beta_n^{AFV}\big) C_n^m + \beta_n^{AFV} C_{n+1}^m\Big) \\
& + p\Big((1-\theta)\max\big(\kappa S_n(1-\eta) - RB_n^m, 0\big) \\
& + \theta\max\big(\kappa S_n(1-\eta) - RB_n^{m+1}, 0\big)\Big)\Delta\tau, \qquad (3.24)
\end{aligned}
$$

3.3 Boundary Conditions

In this section, we discuss how to incorporate the boundary conditions for the transformed PDEs described in Section 3.1.

3.3.1 For the τ-*Transformed* PDEs Associated with the TF Model

- When $S = S_{min} = 0$:

  PDEs (3.1) and (3.2) reduce to

$$
\begin{aligned}
\frac{\partial U}{\partial \tau} &= -rU - r_c B, \\
\frac{\partial B}{\partial \tau} &= -(r + r_c)B,
\end{aligned}
$$

  which can be discretized as

$$
\begin{aligned}
\frac{U_0^{m+1} - U_0^m}{\Delta\tau} &= -(1-\theta)(rU_0^m + r_c B_0^m) - \theta(rU_0^{m+1} + r_c B_0^{m+1}), \\
\frac{B_0^{m+1} - B_0^m}{\Delta\tau} &= -(1-\theta)(r + r_c)B_0^m - \theta(r + r_c)B_0^{m+1}.
\end{aligned}
$$

The equations above can be rearranged as

$$(1+\theta r\Delta\tau)U_0^{m+1} = \Big(1-(1-\theta)r\Delta\tau\Big)U_0^m - r_c\Delta\tau\Big((1-\theta)B_0^m+\theta B_0^{m+1}\Big), \tag{3.25}$$

$$\Big(1+\theta(r+r_c)\Delta\tau\Big)B_0^{m+1} = \Big(1-(1-\theta)(r+r_c)\Delta\tau\Big)B_0^m, \tag{3.26}$$

for $m = 0, \ldots, M-1$.

- When $S = S_{max}$:

  In this case, the convertible bond will be converted into stock, so we have

$$U_N^{m+1} = \kappa S_N, \tag{3.27}$$

$$B_N^{m+1} = 0, \tag{3.28}$$

for $m = 0, \ldots, M-1$.

3.3.2 For the *x-Transformed* PDEs Associated with the TF Model

- When $x = x_{min}$:

  We saw earlier that, when $S = 0$, the PDEs (3.1) and (3.2) reduce to

$$\frac{\partial U}{\partial \tau} = -rU - r_cB,$$
$$\frac{\partial B}{\partial \tau} = -(r+r_c)B.$$

  Since these equations do not involve any derivatives with respect to S, they also apply to the *x-transformed* PDEs (3.3) and (3.4) at $x = -\infty$. Therefore, we are approximating the boundary condition at $x = -\infty$ by our numerical boundary condition at $x = x_{min}$, and we use (3.25) and (3.26) for the boundary conditions of the discretized versions of (3.3) and (3.4) as well.

- When $x = x_{max}$:

  As for the case $S = S_{max}$ for the τ-*transformed* PDEs, we have

$$U_N^{m+1} = \kappa S_{int} \exp(x_N), \tag{3.29}$$

$$B_N^{m+1} = 0, \tag{3.30}$$

  for $m = 0, \ldots, M-1$.

3.3.3 For Transformed PDEs Associated with the AFV Model

- When $S = S_{min} = 0$:

  PDEs (3.5), (3.6) and (3.7) reduce to

$$\begin{aligned}
\frac{\partial U}{\partial \tau} &= -(r+p)U + pRB, \\
\frac{\partial B}{\partial \tau} &= -\Big(r + (1-R)p\Big)B, \\
\frac{\partial C}{\partial \tau} &= -(r+p)C,
\end{aligned}$$

  which can be discretized as

$$\begin{aligned}
\frac{U_0^{m+1} - U_0^m}{\Delta\tau} &= -(1-\theta)\Big((r+p)U_0^m - pRB_0^m\Big) - \theta\Big((r+p)U_0^{m+1} - pRB_0^{m+1}\Big), \\
\frac{B_0^{m+1} - B_0^m}{\Delta\tau} &= -(1-\theta)\Big(r + (1-R)p\Big)B_0^m - \theta\Big(r + (1-R)p\Big)B_0^{m+1}, \\
\frac{C_0^{m+1} - C_0^m}{\Delta\tau} &= -(1-\theta)(r+p)C_0^m - \theta(r+p)C_0^{m+1}.
\end{aligned}$$

  The equations above can be rearranged as

$$\begin{aligned}
\Big(1 + \theta(r+p)\Delta\tau\Big)U_0^{m+1} &= \Big(1 - (1-\theta)(r+p)\Delta\tau\Big)U_0^m \\
&\quad + pR\Delta\tau\Big((1-\theta)B_0^m + \theta B_0^{m+1}\Big),
\end{aligned} \tag{3.31}$$

$$\Big(1 + \theta\Big(r + (1-R)p\Big)\Delta\tau\Big)B_0^{m+1} = \Big(1 - (1-\theta)\Big(r + (1-R)p\Big)\Delta\tau\Big)B_0^m, \tag{3.32}$$

$$\Big(1+\theta(r+p)\Delta\tau\Big)C_0^{m+1} = \Big(1-(1-\theta)(r+p)\Delta\tau\Big)C_0^m, \tag{3.33}$$

for $m = 0, \ldots, M-1$.

- When $S = S_{max}$:

  Similarly as in the case of PDEs associated with the TF model, we have the following:

$$U_N^{m+1} = \kappa S_N, \tag{3.34}$$

$$B_N^{m+1} = 0, \tag{3.35}$$

$$C_N^{m+1} = \kappa S_N, \tag{3.36}$$

  for $m = 0, \ldots, M-1$.

3.4 Matrix Formulation

The discretized equations and boundary conditions described in the previous sections can be written in matrix format. For $m = 0, \ldots, M$, define

$$\begin{aligned} U^m &= (U_0^m \;\; U_1^m \;\; \cdots \;\; U_{N-1}^m \;\; U_N^m)^T, \\ B^m &= (B_0^m \;\; B_1^m \;\; \cdots \;\; B_{N-1}^m \;\; B_N^m)^T, \\ C^m &= (C_0^m \;\; C_1^m \;\; \cdots \;\; C_{N-1}^m \;\; C_N^m)^T, \end{aligned}$$

and let $\mathbf{I}$ be the $(N+1)\times(N+1)$ identity matrix. Then we have the following:

- For the τ-*Transformed* PDEs Associated with the TF Model:

  Equation (3.10), for $n = 1, \ldots, N-1$, together with the boundary conditions for U in (3.25) and (3.27) can be written in matrix format as

$$(\mathbf{I}-\theta\mathbf{M_U})U^{m+1} = \Big(\mathbf{I}+(1-\theta)\mathbf{M_U}\Big)U^m - r_c\Delta\tau\Big((1-\theta)B^m+\theta B^{m+1}\Big), \tag{3.37}$$

where

$$\mathbf{M_U} = \begin{pmatrix} -r\Delta\tau & 0 & 0 & \dots & 0 \\ \alpha_1 & -(r\Delta\tau + \alpha_1 + \beta_1) & \beta_1 & \dots & 0 \\ 0 & \alpha_2 & -(r\Delta\tau + \alpha_2 + \beta_2) & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & \alpha_{N-1} & -(r\Delta\tau + \alpha_{N-1} + \beta_{N-1}) & \beta_{N-1} \\ 0 & 0 & 0 & \dots & 0 \end{pmatrix},$$

and α_n and β_n (for $n = 1, \dots, N-1$) are defined in (3.9).

Similarly, Equation (3.12), for $n = 1, \dots, N-1$, together with the boundary conditions for B in (3.26) and (3.28) can be written in matrix format as

$$(\mathbf{I} - \theta\mathbf{M_B})B^{m+1} = \Big(\mathbf{I} + (1-\theta)\mathbf{M_B}\Big)B^m, \tag{3.38}$$

where

$$\mathbf{M_B} = \begin{pmatrix} -(r+r_c)\Delta\tau & 0 & 0 & \dots & 0 \\ \alpha_1 & -((r+r_c)\Delta\tau + \alpha_1 + \beta_1) & \beta_1 & \dots & 0 \\ 0 & \alpha_2 & -((r+r_c)\Delta\tau + \alpha_2 + \beta_2) & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \beta_{N-1} \\ 0 & 0 & 0 & \dots & 0 \end{pmatrix}.$$

At the first glance, it is not easy to see that the boundary conditions for U and B in Equations (3.27) and (3.28) are incorporated in the matrix formulations (3.37) and (3.38). In the following, we show that they are. The last row in (3.38) is equivalent to

$$B_N^{m+1} = B_N^m, \tag{3.39}$$

whence

$$B_N^M = B_N^{M-1} = B_N^{M-2} = \cdots = B_N^0 = 0 \tag{3.40}$$

since $B_N^0 = 0$ follows from the final condition (2.3). Therefore, (3.38) incorporates the boundary condition (3.28). Similarly, The last row in (3.37) is equivalent to

$$U_N^{m+1} = U_N^m - r_c\Delta\tau\Big((1-\theta)B_N^m + \theta B_N^{m+1}\Big), \tag{3.41}$$

which becomes

$$U_N^{m+1} = U_N^m, \tag{3.42}$$

since $B_N^m = B_N^{m+1} = 0$ by (3.40). Therefore,

$$U_N^M = U_N^{M-1} = U_N^{M-2} = \cdots = U_N^0 = \kappa S_N \tag{3.43}$$

since $U_N^0 = \kappa S_N$ follows from the final condition (2.3). Therefore, (3.37) incorporates the boundary condition (3.27).

- For the *x-Transformed* PDEs Associated with the TF Model:

  Equation (3.16), for $n = 1, \ldots, N-1$, together with the boundary conditions for U in (3.25) and (3.29) can be written in matrix format as

$$(\mathbf{I} - \theta\mathbf{M}_\mathbf{U}^\mathbf{x})U^{m+1} = \Big(\mathbf{I} + (1-\theta)\mathbf{M}_\mathbf{U}^\mathbf{x}\Big)U^m - r_c\Delta\tau\Big((1-\theta)B^m + \theta B^{m+1}\Big), \tag{3.44}$$

  where

$$\mathbf{M}_\mathbf{U}^\mathbf{x} = \begin{pmatrix} -r\Delta\tau & 0 & 0 & \ldots & 0 \\ \alpha & -(r\Delta\tau+\alpha+\beta) & \beta & \ldots & 0 \\ 0 & \alpha & -(r\Delta\tau+\alpha+\beta) & \ldots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \ldots & \alpha & -(r\Delta\tau+\alpha+\beta) & \beta \\ 0 & 0 & 0 & \ldots & 0 \end{pmatrix},$$

  and α and β are defined in (3.15).

  Similarly, Equation (3.17), for $n = 1, \ldots, N-1$, together with the boundary conditions for B in (3.26) and (3.30) can be written in matrix format as

$$(\mathbf{I} - \theta\mathbf{M}_\mathbf{B}^\mathbf{x})B^{m+1} = \Big(\mathbf{I} + (1-\theta)\mathbf{M}_\mathbf{B}^\mathbf{x}\Big)B^m, \tag{3.45}$$

where

$$\mathbf{M_B^x} = \begin{pmatrix} -(r+r_c)\Delta\tau & 0 & 0 & \dots & 0 \\ \alpha & -((r+r_c)\Delta\tau+\alpha+\beta) & \beta & \dots & 0 \\ 0 & \alpha & -((r+r_c)\Delta\tau+\alpha+\beta) & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \beta \\ 0 & 0 & 0 & \dots & 0 \end{pmatrix}.$$

Following reasoning similar to that for the τ-*Transformed* PDEs associated with the TF model, we can see that the boundary conditions for U and B in Equations (3.29) and (3.30) have been incorporated in the matrix formulations (3.44) and (3.45), respectively.

- For the τ-*Transformed* PDEs Associated with the AFV Model:

  Equation (3.22), for $n = 1, \dots, N-1$, together with the boundary conditions for U in (3.31) and (3.34) can be written in matrix format as

$$\begin{aligned}(\mathbf{I} - \theta\mathbf{M_U^{AFV}})U^{m+1} &= \left(\mathbf{I} + (1-\theta)\mathbf{M_U^{AFV}}\right)U^m + p\Delta\tau\Big((1-\theta)\max\left(\kappa(1-\eta)S, RB^m\right) \\ &\quad +\theta\max\left(\kappa(1-\eta)S, RB^{m+1}\right)\Big) \qquad (3.46)\end{aligned}$$

  where

$$\mathbf{M_U^{AFV}} = \begin{pmatrix} -r_U\Delta\tau & 0 & 0 & \dots & 0 \\ \alpha_1^{AFV} & -(r_U\Delta\tau+\alpha_1^{AFV}+\beta_1^{AFV}) & \beta_1^{AFV} & \dots & 0 \\ 0 & \alpha_2^{AFV} & -(r_U\Delta\tau+\alpha_2^{AFV}+\beta_2^{AFV}) & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \beta_{N-1}^{AFV} \\ 0 & 0 & 0 & \dots & 0 \end{pmatrix},$$

  α_n^{AFV} and β_n^{AFV} (for $n = 1, \dots, N-1$) are defined in (3.21), $r_U = r + p$, $S = (S_0 \;\; S_1 \;\; \cdots \;\; S_{N-1} \;\; S_N)^T$, and the max in (3.46) and the equations below is taken componentwise (as in MatLab).

Similarly, Equation (3.23), for $n = 1, \ldots, N-1$, together with the boundary conditions for B in (3.32) and (3.35) can be written in matrix format as

$$(\mathbf{I} - \theta \mathbf{M_B^{AFV}}) B^{m+1} = \Big(\mathbf{I} + (1-\theta)\mathbf{M_B^{AFV}}\Big) B^m, \tag{3.47}$$

where

$$\mathbf{M_B^{AFV}} = \begin{pmatrix} -r_B\Delta\tau & 0 & 0 & \ldots & 0 \\ \alpha_1^{AFV} & -(r_B\Delta\tau + \alpha_1^{AFV} + \beta_1^{AFV}) & \beta_1^{AFV} & \ldots & 0 \\ 0 & \alpha_2^{AFV} & -(r_B\Delta\tau + \alpha_2^{AFV} + \beta_2^{AFV}) & \ldots & 0 \\ \vdots & \vdots & \vdots & \ddots & \beta_{N-1}^{AFV} \\ 0 & 0 & 0 & \ldots & 0 \end{pmatrix},$$

and $r_B = r + (1-R)p$.

Finally, Equation (3.24), for $n = 1, \ldots, N-1$, together with the boundary conditions for C in (3.33) and (3.36) can be written in matrix format as

$$\begin{aligned}(\mathbf{I} - \theta \mathbf{M_C^{AFV}}) C^{m+1} \;=\; & \Big(\mathbf{I} + (1-\theta)\mathbf{M_C^{AFV}}\Big) C^m + p\Delta\tau\Big((1-\theta)\max\Big(\kappa(1-\eta)S - RB^m, \mathbf{0}\Big) \\ & + \theta \max\Big(\kappa(1-\eta)S - RB^{m+1}, \mathbf{0}\Big)\Big)\end{aligned} \tag{3.48}$$

where $\mathbf{M_C^{AFV}} = \mathbf{M_U^{AFV}}$, and $\mathbf{0}$ is the $(N+1) \times 1$ zero-vector.

Following similar reasoning as in the case for the τ-*Transformed* PDEs associated with the TF model, we can see that the boundary conditions for U, B and C in Equations (3.34), (3.35) and (3.36) have been incorporated in the matrix formulations (3.46), (3.47) and (3.48), respectively.

Chapter 4

Numerical Algorithms

Due to the complexity of the conversion, callability and puttability features associated with a convertible bond, it is not possible to apply direct methods to solve the discretized systems arising from either the TF model or the AFV model described in the previous chapter. Therefore, iterative methods are needed to incorporate the free boundary conditions arising from conversion, callability and puttability constraints. In this chapter, we consider two iterative methods: the Projected Successive Over-Relaxation (PSOR) method and the penalty method. First, we apply the PSOR method and the penalty method, respectively, to solve the systems arising from a convertible bond without dividends under the TF model, and compare the numerical results produced by these two methods in terms of rate of convergence, number of iterations and computation time. Then we apply the better of the two methods from this comparison to solve the systems arising from a convertible bond with dividend protection under both the TF and AFV models.

4.1 The PSOR Method Versus the Penalty Method for Convertible Bonds Without Dividends Applied to the TF Model

4.1.1 PSOR Method

The PSOR method is an extension of the SOR method for solving free boundary problems such as those arising from American options [27]. It can also be used to solve the free boundary problem arising from convertible bonds [17]. The idea behind the PSOR method is to explicitly apply the conversion, callability and puttability constraints to the SOR values (defined later) of B and U at each iteration so that the effect of the constraints is immediately felt in the calculation of each component of B and U.

In this case, we use the x-transformed PDEs described in Section 3.1 (*i.e.,* set $\tau = T-t$ and $x = \ln(\frac{S}{S_{int}})$). We use the Crank-Nicolson scheme for the discretization in the temporal dimension (*i.e.,* set $\theta = 1/2$ in the θ-method). For the boundary points, we use Equations (3.25), (3.26), (3.29) and (3.29) to compute U_0^{m+1}, B_0^{m+1}, U_N^{m+1} and B_N^{m+1}. Then apply the conversion, callability and puttability constraints explicitly to update their values according to Algorithm 1.

For the interior points, we need to apply the PSOR method. From Equations (3.17) and (3.16), for $n = 1, \ldots, N-1$, we have (setting $\theta = 1/2$)

$$\begin{aligned} & B_n^{m+1} - \frac{1}{2}\left(\alpha B_{n-1}^{m+1} - ((r+r_c)\Delta\tau + \alpha + \beta)B_n^{m+1} + \beta B_{n+1}^{m+1}\right) \\ = \; & B_n^m + (1-\frac{1}{2})\left(\alpha B_{n-1}^m - ((r+r_c)\Delta\tau + \alpha + \beta)B_n^m + \beta B_{n+1}^m\right), \end{aligned}$$

and

$$\begin{aligned} & U_n^{m+1} - \frac{1}{2}\left(\alpha U_{n-1}^{m+1} - (r\Delta\tau + \alpha + \beta)U_n^{m+1} + \beta U_{n+1}^{m+1}\right) \\ = \; & U_n^m + (1-\frac{1}{2})\left(\alpha U_{n-1}^m - (r\Delta\tau + \alpha + \beta)U_n^m + \beta U_{n+1}^m\right) \\ & -r_c\left((1-\frac{1}{2})B_n^m + \theta B_n^{m+1}\right)\Delta\tau. \end{aligned}$$

Algorithm 1 Explicit application of the conversion, callability and puttability constraints to U_n^{m+1} and B_n^{m+1} for the TF model

Given U_n^{m+1} and B_n^{m+1},

{Apply the callability constraint:}

if $U_n^{m+1} > \max(B_c, \kappa S_n)$ **then**

$U_n^{m+1} = \max(B_c, \kappa S_n)$;

$B_n^{m+1} = 0$;

end if

{Apply the puttability constraint:}

if $U_n^{m+1} < B_p$ **then**

$U_n^{m+1} = B_p$;

$B_n^{m+1} = B_p$;

end if

{Apply the conversion constraint:}

if $U_n^{m+1} < \kappa S_n$ **then**

$U_n^{m+1} = \kappa S_n$;

$B_n^{m+1} = 0$;

end if

So the Gauss-Seidel iterates are:

$$\overline{B}_n^{m+1,k+1} = \frac{1}{2+(r+r_c)\Delta\tau+\alpha+\beta}\Big(\alpha B_{n-1}^{m+1,k+1}+\beta B_{n+1}^{m+1,k}+\alpha B_{n-1}^{m} +(2-((r+r_c)\Delta\tau+\alpha+\beta))B_n^m+\beta B_{n+1}^m\Big), \tag{4.1}$$

$$\overline{U}_n^{m+1,k+1} = \frac{1}{2+r\Delta\tau+\alpha+\beta}\Big(\alpha U_{n-1}^{m+1,k+1}+\beta U_{n+1}^{m+1,k}+\alpha U_{n-1}^{m} +(2-(r\Delta\tau+\alpha+\beta))U_n^m+\beta U_{n+1}^m - r_c\Delta\tau(B_n^m+B_n^{m+1,k+1})\Big). \tag{4.2}$$

The SOR values of B_n^{m+1} and U_n^{m+1} at iteration $k+1$ are given by

$$\tilde{B}_n^{m+1,k+1} = B_n^{m+1,k}+\omega(\overline{B}_n^{m+1,k+1}-B_n^{m+1,k}), \tag{4.3}$$

$$\tilde{U}_n^{m+1,k+1} = U_n^{m+1,k}+\omega(\overline{U}_n^{m+1,k+1}-U_n^{m+1,k}), \tag{4.4}$$

where $k \geq 0$ is the iteration index, $m = 0, \ldots, M-1$ the time-step index, and $1 < \omega < 2$ the over-relaxation parameter. Note that we start with the initial guess $B_n^{m+1,0} = B_n^m$ and $U_n^{m+1,0} = U_n^m$.

Then letting $U_n^{m+1,k+1} = \tilde{U}_n^{m+1,k+1}$ and $B_n^{m+1,k+1} = \tilde{B}_n^{m+1,k+1}$, we apply the conversion, callability and puttability constraints to $U_n^{m+1,k+1}$ and $B_n^{m+1,k+1}$ using Algorithm 1 (replacing U_n^{m+1}, B_n^{m+1} and S_n with $U_n^{m+1,k+1}$, $B_n^{m+1,k+1}$ and $S_{int}\exp(x_n)$, respectively). Finally, the conditions for exiting the PSOR iteration are

$$||B^{m+1,k+1}-B^{m+1,k}||_\infty \leq \epsilon,$$

$$||U^{m+1,k+1}-U^{m+1,k}||_\infty \leq \epsilon,$$

where ϵ is the tolerance. When these conditions are satisfied, we terminate the PSOR iteration and set

$$U^{m+1} = U^{m+1,k+1},$$

$$B^{m+1} = B^{m+1,k+1}.$$

The pseudocode for the Crank-Nicolson time-stepping and PSOR iteration are given in Algorithm 2 and Algorithm 3.

Algorithm 2 Crank-Nicolson time-stepping for a Convertible Bond Without Dividends applied to the TF model

$U^0 = B^0 = F + K$; {F: face value of the bond; K: coupon payment at maturity}

for $n = 0$ to N **do**

if $U_n^0 < \kappa S_{int} \exp(x_n)$ **then**

$U_n^0 = \kappa S_{int} \exp(x_n)$;

$B_n^0 = 0$;

end if

end for

for $m = 0$ to $M - 1$ **do**

$t = T - (m+1)\Delta\tau$; {T is the maturity, $\Delta\tau$ is the time interval}

calculate $AccI(t)$ using (2.25);

calculate $B_c(t)$ and $B_p(t)$ using (2.26) if applicable;

calculate U_0^{m+1} and B_0^{m+1} using (3.25) and (3.26);

apply the conversion, callability and puttability constraints explicitly to U_0^{m+1} and B_0^{m+1} using Algorithm 1;

calculate U_N^{m+1} and B_N^{m+1} using (3.29) and (3.30);

apply the conversion, callability and puttability constraints explicitly to U_N^{m+1} and B_N^{m+1} using Algorithm 1;

call the PSOR iteration;

if $t \in$ coupon payment period **then**

$U^{m+1} = U^{m+1} + K$;

$B^{m+1} = B^{m+1} + K$;

end if

end for

Algorithm 3 PSOR iteration for a Convertible Bond Without Dividends applied to the TF model

for $k = 0$ to MAXLOOP **do**

$error_U = error_B = 0$;

for $n = 1$ to $N - 1$ **do**

calculate $\overline{B}_n^{m+1,k+1}$ using (4.1);

calculate $\widetilde{B}_n^{m+1,k+1}$ using (4.3); $B_n^{m+1,k+1} = \widetilde{B}_n^{m+1,k+1}$

calculate $\overline{U}_n^{m+1,k+1}$ using (4.2);

calculate $\widetilde{U}_n^{m+1,k+1}$ using (4.4); $U_n^{m+1,k+1} = \widetilde{U}_n^{m+1,k+1}$

apply the conversion, callability and puttability constraints explicitly to $U_n^{m+1,k+1}$ and $B_n^{m+1,k+1}$ using Algorithm 1;

$error_U = \max(error_U, |U_n^{m+1,k+1} - U_n^{m+1,k}|)$;

$error_B = \max(error_B, |B_n^{m+1,k+1} - B_n^{m+1,k}|)$;

end for

if $error_U \leq \epsilon$ and $error_B \leq \epsilon$ **then**

break;

end if

end for

$U^{m+1} = U^{m+1,k+1}$;

$B^{m+1} = B^{m+1,k+1}$;

4.1.2 Penalty Method

The PSOR method applies the constraints explicitly at each iteration of computing the values of U and B, whereas the penalty method applies the constraints implicitly. Li presented a penalty scheme for a convertible bond without dividends in [17], but we believe her approach is not complete in that she did not incorporate the constraint $U \leq \max(B_c, \kappa S)$ into the penalty term. In this section, we develop a complete penalty scheme for a convertible bond without dividends.

For the penalty method, we work with the *τ-transformed* PDEs described in Section 3.1. According to the TF model [1], the equation for the convertible bond value U can be written as

$$\frac{\partial U}{\partial \tau} = \frac{\sigma^2 S^2}{2}\frac{\partial^2 U}{\partial S^2} + rS\frac{\partial U}{\partial S} - r(U-B) - (r+r_c)B \tag{4.5}$$

subject to the constraints

$$U \geq \max(B_p, \kappa S), \tag{4.6}$$

$$U \leq \max(B_c, \kappa S). \tag{4.7}$$

We apply the penalty method to solve the above constrained problem. Notice that this problem is also dependent on B. We follow the method described in [1] to decouple the resulting system involving U and B. The idea is as follows.

First, the value of B^{m+1} is estimated by solving (3.38), ignoring any constraints. This value of B^{m+1} is then used to compute U^{m+1} by solving (3.37), ignoring any constraints. Then, we apply the conversion, callability and puttability constraints explicitly to U^{m+1} and B^{m+1} as described in Algorithm 1. We then use the adjusted value of B^{m+1} for the constrained problem (4.5), (4.6) and (4.7), and apply the penalty method to solve for U^{m+1}.

Now we detail the penalty scheme for solving the constrained problem (4.5), (4.6) and (4.7). Define the payoff functions for a convertible bond as

$$U^*_{ceil}(S,\tau) = \max(B_c, \kappa S),$$

$$U^*_{floor}(S,\tau) \quad = \quad \max(B_p, \kappa S). \tag{4.8}$$

The penalty scheme can be written as

$$\frac{\partial U}{\partial \tau} \quad = \quad \frac{\sigma^2 S^2}{2}\frac{\partial^2 U}{\partial S^2} + rS\frac{\partial U}{\partial S} - rU - r_c B - \rho \max(U - U^*_{ceil}, 0) + \rho \max(U^*_{floor} - U, 0), \tag{4.9}$$

where $\rho > 0$ is the penalty parameter.

According to the results from [7], in which Forsyth and Vetzal apply the penalty method to the valuation of American options, and since we adopt a similar penalty scheme in the valuation of convertible bonds, we expect that the solution of (4.9) satisfies (or almost satisfies) $U^*_{floor} \leq U \leq U^*_{ceil}$. Notice that for any $U(S,\tau)$, at most one of the two penalty terms in (4.9) is active. We can also see that the penalty method satisfies the constraints approximately by introducing penalty terms. In the following, we discuss the discretization for equation (4.9).

The discretization of equation (4.9) without the penalty terms has been discussed in Section 3.2.2. The discretization of equation (4.9) can be written as

$$\mathcal{F}U_n^{m+1} = -w_n^{m+1} + q_n^{m+1}, \tag{4.10}$$

where

$$\begin{aligned} \mathcal{F}U_n^{m+1} \quad = \quad & (U_n^{m+1} - U_n^m) \\ & -\theta\left(\alpha_n U_{n-1}^{m+1} - (r\Delta\tau + \alpha_n + \beta_n)U_n^{m+1} + \beta_n U_{n+1}^{m+1}\right) \\ & -(1-\theta)\left(\alpha_n U_{n-1}^{m} - (r\Delta\tau + \alpha_n + \beta_n)U_n^{m} + \beta_n U_{n+1}^{m}\right) \\ & +r_c\Delta\tau\left((1-\theta)B_n^m + \theta B_n^{m+1}\right), \end{aligned} \tag{4.11}$$

$$w_n^{m+1} = \begin{cases} (U_n^{m+1} - U^{*,m+1}_{ceil,n})\rho\Delta\tau & \text{if } U_n^{m+1} > U^{*,m+1}_{ceil,n}, \\ 0 & \text{otherwise,} \end{cases}$$

and

$$q_n^{m+1} = \begin{cases} (U^{*,m+1}_{floor,n} - U_n^{m+1})\rho\Delta\tau & \text{if } U_n^{m+1} < U^{*,m+1}_{floor,n}, \\ 0 & \text{otherwise.} \end{cases}$$

Define

$$T_{ceil}(U_n^{m+1}) = \begin{cases} \rho\Delta\tau & \text{if } U_n^{m+1} > U_{ceil,n}^{*,m+1}, \\ 0 & \text{otherwise,} \end{cases} \tag{4.12}$$

and

$$T_{floor}(U_n^{m+1}) = \begin{cases} \rho\Delta\tau & \text{if } U_n^{m+1} < U_{floor,n}^{*,m+1}, \\ 0 & \text{otherwise.} \end{cases} \tag{4.13}$$

Then (4.10) can be written as

$$\begin{aligned} U_n^{m+1} - U_n^m &= \theta\left(\alpha_n U_{n-1}^{m+1} - (r\Delta\tau + \alpha_n + \beta_n)U_n^{m+1} + \beta_n U_{n+1}^{m+1}\right) \\ &\quad +(1-\theta)\left(\alpha_n U_{n-1}^{m} - (r\Delta\tau + \alpha_n + \beta_n)U_n^{m} + \beta_n U_{n+1}^{m}\right) \\ &\quad -r_c\Delta\tau\left((1-\theta)B_n^m + \theta B_n^{m+1}\right) \\ &\quad -T_{ceil}(U_n^{m+1})(U_n^{m+1} - U_{ceil,n}^{*,m+1}) + T_{floor}(U_n^{m+1})(U_{floor,n}^{*,m+1} - U_n^{m+1}). \end{aligned} \tag{4.14}$$

Define $\mathbf{P_{ceil}}(U^{m+1})$ and $\mathbf{P_{floor}}(U^{m+1})$ to be $(N+1)\times(N+1)$ diagonal matrices satisfying:

$$\mathbf{P_{ceil}}(U^{m+1})_{ij} = \begin{cases} \rho\Delta\tau & \text{if } U_i^{m+1} > U_{ceil,i}^{*,m+1} \text{ and } i = j, \\ 0 & \text{otherwise,} \end{cases} \tag{4.15}$$

and

$$\mathbf{P_{floor}}(U^{m+1})_{ij} = \begin{cases} \rho\Delta\tau & \text{if } U_i^{m+1} < U_{floor,i}^{*,m+1} \text{ and } i = j, \\ 0 & \text{otherwise.} \end{cases} \tag{4.16}$$

Extending (3.37), we can write (4.14) in matrix form as

$$\begin{aligned} & [\mathbf{I} - \theta\mathbf{M_U} + \mathbf{P_{ceil}}(U^{m+1}) + \mathbf{P_{floor}}(U^{m+1})]U^{m+1} \\ = \; & [\mathbf{I} + (1-\theta)\mathbf{M_U}]U^m - r_c\Delta\tau[(1-\theta)B^m + \theta B^{m+1}] \\ & +\mathbf{P_{ceil}}(U^{m+1})U_{ceil}^{*,m+1} + \mathbf{P_{floor}}(U^{m+1})U_{floor}^{*,m+1}. \end{aligned} \tag{4.17}$$

In the light of [7], we expect that the solution to (4.17) satisfies

$$U_n^{m+1} - U_{ceil,n}^{*,m+1} \to 0 \quad \text{or} \quad U_n^{m+1} - U_{floor,n}^{*,m+1} \to 0 \quad \text{as} \quad \rho \to \infty$$

at nodes for which $\mathcal{F}U_n^{m+1} \neq 0$. According to [17], we can set $\rho = 1/tol$ in practice, where tol is the tolerance for the penalty iteration which is discussed later.

We use a generalized Newton's Method to solve the discrete non-linear equations (4.17). Define

$$\begin{aligned} F(U^{m+1}) \equiv\ & (\mathbf{I} - \theta\mathbf{M_U})U^{m+1} - [\mathbf{I} + (1-\theta)\mathbf{M_U}]U^m \\ & + r_c\Delta\tau[(1-\theta)B^m + \theta B^{m+1}] \\ & + \mathbf{P_{ceil}}(U^{m+1})(U^{m+1} - U_{ceil}^{*,m+1}) - \mathbf{P_{floor}}(U^{m+1})(U_{floor}^{*,m+1} - U^{m+1}). \end{aligned} \tag{4.18}$$

Therefore, solving (4.17) is equivalent to finding U^{m+1} such that $F(U^{m+1}) = 0$. According to [7], we define the derivative of the penalty terms, required in Newton's Method, as

$$\frac{\partial P_{ceil}(U_n^{m+1})(U_n^{m+1} - U_{ceil,n}^{*,m+1})}{\partial U_n^{m+1}} = \begin{cases} \rho\Delta\tau & \text{if } U_n^{m+1} > U_{ceil,n}^{*,m+1}, \\ 0 & \text{otherwise;} \end{cases}$$

and

$$\frac{\partial P_{floor}(U_n^{m+1})(U_{floor,n}^{*,m+1} - U_n^{m+1})}{\partial U_n^{m+1}} = \begin{cases} -\rho\Delta\tau & \text{if } U_n^{m+1} < U_{floor,n}^{*,m+1}, \\ 0 & \text{otherwise.} \end{cases}$$

Then the Jacobian of $F(U^{m+1})$ can be computed as

$$J(U^{m+1}) = \mathbf{I} - \theta\mathbf{M_U} + \mathbf{P_{ceil}}(U^{m+1}) + \mathbf{P_{floor}}(U^{m+1}).$$

Applying Newton's Method to solve $F(U^{m+1}) = 0$, we have

$$\begin{aligned} J(U^{m+1,k})S_k &= -F(U^{m+1,k}), \\ U^{m+1,k+1} &= U^{m+1,k} + S_k, \end{aligned} \tag{4.19}$$

where S_k is the full Newton step obtained in the kth iteration, and $U^{m+1,k}$ is the kth estimate for U^{m+1}.

From (4.18) and (4.19), we can get the following (after simplification):

$$[\mathbf{I} - \theta\mathbf{M_U} + \mathbf{P_{ceil}}(U^{m+1,k}) + \mathbf{P_{floor}}(U^{m+1,k})]U^{m+1,k+1}$$

$$= [\mathbf{I} + (1-\theta)\mathbf{M_U}]U^m - r_c\Delta\tau[(1-\theta)B^m + \theta B^{m+1}]$$
$$+\mathbf{P_{ceil}}(U^{m+1,k})U^{*,m+1}_{ceil} + \mathbf{P_{floor}}(U^{m+1,k})U^{*,m+1}_{floor}, \quad (4.20)$$

which yields the penalty iteration described in Algorithm 4. The penalty time-stepping algorithm is shown in Algorithm 5. Notice that in Algorithm 5 we use the fully implicit scheme to get the estimates for B^{m+1} and U^{m+1} prior to applying the penalty method. According to [7] and [17], the fully implicit step is necessary for smoothing the discontinuity for both the value of the COCB and that of the CB.

4.1.3 Comparison of Numerical Results

In this section, we compare the numerical results for the PSOR and penalty methods for a convertible bond without dividends under the TF model. We compare the convergence ratio, number of iterations and computation time incurred by these two methods, and pick the better method to apply to the convertible bond with dividend protection under the TF and AFV models. Otherwise stated, throughout this research thesis, all implementation was in Matlab 6 and all experiments were performed on a Dell PowerEdge SC1425 (2xP4Xeon-3.6GHz) compute server under the operating system of RedHat Linux 7.3 (2.4.x kernel).

The parameters for the Convertible Bond without dividends are shown in Table 4.1. In the numerical results tables, "Price" is the convertible bond price at $t = 0$ and $S_{int} = 100$; "Grid Size" is the number of intervals in the spatial dimension; "Diff" is the difference between the solutions of the coarser grid and the finer grid; "Ratio" is the ratio of successive differences; "No. of Iterations" is the number of PSOR or penalty method iterations for each time-step; "max" is the maximum number of iterations over all time-steps; "min" is the minimum number of iterations; "avg" is the average number of iterations for each time-step; and "Time" is the total computation time in seconds obtained with the Matlab function CPUTIME. At each refinement, the number of time-steps and the grid size

Algorithm 4 Penalty iteration for a Convertible Bond without dividends under the TF model

$U^*_{ceil} = \max(B_c, \kappa S)$;

$U^*_{floor} = \max(B_p, \kappa S)$;

for $n = 0$ to N **do**

if $U_n^{m+1,0} > U^{*,m+1}_{ceil,n}$ **then**

$P_{ceil}(U_n^{m+1,0}) = \rho \Delta\tau$;

else

$P_{ceil}(U_n^{m+1,0}) = 0$;

end if

if $U_n^{m+1,0} < U^{*,m+1}_{floor,n}$ **then**

$P_{floor}(U_n^{m+1,0}) = \rho \Delta\tau$;

else

$P_{floor}(U_n^{m+1,0}) = 0$;

end if

end for

for $k = 0$, ..., until convergence **do**

solve (4.20);

update P_{ceil} and P_{floor} using $U^{m+1,k+1}$ by Equations (4.15) and (4.16);

if $\left[||U^{m+1,k+1} - U^{m+1,k}||_\infty \leq tol\right]$ or

$\left[P_{ceil}(U^{m+1,k+1}) == P_{ceil}(U^{m+1,k}) \text{ and } P_{floor}(U^{m+1,k+1}) == P_{floor}(U^{m+1,k})\right]$ **then**

exit from for loop;

end if

end for

$U^{m+1} = U^{m+1,k+1}$;

Algorithm 5 Penalty time-stepping for a Convertible Bond without dividends under the TF model

$U = B = F + K$; {F: face value of the bond; K: coupon payment at maturity}

for $n = 0$ to N **do**

if $U_n < \kappa S_n$ **then**

$U_n = \kappa S_n$;

$B_n = 0$;

end if

end for

for $m = 0$ to $M - 1$ **do**

$t = T - (m + 1)\Delta\tau$; {$T$ is the maturity, $\Delta\tau$ is the time interval}

calculate $AccI(t)$ using (2.25);

calculate $B_c(t)$ and $B_p(t)$ using (2.26) if applicable;

calculate B^{m+1} using the fully implicit method (3.38);

calculate U^{m+1} using B^{m+1} and the fully implicit method (3.37);

apply constraints to U^{m+1} and B^{m+1} explicitly using Algorithm 1;

call the Penalty iteration with the Crank-Nicolson method to compute U^{m+1};

$B^{m+1} = \min(B^{m+1}, U^{m+1})$;

if $t \in$ coupon payment period **then**

$U^{m+1} = U^{m+1} + K$;

$B^{m+1} = B^{m+1} + K$;

end if

end for

are doubled.

Table 4.1: Model parameters for the Convertible Bond without dividends

Parameter	Value
Time to maturity T	5 years
Conversion	0 to 5 years into 1 share
Conversion ratio κ	1.0
Face value F	100
Clean call price B_c^{cl}	110 from year 3 to year 5
Clean put price B_p^{cl}	105 during year 3
Coupon payments K	\$4.0
Coupon dates	.5, 1.0, 1.5, ..., 5.0
Risk-free interest rate r	5% or 0.05
Credit risk r_c	2% or 0.02
Volatility σ	20% or 0.20
Underlying stock price at $t = 0$ (S_{int})	100
Tolerance for PSOR Iteration ϵ	1.0e-06
Tolerance for Penalty Iteration tol	1.0e-06

The results in Tables 4.2 and 4.3 are for a convertible bond without dividends under the TF model, with parameter values given in Table 4.1. Table 4.2 shows the numerical results obtained using the PSOR method and the transformation $S = 100e^x$; Table 4.3 shows the numerical results obtained using the Penalty method.

Notice that we use the transformation $S = 100e^x$ for the PSOR method, while not for the penalty method. To make the results comparable, we choose the grid size used for these two methods such that the stepsizes near the grid point $S = 100$ are approximately

Table 4.2: Results for the PSOR method for a convertible bond without dividends under the TF model computed in the x variables with $S = 100e^x$

Time-steps	Grid Size				No. of Iterations			
(M)	(N)	Price	Diff	Ratio	max	min	avg	Time
100	288	124.11018940			11	6	8.2	0s
200	576	124.00570243	-0.10448697		15	8	9.8	1s
400	1152	123.98757448	-0.01812795	5.8	26	9	12.1	3s
800	2304	123.98069631	-0.00687817	2.6	24	10	14.4	10s
1600	4608	123.97358049	-0.00711581	1.0	36	12	18.3	59s
3200	9216	123.96759892	-0.00598157	1.2	62	14	23.2	299s
6400	18432	123.96639480	-0.00120412	5.0	110	17	29.7	1362s
12800	36864	123.96588647	-0.00050832	2.4	201	19	38.0	7008s

Table 4.3: Results for the penalty method for a convertible bond without dividends under the TF model computed in the S variables

Time-steps (M)	Grid Size (N)	Price	Diff	Ratio	No. of Iterations			Time
					max	min	avg	
100	100	124.02819396			4	1	1.7	0s
200	200	124.03535203	0.00715807		4	1	1.9	1s
400	400	123.98909449	-0.04625754	0.2	4	1	2.0	2s
800	800	123.98167193	-0.00742255	6.2	7	1	2.1	7s
1600	1600	123.97510793	-0.00656401	1.1	10	1	2.2	27s
3200	3200	123.96949235	-0.00561558	1.2	13	1	2.4	112s
6400	6400	123.96709697	-0.00239538	2.3	15	1	2.6	500s
12800	12800	123.96577303	-0.00132394	1.8	15	1	2.8	2186s

equal for both methods. Therefore, we have

$$\begin{aligned}
100e^{\Delta x} &= 100 + \Delta S \\
e^{\Delta x} &= 1 + \frac{\Delta S}{100} \\
1 + \Delta x &\approx 1 + \frac{\Delta S}{100} \qquad (\text{since } 1 + x \approx e^x \text{ for small } x) \\
\Delta x &\approx \frac{\Delta S}{100} \\
\frac{18}{N_x} &\approx \frac{5}{N_S} \\
N_x &\approx 3.6 N_S,
\end{aligned}$$

where N_x is the grid size in the x-dimension, and N_S the grid size in the S-dimension. The choice of grid size in Tables 4.2 and 4.3 roughly reflects the above relationship.

From Tables 4.2 and 4.3, we can see that as the time and grid step-sizes are reduced, both methods appear to be converging to a value which is approximately 123.965. In the following, we compare the numerical results for these two methods in more detail.

Comparison of the Convergence Ratio

Table 4.4 shows the average convergence ratio and its standard deviation obtained by the PSOR method (Table 4.2) and the penalty method (Table 4.3). From Tables 4.2, 4.3 and 4.4, we can see that both methods achieve about a first-order convergence rate with respect to $\Delta\tau$ on average, although Crank-Nicolson time stepping, which has second-order convergence with respect to $\Delta\tau$ for sufficiently smooth problems, is used as the underlying method for both the PSOR and the penalty iterations. We also see that for both methods, the order of convergence oscillates erratically, with standard deviation 1.98 for the PSOR method and 2.11 for the penalty method. Similar convergence rate oscillations for convertible bond pricing can also be observed in [1] and [9]. We believe that the reason for the convergence rate degeneration and oscillation comes from the discontinuities associated with the convertability, callability, puttability and discrete coupon payments

associated with the convertible bond. As far as the convergence ratio is concerned, we cannot claim that one of the two methods is substantially better than the other.

Table 4.4: Comparison of the convergence ratio for the PSOR and penalty methods (CB without dividends under the TF model)

Method	Average Ratio	Standard Deviation
PSOR method	3.0	1.98
Penalty method	2.1	2.11

Comparison of the Number of Iterations and Computation Time

Table 4.5 shows the comparison on the average number of iterations for each time-step and the computation time required by the PSOR method and the penalty method, respectively. The saved percentage for the average number of iterations for each time-step and the computation time is computed by

$$\frac{\text{PSOR Iters} - \text{Penalty Iters}}{\text{PSOR Iters}} \quad \text{or} \quad \frac{\text{PSOR Time} - \text{Penalty Time}}{\text{PSOR Time}}.$$

From Table 4.5, it is clear that the penalty method saves significantly in both the average number of iterations for each time-step and the overall computation time. The saving becomes more significant as the grid is refined. In terms of the number of iterations and the computation time, the penalty method has a significant advantage over the PSOR method.

Combining the above comparisons together, we can conclude that the penalty method is more efficient than the PSOR method, therefore we choose to use the penalty method in our numerical scheme to compute the price of a convertible bond with dividend protection in the following section.

Table 4.5: Comparison of the average number of iterations for each time-step and computation time for the PSOR and penalty methods (CB without dividends under the TF model)

	PSOR Method			Penalty Method			Percentage Saved	
Time-steps(M)	Price	Iters	Time	Price	Iters	Time	Iters	Time
100	124.11018940	8.2	0s	124.02819396	1.7	0s	79.3%	–
200	124.00570243	9.8	1s	124.03535203	1.9	1s	80.6%	0%
400	123.98757448	12.1	3s	123.98909449	2.0	2s	83.5%	33.3%
800	123.98069631	14.4	10s	123.98167193	2.1	7s	85.4%	30.0%
1600	123.97358049	18.3	59s	123.97510793	2.2	27s	88.0%	54.2%
3200	123.96759892	23.2	299s	123.96949235	2.4	112s	89.7%	62.5%
6400	123.96639480	29.7	1362s	123.96709697	2.6	500s	91.3%	63.3%
12800	123.96588647	38.0	7008s	123.96577303	2.8	2186s	92.6%	68.8%

4.2 Penalty Method for Convertible Bonds With Dividend Protection

In this section, we discuss how to apply the penalty method to convertible bonds with dividend protection under the TF and AFV models, respectively.

4.2.1 Under the TF Model

Conversion Ratio Adjustment

As described in Section 2.2.1, the *Conversion Ratio Adjustment* comes into play at the final/boundary conditions for the problem. Therefore, the algorithm is similar to the one for convertible bonds without dividends except that we need to apply appropriate conversion ratios to different time periods during the life of the convertible bond according to (2.18), and to handle discrete dividend payments on the underlying stock according to (2.20), (2.21), (2.22) and (2.23). The penalty iteration algorithm is the same as Algorithm 4. We give the pseudocode for the penalty time-stepping algorithm in Algorithm 6.

Dividend Pass-Thru

As described in Section 2.2.2, for *Dividend Pass-Thru*, the PDEs and boundary/final conditions are the same in this case as for a convertible bond without dividends. The only difference is that we treat the excess dividend payouts at dividend payment dates as coupon payments to the convertible bond. Therefore, the algorithm is similar to the one for convertible bonds without dividends except that we need to add the excess dividend payouts at dividend payment dates computed by (2.19) as coupon payments to the convertible bond according to (2.24), and to handle discrete dividend payments on the underlying stock according to (2.20), (2.21), (2.22) and (2.23). The penalty iteration algorithm is the same as Algorithm 4. We give the pseudocode for the penalty time-

Algorithm 6 Penalty time-stepping for a Convertible Bond with dividend protection using *Conversion Ratio Adjustment* under the TF model

$D = [D_0\ D_1\ \dots\ D_d]$; {discrete dividends on the underlying stock}

$\kappa = [\kappa_0\ \kappa_1\ \dots\ \kappa_d]$ computed by (2.18);{conversion ratios for different time periods}

$U = B = F + K$; {F: face value of the bond; K: coupon payment at maturity}

for $n = 0$ to N **do**

if $U_n < \kappa_d S_n$ **then**

$U_n = \kappa_d S_n$; $B_n = 0$;

end if

end for

for $m = 0$ to $M - 1$ **do**

$t = T - (m+1)\Delta\tau$; {T is the maturity, $\Delta\tau$ is the time interval}

calculate $AccI(t)$ using (2.25);

calculate $B_c(t)$ and $B_p(t)$ using (2.26) if applicable;

calculate B^{m+1} using the fully implicit method in (3.38);

calculate U^{m+1} using B^{m+1} and the fully implicit method in (3.37);

get corresponding κ for t from $[\kappa_0\ \kappa_1\ \dots\ \kappa_d]$;

apply constraints to U^{m+1} and B^{m+1} explicitly using Algorithm 1;

call the Penalty iteration with the Crank-Nicolson method to compute U^{m+1};

$B^{m+1} = \min(B^{m+1}, U^{m+1})$;

if $t \in$ coupon payment period **then**

$U^{m+1} = U^{m+1} + K$; $B^{m+1} = B^{m+1} + K$;

end if

if $t \in$ dividend payment period **then**

adjust U^{m+1} and B^{m+1} according to (2.21) and (2.22);

end if

end for

stepping algorithm in Algorithm 7.

4.2.2 Under the AFV Model

We first consider the penalty method for convertible bonds without dividends, and then extend the algorithm to handle convertible bonds with dividend protection using *Conversion Ratio Adjustment* and *Dividend Pass-Thru*, respectively.

For Convertible Bonds Without Dividends

We work with the transformed PDEs under the AFV model described in section 3.1. According to [1], the equations for the convertible bond value U in (2.7), (2.8) and (2.9) can be written (after transformation) as

$$\frac{\partial U}{\partial \tau} = \frac{\sigma^2 S^2}{2}\frac{\partial^2 U}{\partial S^2} + rS\frac{\partial U}{\partial S} - (r+p)U + p\max(\kappa S(1-\eta), RB) \tag{4.21}$$

subject to the constraints

$$U \geq \max(B_p, \kappa S), \tag{4.22}$$

$$U \leq \max(B_c, \kappa S). \tag{4.23}$$

We apply the penalty method to solve the above constrained problem. As for the TF model, we use the method described in [1] to decouple the resulting system involving U and B, as follows.

First, the value of B^{m+1} is estimated by solving (3.47), ignoring any constraints. This value of B^{m+1} is then used to estimate C^{m+1} by solving (3.48), ignoring any constraints. Then, we apply the constraints encompassed in the Linear Complimentarity Problems (2.10) and (2.11) explicitly to C^{m+1} and B^{m+1}. We then use the adjusted value of B^{m+1} for the constrained problem (4.21), (4.22) and (4.23), and apply the penalty method to solve for U^{m+1}.

Algorithm 7 Penalty time-stepping for a Convertible Bond with dividend protection using *Dividend Pass-Thru* under the TF model

$D = [D_0\ D_1\ \dots\ D_d]$; {discrete dividends on the underlying stock}

$K' = [K'_1\ K'_2\ \dots\ K'_d]$ computed by (2.19); {excess dividend payout on each dividend date}

$U = B = F + K$; {F: face value of the bond; K: coupon payment at maturity}

for $n = 0$ to N **do**

if $U_n < \kappa S_n$ **then**

$U_n = \kappa S_n$; $B_n = 0$;

end if

end for

for $m = 0$ to $M - 1$ **do**

$t = T - (m+1)\Delta\tau$; {T is the maturity, $\Delta\tau$ is the time interval}

calculate $AccI(t)$ using (2.25);

calculate $B_c(t)$ and $B_p(t)$ using (2.26) if applicable;

calculate B^{m+1} using the fully implicit method in (3.38);

calculate U^{m+1} using B^{m+1} and the fully implicit method in (3.37);

apply constraints to U^{m+1} and B^{m+1} explicitly using Algorithm 1;

call the Penalty iteration with the Crank-Nicolson method to compute U^{m+1};

$B^{m+1} = \min(B^{m+1}, U^{m+1})$;

if $t \in$ coupon payment period **then**

$U^{m+1} = U^{m+1} + K$; $B^{m+1} = B^{m+1} + K$;

end if

if $t \in$ dividend payment period **then**

get corresponding K' for t from $[K'_1\ K'_2\ \dots\ K'_d]$;

$U^{m+1} = U^{m+1} + K'$; $B^{m+1} = B^{m+1} + K'$;

adjust U^{m+1} and B^{m+1} according to (2.21) and (2.22);

end if

end for

Following the same approach as discussed above for the TF model, we can write the penalty scheme for the constrained problem (4.21), (4.22) and (4.23) as

$$\begin{aligned}\frac{\partial U}{\partial \tau} &= \frac{\sigma^2 S^2}{2}\frac{\partial^2 U}{\partial S^2} + rS\frac{\partial U}{\partial S} - (r+p)U + p\max(\kappa S(1-\eta), RB) \\ &\quad -\rho\max(U - U^*_{ceil}, 0) + \rho\max(U^*_{floor} - U, 0), \end{aligned} \tag{4.24}$$

where $\rho > 0$ is the penalty parameter, R the recovery rate, and the payoff functions U^*_{ceil} and U^*_{floor} are defined in (4.8).

Similarly as described above for the TF model, taking (3.46) into account, we can get the corresponding discretization equations in matrix form as

$$\begin{aligned}&[\mathbf{I} - \theta\mathbf{M}_{\mathbf{U}}^{\mathbf{AFV}} + \mathbf{P_{ceil}}(U^{m+1}) + \mathbf{P_{floor}}(U^{m+1})]U^{m+1} \\ =\ &[\mathbf{I} + (1-\theta)\mathbf{M}_{\mathbf{U}}^{\mathbf{AFV}}]U^m + p\Delta\tau[(1-\theta)\max(\kappa(1-\eta)S, RB^m) + \theta\max(\kappa(1-\eta)S, RB^{m+1})] \\ &+\mathbf{P_{ceil}}(U^{m+1})U^{*,m+1}_{ceil} + \mathbf{P_{floor}}(U^{m+1})U^{*,m+1}_{floor}, \end{aligned} \tag{4.25}$$

where $\mathbf{P_{ceil}}(U^{m+1})$ and $\mathbf{P_{floor}}(U^{m+1})$ are defined in (4.15) and (4.16).

Applying the generalized Newton's Method to solve the discrete non-linear equations (4.25), we get the following penalty iteration

$$\begin{aligned}&[\mathbf{I} - \theta\mathbf{M}_{\mathbf{U}}^{\mathbf{AFV}} + \mathbf{P_{ceil}}(U^{m+1,k}) + \mathbf{P_{floor}}(U^{m+1,k})]U^{m+1,k+1} \\ =\ &[\mathbf{I} + (1-\theta)\mathbf{M}_{\mathbf{U}}^{\mathbf{AFV}}]U^m + p\Delta\tau[(1-\theta)\max(\kappa(1-\eta)S, RB^m) + \theta\max(\kappa(1-\eta)S, RB^{m+1})] \\ &+\mathbf{P_{ceil}}(U^{m+1,k})U^{*,m+1}_{ceil} + \mathbf{P_{floor}}(U^{m+1,k})U^{*,m+1}_{floor}. \end{aligned} \tag{4.26}$$

The pseudocode for the penalty iteration and the penalty time-stepping algorithm is shown in Algorithms 8 and 9, respectively. Note that Algorithm 8 differs from Algorithm 4 only in the point where (4.26) is solved.

Conversion Ratio Adjustment

The handling of the *Conversion Ratio Adjustment* is similar to the corresponding case of the TF model. The penalty iteration algorithm is the same as Algorithm 8. The pseudocode for the penalty time-stepping algorithm is shown in Algorithm 10.

Algorithm 8 Penalty iteration for a Convertible Bond without dividends under the AFV model

$U^*_{ceil} = \max(B_c, \kappa S)$;

$U^*_{floor} = \max(B_p, \kappa S)$;

for $n = 0$ to N **do**

if $U_n^{m+1,0} > U^{*,m+1}_{ceil,n}$ **then**

$P_{ceil}(U_n^{m+1,0}) = \rho \Delta\tau$;

else

$P_{ceil}(U_n^{m+1,0}) = 0$;

end if

if $U_n^{m+1,0} < U^{*,m+1}_{floor,n}$ **then**

$P_{floor}(U_n^{m+1,0}) = \rho \Delta\tau$;

else

$P_{floor}(U_n^{m+1,0}) = 0$;

end if

end for

for $k = 0$, ..., until convergence **do**

solve (4.26);

update P_{ceil} and P_{floor} using $U^{m+1,k+1}$ by Equations (4.15) and (4.16);

if $\left[||U^{m+1,k+1} - U^{m+1,k}||_\infty \leq tol\right]$ or

$\left[P_{ceil}(U^{m+1,k+1}) == P_{ceil}(U^{m+1,k}) \text{ and } P_{floor}(U^{m+1,k+1}) == P_{floor}(U^{m+1,k})\right]$ **then**

exit from for loop;

end if

end for

$U^{m+1} = U^{m+1,k+1}$;

Algorithm 9 Penalty time-stepping for a Convertible Bond without dividends under the AFV model

$U = B = F + K$; {F: face value of the bond; K: coupon payment at maturity}

for $n = 0$ to N **do**

if $U_n < \kappa S_n$ **then**

$U_n = \kappa S_n$; $B_n = 0$;

end if

end for

for $m = 0$ to $M - 1$ **do**

$t = T - (m + 1)\Delta\tau$; {$T$ is the maturity, $\Delta\tau$ is the time interval}

calculate $AccI(t)$ using (2.25);

calculate $B_c(t)$ and $B_p(t)$ using (2.26) if applicable;

calculate B^{m+1} using the fully implicit method in (3.47);

calculate C^{m+1} using B^{m+1} and the fully implicit method in (3.48); {apply constraints to B^{m+1} and C^{m+1} explicitly:}

for $n = 0$ to N **do**

$B_n^{m+1} = \min(B_c, B_n^{m+1})$;

if $B_p > \kappa S_n$ **then**

$B_n^{m+1} = \max(B_n^{m+1}, B_p - C_n^{m+1})$;

else

$C_n^{m+1} = \max(\kappa S_n - B_n^{m+1}, C_n^{m+1})$;

end if

end for

$C^{m+1} = \min(C^{m+1}, \max(\kappa S, B_c) - B^{m+1})$;

call the Penalty iteration with the Crank-Nicolson scheme to compute U^{m+1};

$B^{m+1} = \min(B^{m+1}, U^{m+1})$; $C^{m+1} = U^{m+1} - B^{m+1}$;

if $t \in$ coupon payment period **then**

$U^{m+1} = U^{m+1} + K$; $B^{m+1} = B^{m+1} + K$;

end if

end for

Algorithm 10 Penalty time-stepping for a Convertible Bond with dividend protection using *Conversion Ratio Adjustment* under the AFV model

$D = [D_0\ D_1\ \ldots\ D_d]$; {discrete dividends on the underlying stock}

$\kappa = [\kappa_0\ \kappa_1\ \ldots\ \kappa_d]$ computed by (2.18); {conversion ratios for different time periods}

$U = B = F + K$; {F: face value of the bond; K: coupon payment at maturity}

for $n = 0$ to N **do**

if $U_n < \kappa_d S_n$ **then**

$U_n = \kappa_d S_n$; $B_n = 0$;

end if

end for

for $m = 0$ to $M - 1$ **do**

$t = T - (m+1)\Delta\tau$; {T is the maturity, $\Delta\tau$ is the time interval}

calculate $AccI(t)$ using (2.25);

calculate $B_c(t)$ and $B_p(t)$ using (2.26) if applicable;

calculate B^{m+1} using the fully implicit method in (3.47);

calculate C^{m+1} using B^{m+1} and the fully implicit method in (3.48);

get corresponding κ for t from $[\kappa_0\ \kappa_1\ \ldots\ \kappa_d]$;

apply constraints to B^{m+1} and C^{m+1} explicitly as in Algorithm 9;

call the Penalty iteration with the Crank-Nicolson scheme to compute U^{m+1};

$B^{m+1} = \min(B^{m+1}, U^{m+1})$; $C^{m+1} = U^{m+1} - B^{m+1}$;

if $t \in$ coupon payment period **then**

$U^{m+1} = U^{m+1} + K$; $B^{m+1} = B^{m+1} + K$;

end if

if $t \in$ dividend payment period **then**

adjust U^{m+1}, B^{m+1} and C^{m+1} according to (2.21), (2.22) and (2.23);

end if

end for

Dividend Pass-Thru

The handling of the *Dividend Pass-Thru* is similar to the corresponding case of the TF model. The penalty iteration algorithm is the same as Algorithm 8. The pseudocode for the penalty time-stepping algorithm is given in Algorithm 11.

Algorithm 11 Penalty time-stepping for a Convertible Bond with dividend protection using *Dividend Pass-Thru* under the AFV model

$D = [D_0\ D_1\ \ldots\ D_d]$; {discrete dividends on the underlying stock}

$K' = [K'_1\ K'_2\ \ldots\ K'_d]$ computed by (2.19); {excess dividend payout on each dividend date}

$U = B = F + K$; {F: face value of the bond; K: coupon payment at maturity}

for $n = 0$ to N **do**

if $U_n < \kappa S_n$ **then**

$U_n = \kappa S_n$; $B_n = 0$;

end if

end for

for $m = 0$ to $M - 1$ **do**

$t = T - (m+1)\Delta\tau$; {T is the maturity, $\Delta\tau$ is the time interval}

calculate $AccI(t)$ using (2.25);

calculate $B_c(t)$ and $B_p(t)$ using (2.26) if applicable;

calculate B^{m+1} using the fully implicit method in (3.47);

calculate C^{m+1} using B^{m+1} and the fully implicit method in (3.48);

apply constraints to B^{m+1} and C^{m+1} explicitly as in Algorithm 9;

call the Penalty iteration with the Crank-Nicolson scheme to compute U^{m+1};

$B^{m+1} = \min(B^{m+1}, U^{m+1})$; $C^{m+1} = U^{m+1} - B^{m+1}$;

if $t \in$ coupon payment period **then**

$U^{m+1} = U^{m+1} + K$; $B^{m+1} = B^{m+1} + K$;

end if

if $t \in$ dividend payment period **then**

get corresponding K' for t from $[K'_1\ K'_2\ \ldots\ K'_d]$;

$U^{m+1} = U^{m+1} + K'$; $B^{m+1} = B^{m+1} + K'$;{add excess dividend payout}

adjust U^{m+1}, B^{m+1} and C^{m+1} according to (2.21), (2.22) and (2.23);

end if

end for

Chapter 5

Numerical Results

In this chapter, we illustrate numerical results from the penalty method described in Chapter 4 under the TF and AFV models, respectively, for the following three cases: a convertible bond without dividends, a convertible bond with dividends but without dividend protection, and a convertible bond with dividends and with dividend protection. We will compare the results under the TF and AFV models, and present some plots based on these results.

The parameters for the Convertible Bond without dividends are shown in Table 4.1 in Chapter 4, and the parameters for the Convertible Bond with dividends are shown in Table 5.1. Recall that in the numerical results tables, "Price" is the convertible bond price at $t = 0$ and $S_{int} = 100$; "Grid Size" is the number of intervals in the spatial dimension; "Diff" is the difference in the solution from the coarser grid; "Ratio" is the ratio of successive differences; "No. of Iterations" is the number of iterations for each time-step; "max" is the maximum number of iterations over all time-steps; "min" is the minimum number of iterations; and "avg" is the average number of iterations for each time-step. At each refinement, the number of time-steps and the grid size are doubled.

Also notice that, we preset the dividends scheme in Table 5.1. This handling is only for research purpose. In practice, there is a designated team in the issuing company

responsible for predicting the future dividends scheme.

Table 5.1: Model parameters for the Convertible Bond with dividends

Parameter	Value
Time to maturity T	5 years
Conversion	0 to 5 years into κ shares
Conversion ratio κ	1.0
Face value F	100
Clean call price B_c^{cl}	110 from year 3 to year 5
Clean put price B_p^{cl}	105 during year 3
Coupon payments K	\$4.0
Coupon dates	.5, 1.0, 1.5, ..., 5.0
Dividends D_i	2, 3, 4, 4, 4
Dividend dates	0, 1.0, 2.0, 3.0, 4.0
Risk-free interest rate r	5% or 0.05
Credit risk r_c	2% or 0.02
Hazard rate p	2% or 0.02
Volatility σ	20% or 0.20
Underlying stock price at $t = 0$ (S_{int})	100
Tolerance for Penalty Iteration tol	1.0e-06

5.1 Results under the TF Model

Table 5.2 shows the numerical results obtained using the penalty method for a convertible bond without dividends. Table 5.3 shows the numerical results obtained using the penalty method for a convertible bond with dividends, but without dividend protection. Table

5.4 shows the numerical results obtained using the penalty method for a convertible bond with dividend protection via *Conversion Ratio Adjustment*. Table 5.5 shows the numerical results obtained using the penalty method for a convertible bond with dividend protection via *Dividend Pass-Thru*.

Table 5.2: Results of penalty method for a convertible bond without dividends under the TF model

Time-steps (M)	Grid Size (N)	Price	Diff	Ratio	No. of Iterations		
					max	min	avg
100	100	124.02819396			4	1	1.7
200	200	124.03535203	0.00715807		4	1	1.9
400	400	123.98909449	-0.04625754	0.2	4	1	2.0
800	800	123.98167193	-0.00742255	6.2	7	1	2.1
1600	1600	123.97510793	-0.00656401	1.1	10	1	2.2
3200	3200	123.96949235	-0.00561558	1.2	13	1	2.4
6400	6400	123.96709697	-0.00239538	2.3	15	1	2.6
12800	12800	123.96577303	-0.00132394	1.8	15	1	2.8

From Tables 5.2 - 5.5, we can see that as the time and grid step-sizes are reduced, for all cases, the numerical solutions appear to be converging to a final value. In all cases, the price in the last line of each table appears to be accurate to the 1-cent level. We can also observe that, for all cases, the average number of iterations per time-step is quite small, ranging from 1.7 to 3.2, which means that for typical grids and time step-sizes, the penalty method converges rapidly, and thus works efficiently. Also, as the time step-size is reduced, the average number of iterations per time-step is relatively stable. As for the convergence ratio, we can see that it often oscillates. Table 5.6 shows the average ratio and the standard deviation for each case.

Table 5.3: Results of penalty method for a convertible bond with dividends but without dividend protection under the TF model

Time-steps	Grid Size				No. of Iterations		
(M)	(N)	Price	Diff	Ratio	max	min	avg
100	100	119.21868575			5	1	1.9
200	200	119.12859464	-0.09009111		5	1	1.9
400	400	119.10405994	-0.02453470	3.7	5	1	1.9
800	800	119.10258054	-0.00147940	16.6	7	1	1.9
1600	1600	119.09395718	-0.00862336	0.2	12	1	1.9
3200	3200	119.08816203	-0.00579515	1.5	15	1	2.0
6400	6400	119.08562921	-0.00253282	2.3	19	1	2.0
12800	12800	119.08481504	-0.00081417	3.1	20	1	2.1

Table 5.4: Results of penalty method for a convertible bond with dividend protection via *Conversion Ratio Adjustment* under the TF model

Time-steps	Grid Size				No. of Iterations		
(M)	(N)	Price	Diff	Ratio	max	min	avg
100	100	120.25106182			3	1	1.7
200	200	120.15377813	-0.09728368		3	1	1.7
400	400	120.11342896	-0.04034918	2.4	4	1	1.7
800	800	120.09813870	-0.01529026	2.6	6	1	1.7
1600	1600	120.09001783	-0.00812087	1.9	9	1	1.8
3200	3200	120.08261291	-0.00740492	1.1	12	1	1.8
6400	6400	120.07958142	-0.00303149	2.4	14	1	1.9
12800	12800	120.07660169	-0.00297973	1.0	14	1	2.0

Table 5.5: Results of penalty method for a convertible bond with dividend protection via *Dividend Pass-Thru* under the TF model

Time-steps (M)	Grid Size (N)	Price	Diff	Ratio	No. of Iterations		
					max	min	avg
100	100	123.14311958			4	1	1.8
200	200	123.07521340	-0.06790618	0.0	4	1	2.0
400	400	123.06847768	-0.00673573	10.1	4	1	2.1
800	800	123.07564489	0.00716721	0.9	7	1	2.2
1600	1600	123.07725381	0.00160892	4.5	10	1	2.4
3200	3200	123.07849987	0.00124607	1.3	13	1	2.7
6400	6400	123.08055845	0.00205858	0.6	15	1	3.0
12800	12800	123.08115076	0.00059231	3.5	15	1	3.2

Table 5.6: Comparison of penalty method for different featured CBs on convergence ratio under the TF model

CBs	Average Ratio	Standard Deviation
CB without dividends	2.1	2.11
CB with dividends without dividend protection	4.6	6.02
CB with dividend protection (*Conversion Ratio Adjustment*)	1.9	0.70
CB with dividend protection (*Dividend Pass-Thru*)	3.5	3.59

From Table 5.6, we can see that the penalty method achieves roughly between first-order and second-order convergence with respect to $\Delta\tau$ on average, and the convergence ratio oscillates erratically. We conjecture that the following factors may be the reason: the discontinuities caused by the convertability, callability, puttability, discrete coupon payments and dividends associated with a convertible bond; and the use of a fully implicit scheme to get the estimate of the value of B before applying the penalty method to compute U for each time-step. As stated in Chapter 4, the convergence oscillation for convertible bonds pricing can also be observed in the works of other researchers; see, for example, [1] and [9].

To see the impact of dividend protection on the convertible bond value, we present the following plots. Figure 5.1 shows the convertible bond price without dividends, with dividends but without dividend protection, and with dividend protection via *Conversion Ratio Adjustment*, respectively, under the TF model for different underlying stock prices. Figure 5.2 is similar except that the dividend protection is via *Dividend Pass-Thru*. Figure 5.3 shows the difference between the convertible bond price without dividend protection and the one with dividend protection via *Conversion Ratio Adjustment* and *Dividend Pass-Thru*, respectively.

From Figures 5.1 and 5.2, we can see that the value of the convertible bond without dividends is higher than the one with dividends but without dividend protection as expected. The dividend protection either via *Conversion Ratio Adjustment* or via *Dividend Pass-Thru* increases the convertible bond value compared to the one without dividend protection. From Figure 5.3, we can see that the *Conversion Ratio Adjustment* and *Dividend Pass-Thru* results exhibit different trends on the increase in percentage of the convertible bond value compared to the one without protection: for the former, the increase is small for small S, increases as S increases and reaches the maximum of about 1.1% at $S \approx 140$, and then decreases after this point as S increases; for the latter, the increase is very large (around 4%) for small S, reaches a maximum of about 4.1% at

Figure 5.1: Price comparison for Convertible Bonds without dividends, with dividends but without dividend protection, and with dividend protection via *Conversion Ratio Adjustment* under the TF model

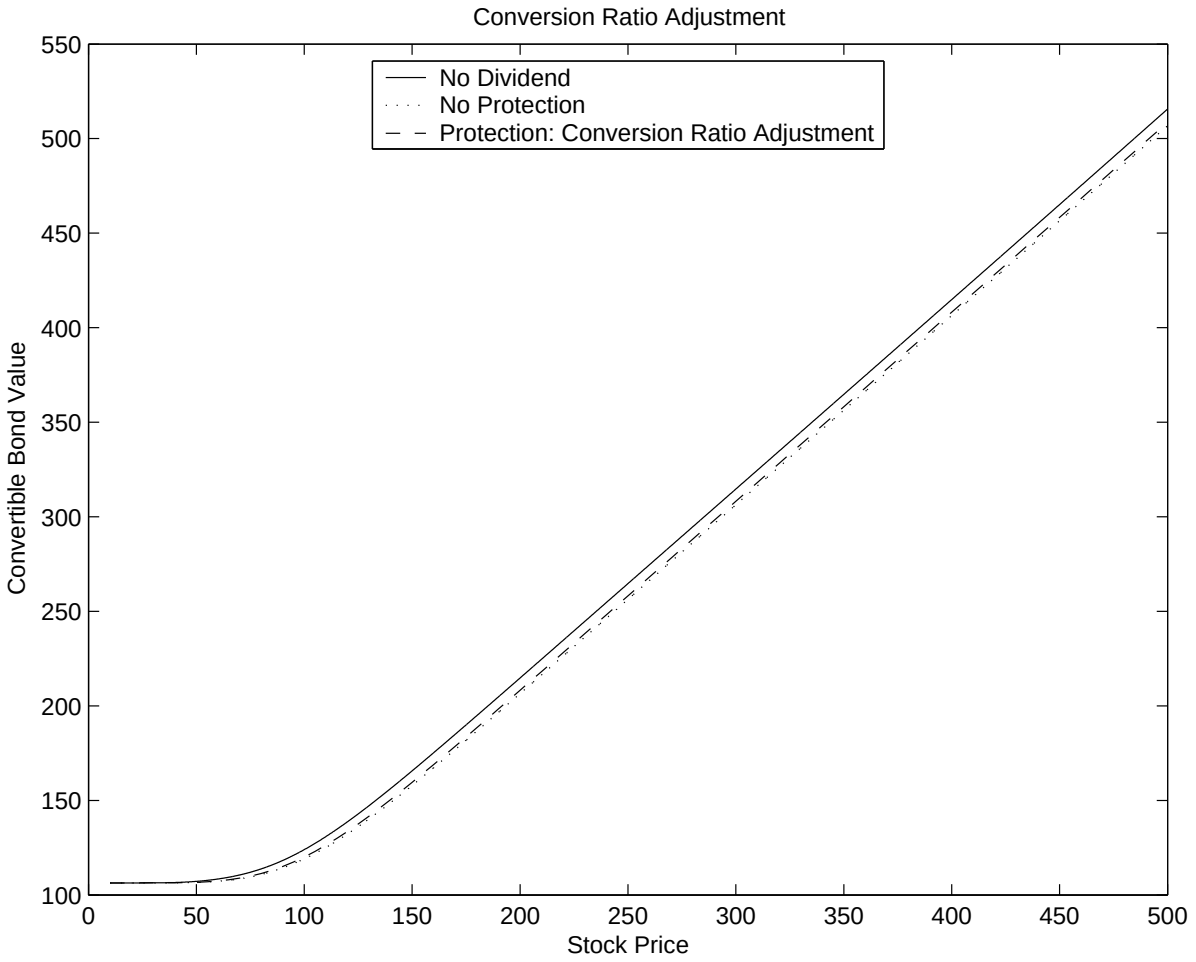


Figure 5.2: Price comparison for Convertible Bonds without dividends, with dividends but without dividend protection, and with dividend protection via *Dividend Pass-Thru* under the TF model

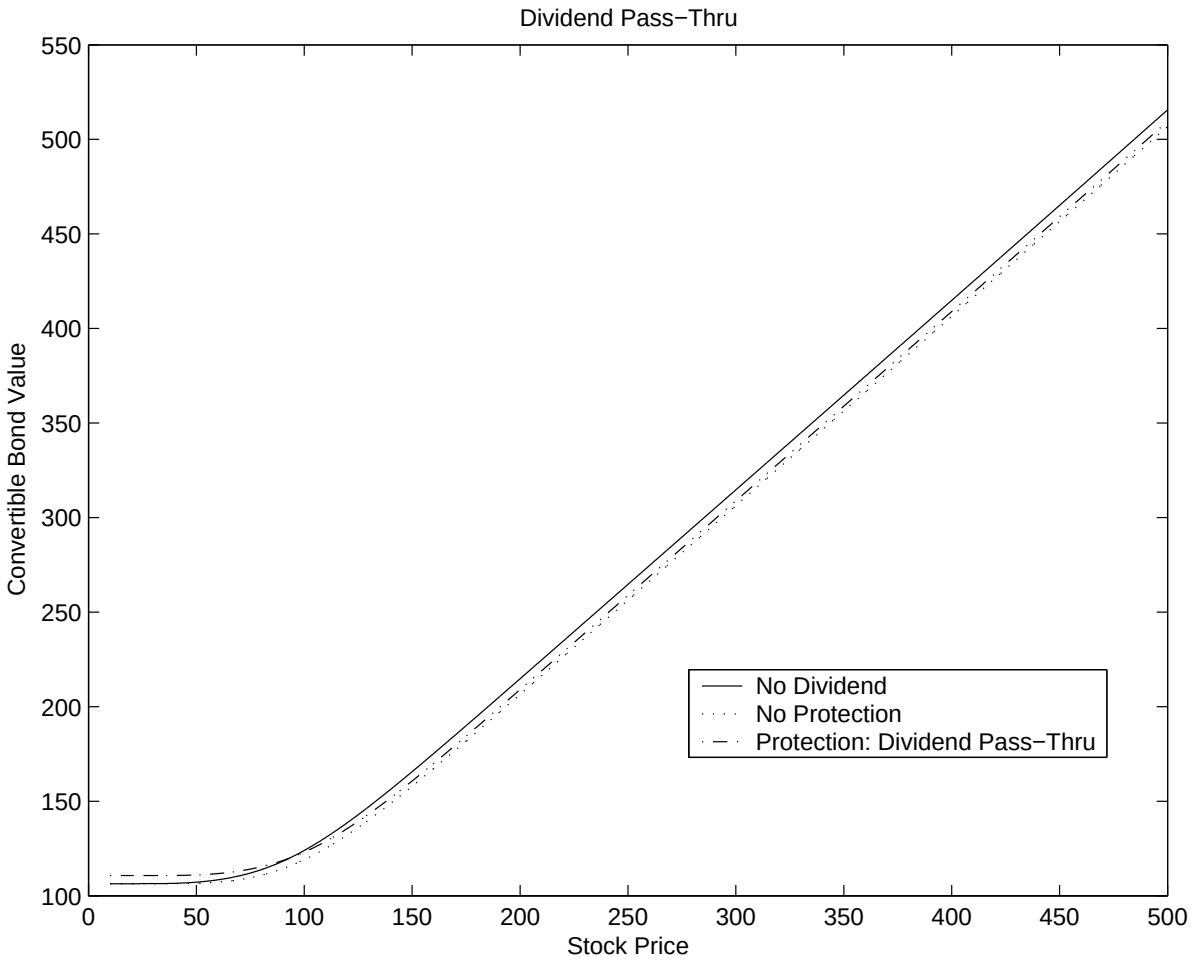
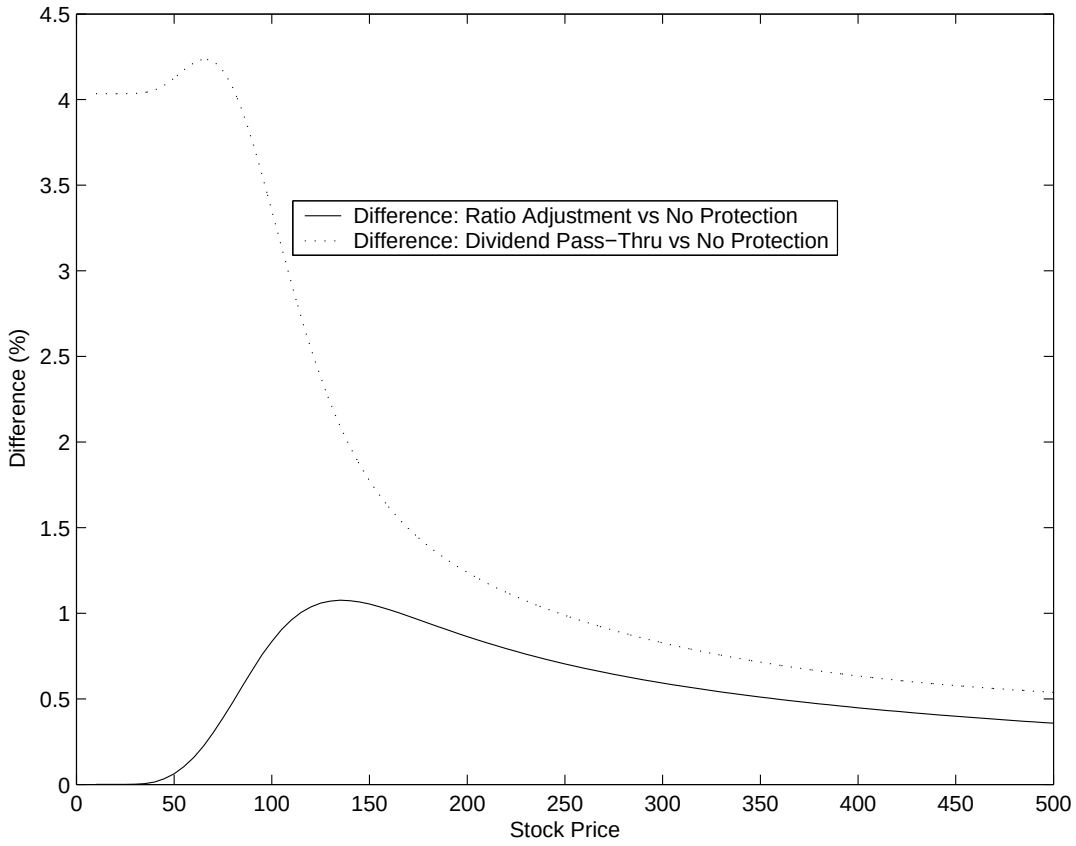


Figure 5.3: Price difference between Convertible Bonds without dividend protection and with dividend protection under the TF model



$S \approx 55$, and then decreases after this point as S increases. Overall, we find that dividend protection via *Dividend Pass-Thru* provides better protection than via *Conversion Ratio Adjustment* against future dividend payouts on the common stock. This is because *Dividend Pass-Thru* injects a new cash flow that did not exist before, *i.e.*, a windfall. It is interesting to observe that the value of the convertible bond with dividend protection via *Dividend Pass-Thru* is even higher than the one without dividends for small S in Figure 5.2. We believe that there are two reasons for this. First, for small S, a convertible bond behaves just like a straight bond and thus the extra dividend on the stock does not penalize the value of the embedded conversion option. Second, as stated earlier, *Dividend Pass-Thru* injects a new cash flow that did not exist before. This over-protection phenomenon for small S is also observed in [13].

5.2 Results under the AFV Model

Recall that according to the AFV model, specific assumptions are made about the behavior of the stock price on default, and recovery after default, *i.e.*, we have various choices of $\eta \in [0, 1]$ and $R \in [0, 1]$. In the following sections, we present numerical results obtained for different values of η and R.

5.2.1 Setting $\eta = 0$ and $R = 0$

As noted in Section 2.1.3, the TF model is a partial default model (the stock price does not jump upon default, and thus $\eta = 0$), and the PDEs governing the value of B under the TF and AFV models are exactly the same if $r_c = p(1 - R)$. Therefore, to make the results from the AFV model as comparable as possible to the ones obtained under the TF model, we set $\eta = 0$ and $R = 0$ (thus making $p = r_c$) for the AFV model.

Table 5.7 shows the numerical results obtained using the penalty method for a convertible bond without dividends. Table 5.8 shows the numerical results obtained using

the penalty method for a convertible bond with dividends, but without dividend protection. Table 5.9 shows the numerical results obtained using the penalty method for a convertible bond with dividend protection via *Conversion Ratio Adjustment*. Table 5.10 shows the numerical results obtained using the penalty method for a convertible bond with dividend protection via *Dividend Pass-Thru*.

Table 5.7: Numerical results for the penalty method for a convertible bond without dividends under the AFV model ($\eta = 0$ and $R = 0$)

Time-steps	Grid Size				No. of Iterations		
(M)	(N)	Price	Diff	Ratio	max	min	avg
100	100	124.88822145			4	1	1.9
200	200	124.91342626	0.02520481		4	1	2.0
400	400	124.91820687	0.00478061	5.3	4	1	2.1
800	800	124.91864116	0.00043430	11.0	6	1	2.2
1600	1600	124.91835971	-0.00028146	1.5	9	1	2.4
3200	3200	124.91810039	-0.00025932	1.1	12	1	2.7
6400	6400	124.91794182	-0.00015857	1.6	14	1	3.1
12800	12800	124.91789360	-0.00004822	3.3	15	1	3.3

From Tables 5.7 - 5.10, we can see that, as the time and grid step-sizes are reduced, each numerical solution appears to be converging to a final value. Moreover, the price in the last line of each table appears to be accurate to at least the 0.1-cent level. We can also observe that, for all cases, the average number of iterations per time-step is quite small, ranging from 1.8 to 3.8, which means that, for typical grids and time-steps, the penalty method converges rapidly per time-step, and thus works efficiently. Also, as the time step-size is reduced, the average number of iterations per time-step is relatively stable. As for the convergence ratio, we can see that it often oscillates in all cases, much as it

Table 5.8: Numerical results for the penalty method for a convertible bond with dividends but without dividend protection under the AFV model ($\eta = 0$ and $R = 0$)

Time-steps (M)	Grid Size (N)	Price	Diff	Ratio	No. of Iterations		
					max	min	avg
100	100	120.91900823			6	1	1.9
200	200	120.86225303	-0.05675520		8	1	2.0
400	400	120.85031078	-0.01194225	4.8	5	1	2.1
800	800	120.84498418	-0.00532660	2.2	10	1	2.1
1600	1600	120.84333085	-0.00165334	3.2	9	1	2.3
3200	3200	120.84253260	-0.00079825	2.1	12	1	2.4
6400	6400	120.84220182	-0.00033077	2.4	14	1	2.7
12800	12800	120.84212648	-0.00007534	4.4	15	1	2.9

Table 5.9: Numerical results for the penalty method for a convertible bond with dividend protection via *Conversion Ratio Adjustment* under the AFV model ($\eta = 0$ and $R = 0$)

Time-steps (M)	Grid Size (N)	Price	Diff	Ratio	No. of Iterations		
					max	min	avg
100	100	121.85920559			4	1	1.8
200	200	121.76751971	-0.09168588		4	1	1.9
400	400	121.75351681	-0.01400290	6.5	4	1	2.0
800	800	121.74696831	-0.00654850	2.1	6	1	2.1
1600	1600	121.74471433	-0.00225397	2.9	9	1	2.2
3200	3200	121.74391364	-0.00080069	2.8	12	1	2.4
6400	6400	121.74358547	-0.00032817	2.4	14	1	2.6
12800	12800	121.74350360	-0.00008187	4.0	15	1	2.8

Table 5.10: Numerical results for the penalty method for a convertible bond with dividend protection via *Dividend Pass-Thru* under the AFV model ($\eta = 0$ and $R = 0$)

Time-steps	Grid Size				No. of Iterations		
(M)	(N)	Price	Diff	Ratio	max	min	avg
100	100	124.21280355			4	1	1.9
200	200	124.15998867	-0.05281489		4	1	2.1
400	400	124.15176246	-0.00822621	6.4	4	1	2.2
800	800	124.14842095	-0.00334151	2.5	6	1	2.4
1600	1600	124.14786001	-0.00056094	6.0	9	1	2.6
3200	3200	124.14762734	-0.00023267	2.4	12	1	2.9
6400	6400	124.14757991	-0.00004743	4.9	14	1	3.4
12800	12800	124.14755784	-0.00002207	2.1	15	1	3.8

did for the TF model. Table 5.11 shows the average ratio and the standard deviation for each case.

From Table 5.11, we can see that the penalty method achieves roughly between first-order and second-order convengence with respect to $\Delta\tau$ on average, and the convergence ratio oscillates erratically. We believe the reason for this is similar to the one stated for the TF model.

Figure 5.4 shows the convertible bond price without dividends, with dividends but without dividend protection, and with dividend protection via *Conversion Ratio Adjustment*, respectively, for different underlying stock prices. Figure 5.5 is similar except that the dividend protection is via *Dividend Pass-Thru*. Figure 5.6 shows the difference between the convertible bond price without dividend protection and the one with dividend protection via *Conversion Ratio Adjustment* and *Dividend Pass-Thru*, respectively.

We found that Figures 5.4 to 5.6 exhibit similar phenomena to those observed in

Figure 5.4: Price comparison for Convertible Bonds without dividends, with dividends but without dividend protection, and with dividend protection via *Conversion Ratio Adjustment* under the AFV model ($\eta = 0$ and $R = 0$)

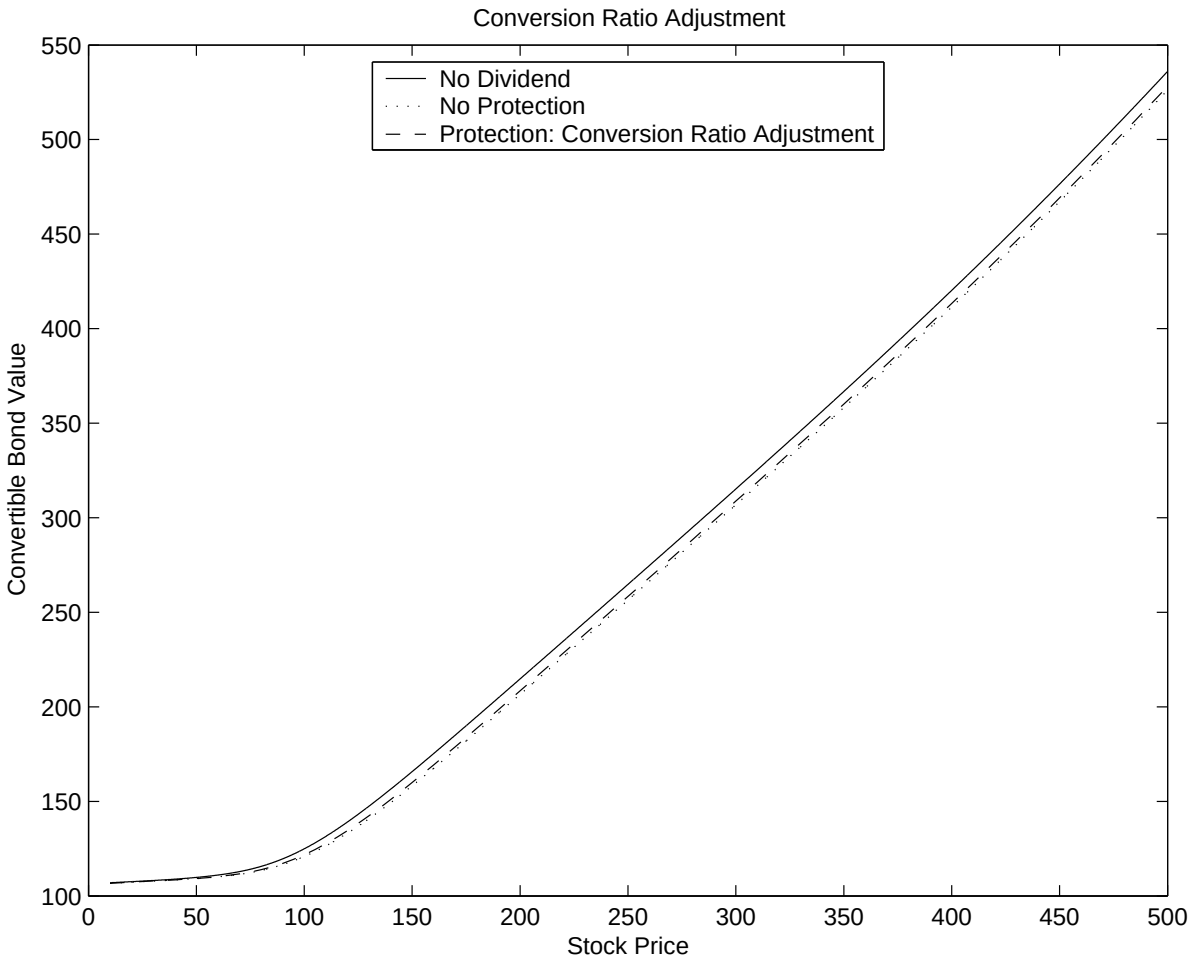


Figure 5.5: Price comparison for Convertible Bonds without dividends, with dividends but without dividend protection, and with dividend protection via *Dividend Pass-Thru* under the AFV model ($\eta = 0$ and $R = 0$)

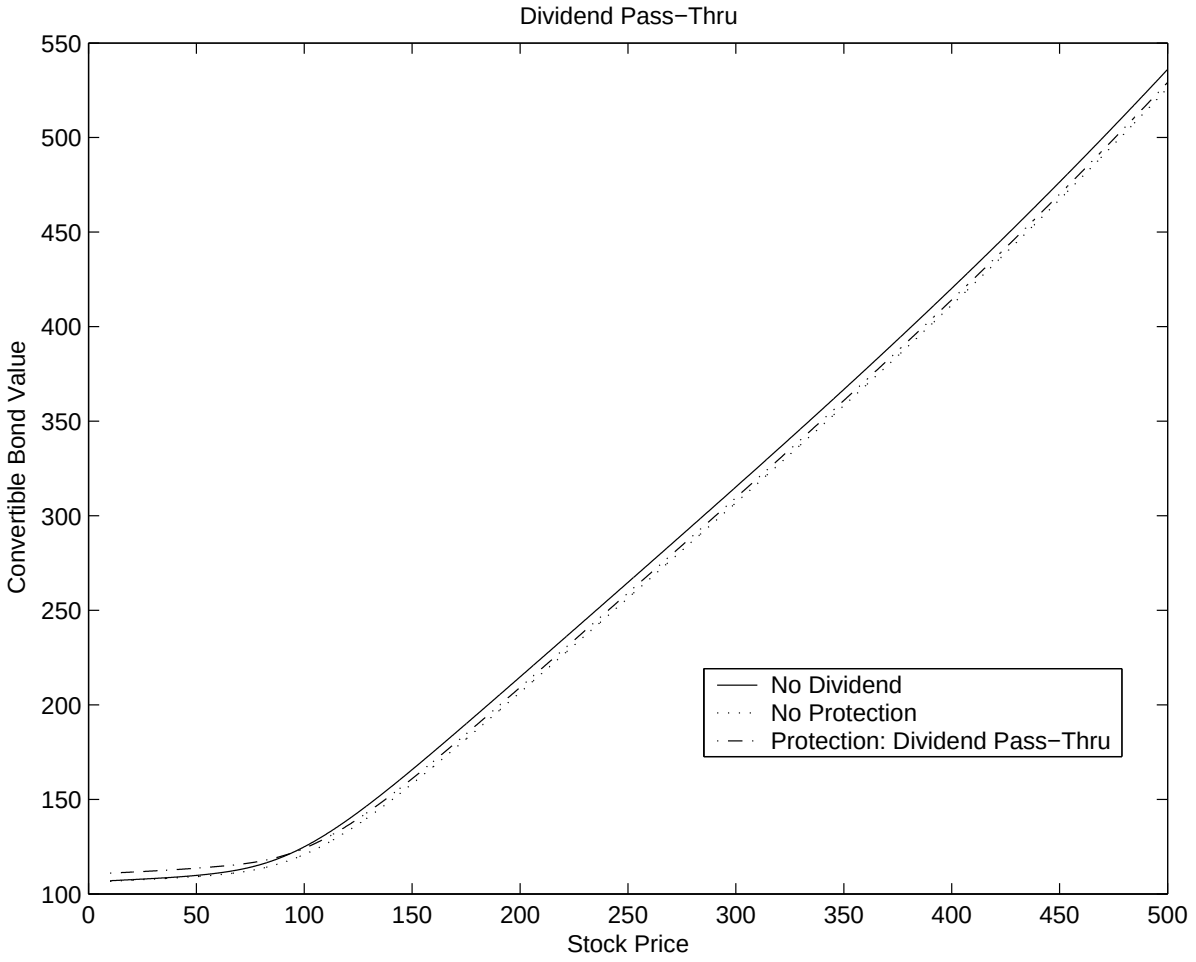


Figure 5.6: Price difference between Convertible Bonds without dividend protection and with dividend protection under the AFV model ($\eta = 0$ and $R = 0$)

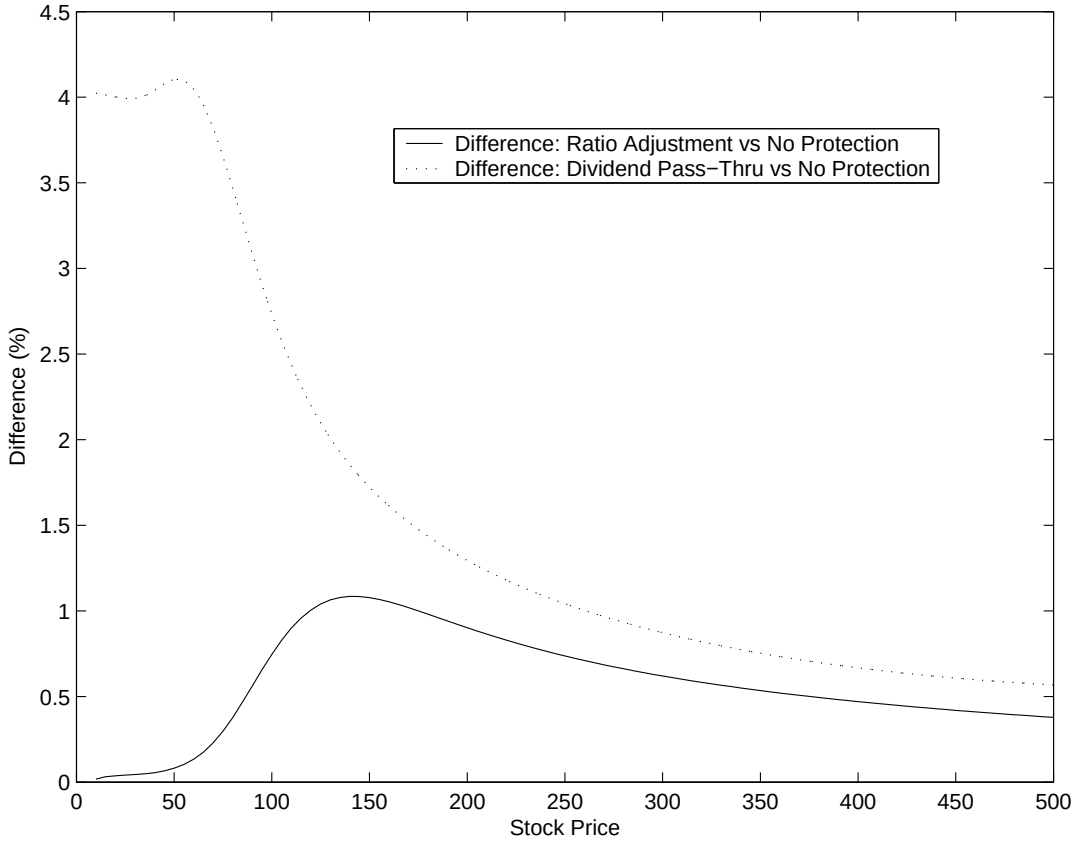


Table 5.11: Comparison of the convergence ratio for the penalty method for CBs with different dividend features under the AFV model ($\eta = 0$ and $R = 0$)

CBs	Average Ratio	Standard Deviation
CB without dividends	4.0	3.78
CB with dividends without dividend protection	3.2	1.17
CB with dividend protection (*Conversion Ratio Adjustment*)	3.5	1.63
CB with dividend protection (*Dividend Pass-Thru*)	4.1	1.95

Figures 5.1 to 5.3 obtained under the TF model.

5.2.2 Setting $\eta = 1$ and $R = 0.5$

To see the performance of the penalty method for different settings of η and R under the AFV model, we set $\eta = 1$ and $R = 0.5$, *i.e.*, we assume that the stock price jumps to zero upon default and the recovery rate is 0.5 after default. In Tables 5.12 to 5.15, we present the numerical results for the four cases mentioned earlier under the AFV model with $\eta = 1$ and $R = 0.5$.

From Tables 5.12 - 5.15, we can see similar behavior to the setting of $\eta = 0$ and $R = 0$. As the time and grid step-sizes are reduced, each numerical solution appears to be converging to a final value. Moreover, the price in the last line of each table appears to be accurate to at least the 0.1-cent level. For all cases, the penalty method converges rapidly per time-step. In addition, the average number of iterations per time-step is quite small, ranging from 1.3 to 3.1; as the time step-size is reduced, the average number of iterations per time-step is relatively stable, but the convergence ratio often oscillates

Table 5.12: Numerical results for the penalty method for a convertible bond without dividends under the AFV model ($\eta = 1$ and $R = 0.5$)

Time-steps (M)	Grid Size (N)	Price	Diff	Ratio	No. of Iterations		
					max	min	avg
100	100	127.97396334			3	1	1.4
200	200	128.00504688	0.03108354		3	1	1.5
400	400	128.01244098	0.00739410	4.2	4	1	1.6
800	800	128.01406851	0.00162752	4.5	6	1	1.8
1600	1600	128.01437732	0.00030882	5.3	9	1	2.0
3200	3200	128.01440547	0.00002815	11.0	13	1	2.3
6400	6400	128.01438972	-0.00001576	1.8	14	1	2.8
12800	12800	128.01437698	-0.00001273	1.2	14	1	3.1

Table 5.13: Numerical results for the penalty method for a convertible bond with dividends but without dividend protection under the AFV model ($\eta = 1$ and $R = 0.5$)

Time-steps (M)	Grid Size (N)	Price	Diff	Ratio	No. of Iterations		
					max	min	avg
100	100	123.80720808			3	1	1.4
200	200	123.75014376	-0.05706432		3	1	1.4
400	400	123.74035011	-0.00979366	5.8	4	1	1.5
800	800	123.73616887	-0.00418123	2.3	6	1	1.7
1600	1600	123.73519223	-0.00097664	4.3	9	1	1.9
3200	3200	123.73475525	-0.00043699	2.2	13	1	2.1
6400	6400	123.73461316	-0.00014209	3.1	14	1	2.5
12800	12800	123.73455692	-0.00005623	2.5	14	1	2.9

Table 5.14: Numerical results for the penalty method for a convertible bond with dividend protection via *Conversion Ratio Adjustment* under the AFV model ($\eta = 1$ and $R = 0.5$)

Time-steps (M)	Grid Size (N)	Price	Diff	Ratio	No. of Iterations		
					max	min	avg
100	100	124.85010965			3	1	1.4
200	200	124.76148924	-0.08862040		3	1	1.4
400	400	124.74955829	-0.01193096	7.4	4	1	1.5
800	800	124.74426437	-0.00529391	2.3	6	1	1.7
1600	1600	124.74275811	-0.00150626	3.5	9	1	1.9
3200	3200	124.74232206	-0.00043605	3.5	13	1	2.1
6400	6400	124.74217947	-0.00014259	3.1	14	1	2.5
12800	12800	124.74212179	-0.00005768	2.5	14	1	2.8

Table 5.15: Numerical results for the penalty method for a convertible bond with dividend protection via *Dividend Pass-Thru* under the AFV model ($\eta = 1$ and $R = 0.5$)

Time-steps (M)	Grid Size (N)	Price	Diff	Ratio	No. of Iterations		
					max	min	avg
100	100	126.78643871			3	1	1.3
200	200	126.72922621	-0.05721250		3	1	1.4
400	400	126.72048433	-0.00874188	6.5	4	1	1.5
800	800	126.71689842	-0.00358591	2.4	6	1	1.6
1600	1600	126.71631899	-0.00057944	6.2	9	1	1.7
3200	3200	126.71607846	-0.00024052	2.4	13	1	2.0
6400	6400	126.71603444	-0.00004402	5.5	14	1	2.3
12800	12800	126.71601550	-0.00001894	2.3	14	1	2.6

erratically. Table 5.16 shows the average ratio and the standard deviation for each case.

Table 5.16: Comparison of the convergence ratio for the penalty method for CBs with different dividend features under the AFV model ($\eta = 1$ and $R = 0.5$)

CBs	Average Ratio	Standard Deviation
CB without dividends	4.7	3.49
CB with dividends without dividend protection	3.4	1.42
CB with dividend protection (*Conversion Ratio Adjustment*)	3.7	1.87
CB with dividend protection (*Dividend Pass-Thru*)	4.2	2.05

From Table 5.16, we can see that the penalty method achieves roughly between first-order and second-order convengence with respect to $\Delta\tau$ on average, and the convergence ratio oscillates erratically.

5.3 Comparison of Numerical Results for the TF and AFV models

As stated earlier in this chapter, the TF model is similar to the partial default AFV model ($\eta = 0$) with recovery rate $R = 0$. In this section, we compare the numerical results for a convertible bond with dividend protection using the TF and AFV models (with $\eta = 0$ and $R = 0$), respectively.

Table 5.17 shows the numerical results for a convertible bond with dividend protection under both models. We can see that for both *Conversion Ratio Adjustment* and *Dividend Pass-Thru*, the TF model achieves solutions correct to about \$0.01, while the AFV model

can achieve solutions correct to about \$0.0001, on the finest grid smallest time step listed in the table. We can conclude from this that, to achieve the same level of accuracy, the TF model needs considerably finer grids and/or smaller time steps than the AFV model. We believe that this is because the bond component of the TF model has a time-dependent knock-out barrier introduced by the callability constraint (see Equation (2.5)), which is difficult to compute accurately [1]. We can also see that the AFV model (with $\eta = 0$ and $R = 0$) gives a price about \$1.00 higher than the TF price, which agrees with a similar observation in [1] for the price of a convertible bond without dividends using the TF and AFV models.

Table 5.17: Comparison of the numerical results for the TF and AFV models ($\eta = 0$ and $R = 0$) both using the penalty method for a convertible bond with dividend protection

Time-steps	Grid Size	*Conversion Ratio Adjustment*		*Dividend Pass-Thru*	
(M)	(N)	TF	AFV	TF	AFV
100	100	120.25106182	121.85920559	123.14311958	124.21280355
200	200	120.15377813	121.76751971	123.07521340	124.15998867
400	400	120.11342896	121.75351681	123.06847768	124.15176246
800	800	120.09813870	121.74696831	123.07564489	124.14842095
1600	1600	120.09001783	121.74471433	123.07725381	124.14786001
3200	3200	120.08261291	121.74391364	123.07849987	124.14762734
6400	6400	120.07958142	121.74358547	123.08055845	124.14757991
12800	12800	120.07660169	121.74350360	123.08115076	124.14755784

Table 5.18 shows the average convergence ratio and the associated standard deviation for both models. From Table 5.18, we can see that, for both the *Conversion Ratio Adjustment* and *Dividend Pass-Thru*, the AFV model achieves a higher convergence rate with respect to $\Delta\tau$ on average than the TF model, but the convergence ratio oscillates

erratically for both models.

Table 5.18: Comparison of the convergence ratio for a convertible bond with dividend protection under the TF and AFV models ($\eta = 0$ and $R = 0$) both using the penalty method

	Conversion Ratio Adjustment		*Dividend Pass-Thru*	
	TF	AFV	TF	AFV
Average Ratio	1.9	3.5	3.5	4.1
Standard Deviation	0.70	1.63	3.59	1.95

As described in previous sections, for both models, both *Conversion Ratio Adjustment* and *Dividend Pass-Thru* increase the convertible bond value compared to a similar CB without dividend protection. Moreover, the latter offers better protection. However, it is worth noting that we cannot say that *Dividend Pass-Thru* is preferable from the issuers' point of view when they choose the type of dividend protection to offer the convertible bond holders at the time of issuing, since *Dividend Pass-Thru* will erode the earnings of the issuing company to pay the convertible bond holders. Consequently, it may not be fair to the existing equity share holders [13]. Although *Dividend Pass-Thru* increases the convertible bond value more than *Conversion Ratio Adjustment*, this increase comes at a cost: the issuer may increase its future cash outflows to pay the convertible bond holders the excess dividends. This is not the case for *Conversion Ratio Adjustment*. Therefore, in practice, it is not an easy decision for issuers to choose between these two types of dividend protection.

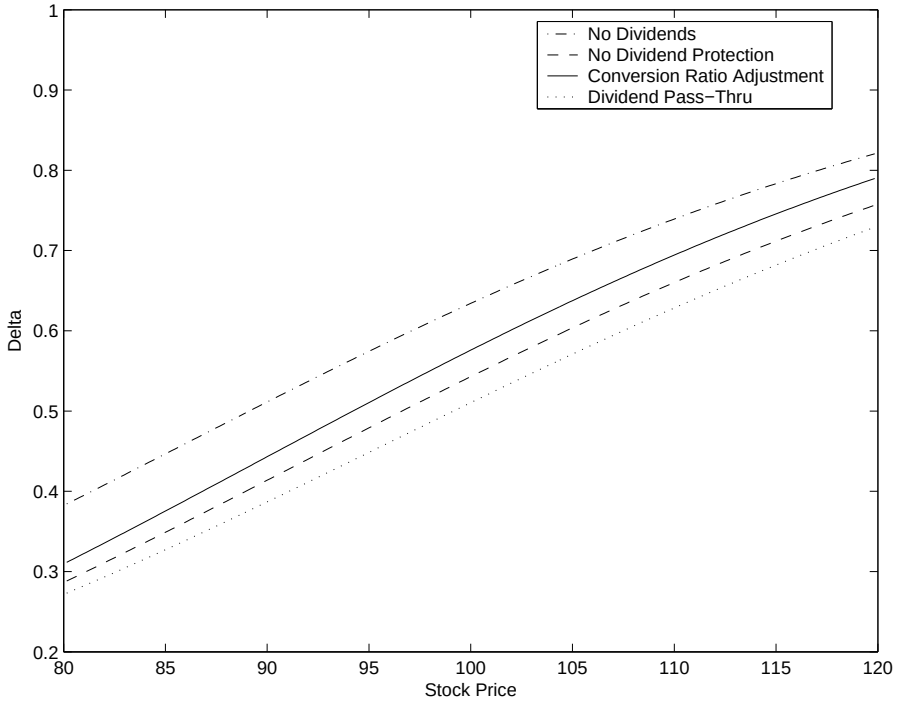
It is important in practice to determine delta and gamma for hedging purposes [12]. Figure 5.7 shows the plots of delta and gamma around $S = 100$ at $t = 0$ for similar convertible bonds with different dividend features under the TF model using the penalty method: without dividends, with dividends but without dividend protection, with div-

idend protection via *Conversion Ratio Adjustment*, and with dividend protection via *Dividend Pass-Thru*. In Figure 5.8, we display similar plots of delta and gamma under the AFV model using the penalty method. In these plots, delta and gamma are approximated by

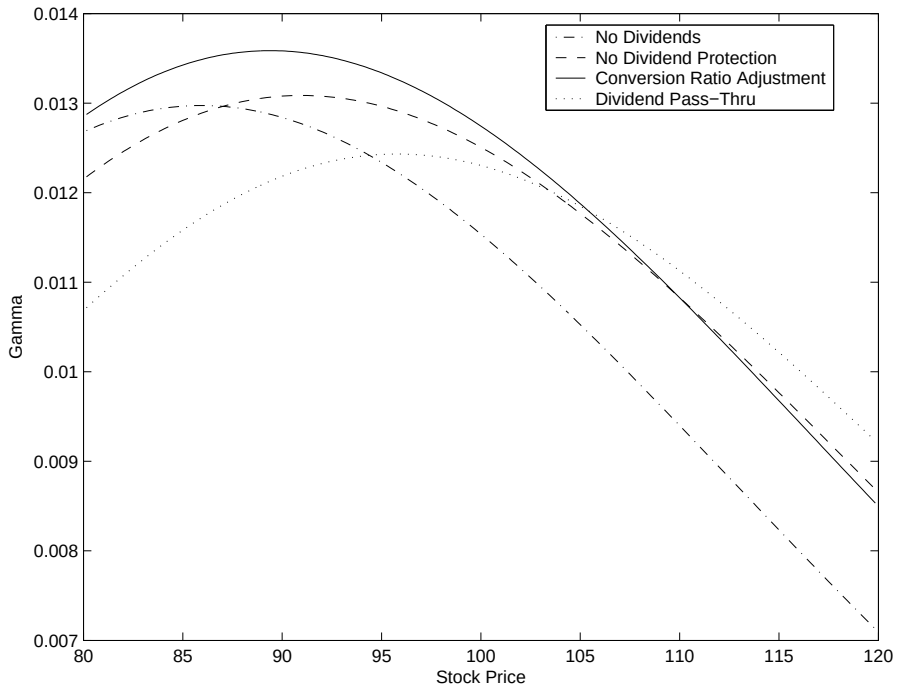
$$\begin{aligned}\left[\frac{\partial U}{\partial S}\right]_{t=0,S=S_n} &\approx \frac{U^M_{n+1} - U^M_{n-1}}{2\Delta S},\\ \left[\frac{\partial^2 U}{\partial S^2}\right]_{t=0,S=S_n} &\approx \frac{U^M_{n-1} - 2U^M_n + U^M_{n+1}}{\Delta S^2},\end{aligned}$$

respectively. When computing delta and gamma, we use the numerical results from the mesh with $(M, N) = (3200, 3200)$. From Figures 5.7 and 5.8, we can see that, for both models, the penalty method results in smooth delta and gamma values for all cases.

Figure 5.7: Delta and Gamma for convertible bonds with different dividend features under the TF model using the penalty method

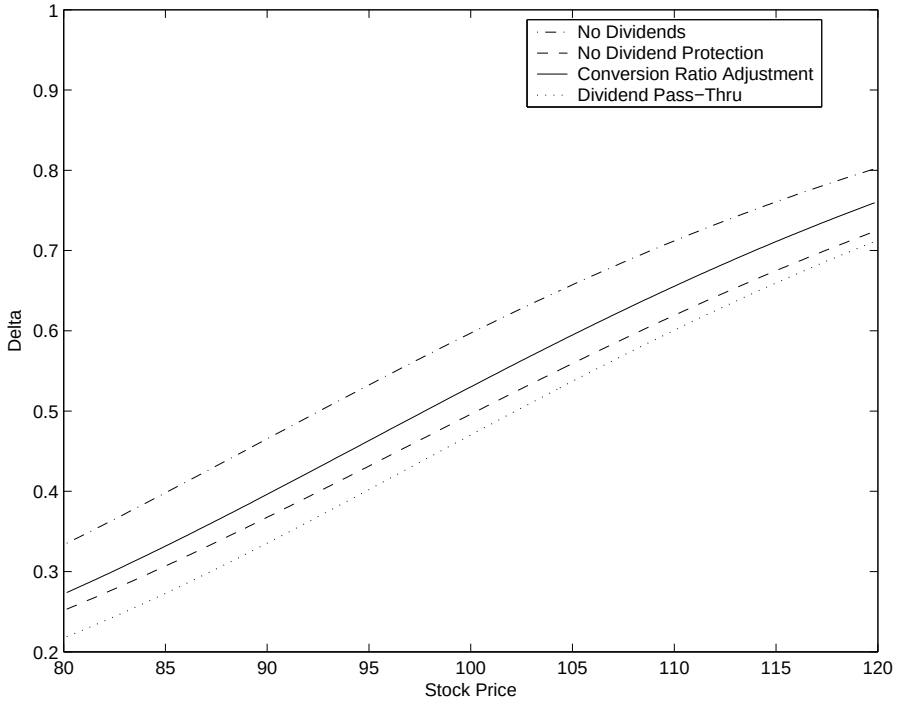


(a) Delta

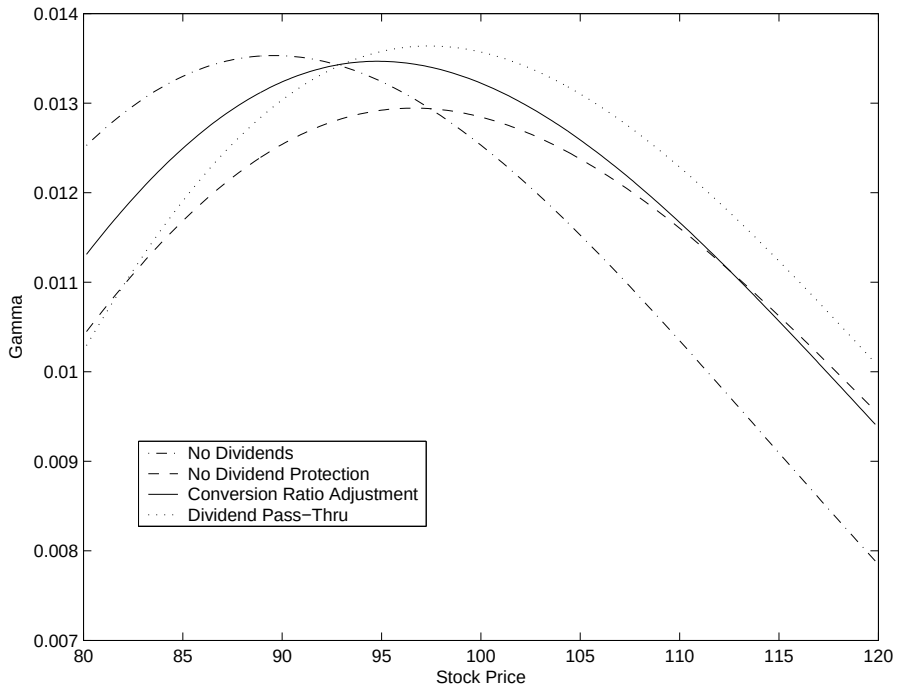


(b) Gamma

Figure 5.8: Delta and Gamma for convertible bonds with different dividend features under the AFV model using the penalty method



(a) Delta



(b) Gamma

Chapter 6

Conclusions and Future Work

Convertible bonds are a popular financial instrument with complex behavior. It is very difficult to model and evaluate convertible bonds, especially when taking credit risk into consideration. In recent years, a new convertible bond feature, dividend protection has emerged and become more and more popular. This new protection provision also makes the convertible bond valuation more complicated. In this paper, we attempt to use a numerical PDE approach to price convertible bonds with dividend protection subject to credit risk.

We considered two existing models, namely the TF and AFV models, and extended them to incorporate the two major dividend protection methods, *Conversion Ratio Adjustment* and *Dividend Pass-Thru*, respectively. Having developed a pricing model for convertible bonds with dividend protection, we sought a fast and robust numerical algorithm to implement the model. We considered two iterative methods, the PSOR and penalty methods, for this purpose. First, we compared the PSOR and penalty methods for pricing a convertible bond without dividends under the extended TF model in terms of the convergence ratio, number of iterations and computation time. We found that both methods achieve about first-order convergence rate with respect to time step-size $\Delta\tau$ on average, and the order of convergence oscillates erratically for both methods. However,

the penalty method saves many iterations and much computation time compared to the PSOR method, and the saving becomes more significant as the grid is refined. Therefore, we choose to apply the penalty method to the convertible bond with dividend protection.

We studied the numerical results for the TF and AFV models both using the penalty method for the following three cases: a convertible bond without dividends, a convertible bond with dividends but without dividend protection, and a convertible bond with dividends and with dividend protection. In particular, we compared the numerical results for a convertible bond with dividend protection via *Conversion Ratio Adjustment* and *Dividend Pass-Thru* using the TF and AFV models, respectively. We found that, for both *Conversion Ratio Adjustment* and *Dividend Pass-Thru*, the AFV model achieves more accurate solutions than the TF model using the same fine grid and small time step; and the AFV model achieves a higher convergence rate with respect to $\Delta\tau$ on average than the TF model, but the convergence ratio oscillates erratically for both models. We believe this is because of the discontinuities associated with the convertibility, callability, puttability and discrete coupon payments of the convertible bond. We also observed that, for both models, both *Conversion Ratio Adjustment* and *Dividend Pass-Thru* increase the convertible bond value compared to a similar CB without dividend protection, and the latter offers better protection but over-protects for small S. In addition, the penalty method results in smooth delta and gamma for both models.

As for the future work, there are two major issues. First, from the numerical results, we can see that the convergence rate of our proposed numerical algorithm for pricing convertible bonds with dividend protection often oscillates erratically. Further research needs to be conducted to make it stable. Second, we apply the penalty method to remove the free boundary associated with the convertible bond pricing problem, but the penalty term is discrete in the current approach. In the future, we may consider using a continuous penalty term for the penalty method.

Bibliography

[1] E. Ayache, P. A. Forsyth, and K. R. Vetzal. The valuation of convertible bonds with credit risk. *Journal of Derivatives*, fall(11):9–29, 2003.

[2] Martin Baxter. *Financial calculus: an introduction on derivative pricing*. Cambridge University Press, 1996.

[3] Michael J. Brennan and Eduardo S. Schwartz. Convertible bonds: Valuation and optimal strategies for call and conversion. *Journal of Finance*, 32:1699–1715, 1977.

[4] Michael J. Brennan and Eduardo S. Schwartz. Analyzing convertible bonds. *Journal of Financial and Quantitative Analysis*, 15:907–929, 1980.

[5] W. Cheung and I. Nelken. Costing the converts. *RISK*, July:47–49, 1994.

[6] Y. D'Halluin, P. A. Forsyth, K. R. Vetzal, and G. Labahn. A numerical PDE approach for pricing callable bonds. *Applied Mathematical Finance*, 8:49–77, 2001.

[7] P. A. Forsyth and K. R. Vetzal. Quadratic convergence for valuing American options using a penalty method. *SIAM J. Sci. Comput.*, 23(6):2095–2122, 2002.

[8] Lawrence Galitz. *Financial engineering: tools and techniques to manage financial risk*. IRWIN Professional Publishing, 1995.

[9] A. J. Grau, P. A. Forsyth, and K. R. Vetzal. Convertible bonds with call notice periods. *Working paper, University of Waterloo*, 2003.

[10] Russell Grimwood and Stewart Hodges. The valuation of convertible bonds: A study of alternative pricing models. *Warwick Business School Pre-print PP02-121,* `http://www2.warwick.ac.uk/`, 2002.

[11] Thomas S. Y. Ho and David M. Pfeffer. Convertible bonds: Model, value attribution, and analytics. *Financial Analysts Journal*, Sept/Oct(52):35–44, 1996.

[12] John Hull. *Options, Futures, and Other Derivatives (5th edition).* Prentice Hall, 2002.

[13] Kynex Inc. Dividend protection. *KYNEX Bulletin,* `http://www.kynex.com/Bulletin/Sep2004/main.htm`, September, 2004.

[14] J. Ingersoll. A contingent-claims valuation of convertible securities. *Journal of Financial Economics*, 4:289–322, 1977.

[15] R. Jarrow and S. Turnbull. Pricing derivative securities on financial securities subject to credit risk. *Journal of Finance*, 50:53–85, 1995.

[16] Peter E. Kloeden and Eckhard Platen. *Numerical Solution of Stochastic Differential Equations.* Springer, 1992.

[17] Lucy Xingwen Li. Pricing convertible bonds using partial differential equations. *M.Sc. Thesis, Computer Science Dept., Univ. of Toronto,* `http://www.cs.toronto.edu/pub/reports/na/lucy-05-msc.pdf`, 2005.

[18] J. McConell and E. Schwartz. Lyon taming. *Journal of Finance*, 50:53–85, 1986.

[19] R. Merton. On the pricing of corporate debt: The risk structure of interest rates. *Journal of Finance*, 29(2):449–470, 1974.

[20] A. R. Mitchell and D. F. Griffiths. *The Finite Difference Method in Partial Differential Equations.* John Wiley & Sons, 1987.

[21] K. G. Nyborg. The use and pricing of convertible bonds. *Applied Mathematical Finance*, 3:167–190, 1996.

[22] D. M. Pooley, P. A. Forsyth, and K. R. Vetzal. Numerical convergence properties of option pricing PDEs with uncertain volatility. *IMA Journal of Numerical Analysis*, 23:241–267, 2003.

[23] Goldman Sachs. Valuing convertible bonds as derivatives. *Quantitative Strategies Research Notes, Goldman Sachs*, 1994.

[24] Domingo Tavella and Curt Randall. *Pricing Financial Instruments: the finite Difference Method*. John Wiley& Sons, Inc., 2000.

[25] Kostas Tsiveriotis and Chris Fernandes. Valuing convertible bonds with credit risk. *The Journal of Fixed Income*, 8(2):95–102, 1998.

[26] Paul Wilmott. *Derivatives, the theory and practice of financial engineering*. John Wiley & Sons, Inc., 1998.

[27] Paul Wilmott, Sam Howison, and Jeff Dewynne. *The Mathematics of Financial Derivatives, A Student Introduction*. Cambridge University Press, 1995.

[28] Ali Bora Yigitbasioglu. Pricing convertible bonds with interest rate, equity, credit and FX risk. *ISMA Center, the University of Reading, Discussion paper 2001-14*, `http://www.ismacentre.rdg.ac.uk`, 2001.