Valuation of Contingent Convertibles with Derivatives

Nicolay Schmitt

Frankfurt School of Finance and Management

A thesis submitted for the degree of

Master of Science in Finance

August 2016

This thesis is dedicated to my parents for their love and support.
Thank you!

Abstract

Financial crises have led to higher regulatory standards on the capital adequacy of banks. Banks are required to hold more capital with loss absorbance capacities on their balance sheet. In conjunction with this development, contingent convertible bonds (CoCos) have become an attractive instrument for banks to seek new capital. The defining characteristic of CoCos is the automatic conversion into common equity when a predetermined trigger is met. Loss absorbing capital is created, which instantly improves the capital structure of the distressed bank. However, the pricing of these hybrid instruments remains opaque. In this context, the thesis scrutinizes the valuation of CoCos. Three dominant approaches are examined: the structural approach in accordance to Pennacchi (2010), the credit derivatives approach and the equity derivatives approach both pursuant to De Spiegeleer and Schoutens (2011). The application covers sensitivity analysis to further understand the dynamics of the different methodologies. Based on a case study the viability of those approaches is evaluated. Software is provided for further replication.

Contents

List of Figures

Chapter 1

Introduction and Motivation

1.1 Introduction

1.2 Literature Overview

1.3 Motivation

1.4 Methodology

Chapter 2

Structure of CoCos

2.1 Description of CoCos

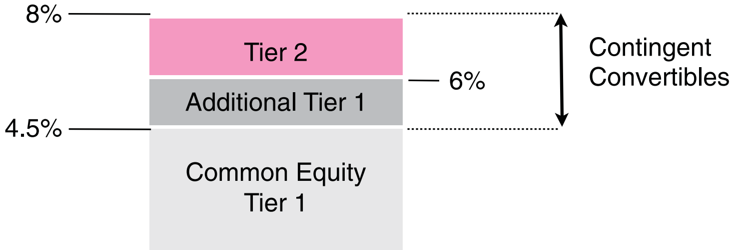

Figure 2.1: CoCos under Basel III (De Spiegeleer et al., 2014)

2.2 Payoff and Risk Profile

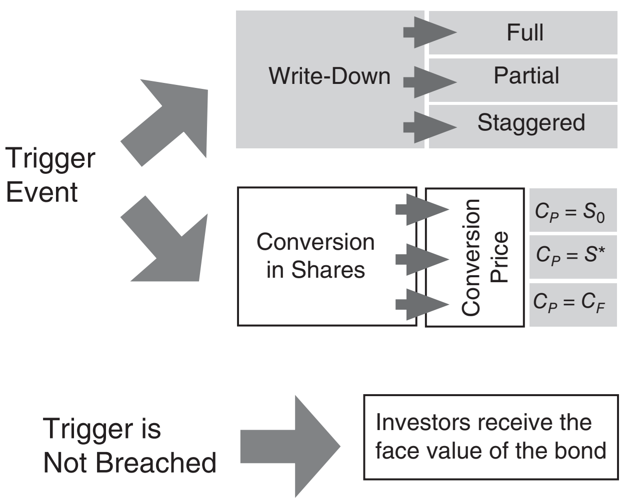

Figure 2.2: Anatomy of CoCos (De Spiegeleer et al., 2014)

2.3 Conversion Trigger

2.3.1 Market Trigger

2.3.2 Accounting Trigger

2.3.3 Regulatory Trigger

2.3.4 Multivariate Trigger

2.4 Conversion Details

2.4.1 Conversion Fraction

- conversion fraction α
- face value N
- conversion amount $N \times \alpha$
- amount remaining in case of partial equity conversion $N \times (1 - \alpha)$

2.4.2 Conversion Price and Ratio

- conversion rate C_r
- conversion price C_p
- recovery rate R_{CoCo}
- stock price at trigger event S_T^*
- loss attributable to CoCo holders L_{CoCo}

$$C_p = \frac{\alpha N}{C_r} \tag{2.1}$$

$$C_r = \frac{\alpha N}{C_p} \tag{2.2}$$

$$R_{CoCo} = \frac{S_T^*}{C_p} \tag{2.3}$$

$$L_{CoCo} = N - (1 - R_{CoCo}) = N\left(1 - \frac{S_T^*}{C_p}\right) \tag{2.4}$$

$$\tag{2.5}$$

$$P_T = \begin{cases} (1-\alpha)N + \frac{\alpha N}{C_p} S_T^* & \text{if converted} \\ N & \text{if not converted} \end{cases} \tag{2.6}$$

Chapter 3

Theory of Pricing

3.1 Credit Derivative Approach

In financial markets the reduced-form approach is widely used in order to price credit risk. It was originally introduced by Jarrow and Turnbull (1995) respectively Duffie and Singleton (1999). The credit derivative approach applies the reduced-form approach to CoCos. In this context, the derivation of a pricing formula for CoCos follows mainly De Spiegeleer and Schoutens (2011).

3.1.1 Reduced-Form Approach and Credit Triangle

The reduced-form approach is an elegant way of bridging the gap between the prediction of default and the pricing of default risk of a straight bond. In the following, we investigate the link between estimated default intensities and credit spreads under the reduced-form approach. (Lando, 2009)

Let τ denote the random time of default of some company. It is assumed that the distribution of τ has a continuous density function f, so that the distribution function F and the curve of survival probabilities q are related as follows:

$$P(\tau \leq t) = F(t) = 1 - q(t) = \int_0^t f(s)ds, \text{ with } t \geq 0 \tag{3.1}$$

The hazard rate respectively the default intensity λ is defined as follows:

$$\lambda(t) = \lim_{\Delta \downarrow 0} \frac{1}{\Delta} P(\tau \leq t + \Delta | \tau > t) = \frac{F'(t)}{q(t)} = \frac{F'(t)}{1 - F(t)} = -\frac{d}{dt} \log q(t) \tag{3.2}$$

Intuitively, the hazard rate is the default rate per year as of today. Using 3.2 we can derive a formula for the survival probability:

$$q(t) = \exp\left(-\int_0^t \lambda(s)ds\right) \tag{3.3}$$

For our application of the reduced-form approach we assume that the hazard rate $\lambda(t)$ is a deterministic function of time. In reality $\lambda(t)$ is not deterministic but itself stochastic. That fits in with the fact that credit spreads are not static but stochastically varying over time. (Schmidt, 2015) However, we further consider the hazard rate to be constant in order to simplify the problem. Hence, a constant hazard rate $\lambda(t) = \lambda$ implies an exponential distribution of the default time:

$$F(t) = 1 - q(t) = 1 - \exp(-\lambda t) \tag{3.4}$$

Under the idealized assumption of a flat zero interest rate curve, a flat spread curve and continuous spread payments, the default intensity λ can be calculated directly from the credit spread s and the recovery rate R by the rule of thumb formula (Schmidt, 2015), which is also known as credit triangle:

$$\lambda = \frac{s}{1 - R} \tag{3.5}$$

Finally, this relationship makes it possible to determine the default probability F from the credit spread s and vice versa.

3.1.2 Adaption to CoCos

In accordance with the aforementioned reduced-form approach, De Spiegeleer and Schoutens (2011) assume that the probability F^*, which measures the likelihood that a CoCo triggers within the next $T - t$ years, follows similar mechanics as the default probability of a straight bond does. Under the credit derivative approach the probability F^* can be expressed as follows:

$$F^* = 1 - \exp\left[-\lambda_{Trigger}(T - t)\right] \tag{3.6}$$

Additionally, the credit derivative approach models F^* with the first exit time equation used in barrier option pricing under a Black-Scholes setting. (Su and Rieger, 2009) Hence, the probability F^* that the trigger level S^* is touched within the next $T - t$ years is given by the following equation with the continuous dividend yield q, the continuous interest rate r, the drift μ, the volatility σ and the current share price S of the issuing company:

$$F^* = \Phi\left(\frac{\log\left(\frac{S^*}{S}\right) - \mu(T-t)}{\sigma\sqrt{(T-t)}}\right) + \left(\frac{S^*}{S}\right)^{\frac{2\mu}{\sigma^2}} \Phi\left(\frac{\log\left(\frac{S^*}{S}\right) + \mu(T-t)}{\sigma\sqrt{(T-t)}}\right) \tag{3.7}$$

In this regard, a CoCo's credit spread s_{CoCo} can be approximated by the credit triangle, where R_{CoCo} denotes the recovery rate of a CoCo and L_{CoCo} is the loss rate:

$$s_{CoCo} = (1 - R_{CoCo})\,\lambda_{Trigger} = L_{CoCo}\lambda_{Trigger} \tag{3.8}$$

In the trigger event, the face value N converts into C_r shares worth S^*. The loss of a long position in a CoCo is therefore determined by the conversion price C_p:

$$Loss_{CoCo} = N - C_r S^* = N\,(1 - R_{CoCo}) = N\left(1 - \frac{S^*}{C_p}\right) \tag{3.9}$$

By combining 3.6, 3.8 and 3.9 we see that the credit spread s_{CoCo} of a CoCo with maturity T at time t is driven by its major design elements, the trigger level S^* and the conversion price C_p:

$$s_{CoCo_t} = -\frac{\log(1 - F^*)}{(T - t)}\left(1 - \frac{S^*}{C_p}\right) \tag{3.10}$$

Subsequently, a pricing formula for CoCos under the credit derivative approach can be derived. The present value V^{cd} at time t is given by:

$$V_t^{cd} = \sum_{i=1}^{T} c_i \exp\left[-(r + s_{CoCo_t})(t_i - t)\right] + N \exp\left[-(r + s_{CoCo_t})(T - t)\right] \tag{3.11}$$

In summary, the credit derivative approach provides us with a concise method to price CoCos. However, one has to bear in mind its largest shortcoming. Losses from cancelled coupons of triggered CoCos are not taken into account in the valuation. Hence, the credit derivative approach naturally overestimates the price of CoCos, but it equips investors with a simple rule of thumb formula.

3.1.3 Data Requirements and Calibration

3.1.4 Valuation Example

3.2 Equity Derivative Approach

In order to assess the value of a CoCo, investors can use a method which depends on equity derivatives. (De Spiegeleer and Schoutens, 2011; De Spiegeleer et al., 2014) The so-called equity derivative approach attempts to compensate for the main drawback of the credit derivative approach, since it takes into account that coupon payments might be cancelled if the trigger of a CoCo was touched.

Subsequently, the valuation can be divided into two steps: In the first step, the value of a CoCo is determined without coupon payments. Such a CoCo is called Zero-Coupon CoCo. In the second step, one has to incorporate the coupon payments in the pricing formula while keeping in mind that they might be knocked out. The closed-form solution is derived under a Black-Scholes setting.

3.2.1 Step One - Zero-Coupon CoCo

To determine the present value of a Zero-Coupon CoCo V^{zcoco} at maturity T we can use equation 2.6. The underlying assumption of the equity derivative approach is that the triggering of a CoCo respectively of a Zero-Coupon CoCo is equivalent to the share price falling below the level S^*.

$$\begin{aligned} V_T^{zcoco} &= \begin{cases} N & \text{if not triggered} \\ (1-\alpha)N + \frac{\alpha N}{C_p}S^* & \text{if triggered} \end{cases} \\ &= N\mathbb{1}_{\{\tau > T\}} + \left[(1-\alpha)\,N + \frac{\alpha N}{C_p}S^*\right]\mathbb{1}_{\{\tau \leq T\}} \\ &= N + \left[\frac{\alpha N}{C_p}S^* - \alpha N\right]\mathbb{1}_{\{\tau \leq T\}} \\ &= N + \left[C_r S^* - \alpha N\right]\mathbb{1}_{\{\tau \leq T\}} \\ &= N + C_r\left[S^* - \frac{\alpha N}{C_r}\right]\mathbb{1}_{\{\tau \leq T\}} \\ &= N + C_r\left[S^* - C_p\right]\mathbb{1}_{\{\tau \leq T\}} \end{aligned} \tag{3.12}$$

It may be inferred that the financial payoff of equation 3.12 consists of two components (Erismann, 2015): (1) the face value N of a zero bond and (2) a long position in C_r shares generating a payoff only if the CoCo materializes at time τ. This component can be approximated with a knock-in forward. The intuition behind equation 3.12 is that if the share price falls below a certain level S^*, an investor will use the face value N to exercise the knock-in forward. That said, the investor is committed to buy the amount of C_r shares for the price of C_p at maturity T.

Before maturity the present value of a Zero-Coupon CoCo V^{zcoco} can be determined by adding up the present value of a zero bond V^{zb} and the present value of a knock-in forward V_t^{kifwd}. Hereinafter, the components will be explained briefly.

$$V_t^{zcoco} = V_t^{zb} + V_t^{kifwd} \tag{3.13}$$

with

$$V_t^{zb} = N \exp\left[-r(T-t)\right] \tag{3.14}$$

Moreover, the long position in shares at time t can be approximated with the respective closed-form solution of a knock-in forward. (Hull, 2006)

$$\begin{aligned} V_t^{kifwd} = C_r \Bigg[& S_t \exp\left[-q\left(T-t\right)\right] \left(\frac{S^*}{S_t}\right)^{2\lambda} \Phi\left(y_1\right) \\ & - K \exp\left[-r\left(T-t\right)\right] \left(\frac{S^*}{S_t}\right)^{2\lambda-2} \Phi\left(y_1 - \sigma\sqrt{T-t}\right) \\ & - K \exp\left[-r\left(T-t\right)\right] \Phi\left(-x_1 - \sigma\sqrt{T-t}\right) \\ & + S_t \exp\left[-q\left(T-t\right)\right] \Phi\left(-x_1\right) \Bigg] \end{aligned} \tag{3.15}$$

with

$$\begin{aligned} C_r &= \frac{\alpha N}{C_p} \\ K &= C_p \\ \lambda &= \frac{r - q + \frac{\sigma^2}{2}}{\sigma^2} \\ x_1 &= \frac{\log\left(\frac{S_t}{S^*}\right)}{\sigma\sqrt{T-t}} + \lambda\sigma\sqrt{T-t} \\ y_1 &= \frac{\log\left(\frac{S^*}{S_t}\right)}{\sigma\sqrt{T-t}} + \lambda\sigma\sqrt{T-t} \end{aligned}$$

It is important, however, to recognize that a subtle difference exists between the actual economic payoff of equation 3.12 and its replication with a knock-in forward, since the knock-in forward replicates an economic ownership of shares at maturity T. Though, the triggering of a CoCo forces investors to accept the conversion immediately. This could lead to an economic ownership of shares at trigger time τ and, thus, prior to T. Therefore, one could argue that receiving a knock-in forward in the trigger event disregards the dividends which a shareholder would receive in particular when a CoCo triggers early. De Spiegeleer and Schoutens (2011) argue that dividends can be neglected because distressed banks are likely to behave with great restraint when it comes to dividend payments.

3.2.2 Step Two - Adding Coupons

As mentioned earlier, the first step excludes coupon payments from the valuation. Yet, in this step we want to include them in order to replicate the exact payout profile of a CoCo. Therefore, we replace the zero bond in equation 3.13 with a straight bond with regular coupon payments c. Besides, a third component has to be added which takes into account the foregone coupon payments if the trigger is touched. This can be modeled with a short position in k binary down-and-in calls with maturity t_i. Those binary down-and-in calls are knocked in if the trigger S^* is met and hence, offset all future coupon payments.

The price of a straight bond can be determined with:

$$V_t^{sb} = \sum_{i=1}^{T} c_i \exp\left[-r(t_i - t)\right] + N \exp\left[-r(T - t)\right] \tag{3.16}$$

To price the down-and-in calls one might use the formula of Rubinstein and Reiner (1991):

$$\begin{aligned} V_t^{bdic} = \alpha \sum_{i=1}^{k} c_i \exp\left(-rt_i\right) \Bigg[&\Phi\left(-x_{1i} + \sigma\sqrt{t_i}\right) \\ &+ \left(\frac{S^*}{S_t}\right)^{2\lambda-2} \Phi\left(y_{1i} - \sigma\sqrt{t_i}\right) \Bigg] \end{aligned} \tag{3.17}$$

with

$$\begin{aligned} x_{1i} &= \frac{\log\left(\dfrac{S_t}{S^*}\right)}{\sigma\sqrt{t_i}} + \lambda\sigma\sqrt{t_i} \\ y_{1i} &= \frac{\log\left(\dfrac{S^*}{S_t}\right)}{\sigma\sqrt{t_i}} + \lambda\sigma\sqrt{t_i} \\ \lambda &= \frac{r - q + \dfrac{\sigma^2}{2}}{\sigma^2} \end{aligned}$$

To sum up, the theoretical price of a CoCo V^{ed} at time t pursuant the equity derivative approach consists of three components: (1) a straight bond V^{sb}, (2) a knock-in-forward V^{kifwd} and (3) a set of binary down-and-in calls V^{bdic}:

$$V_t^{ed} = V_t^{sb} + V_t^{kifwd} - V_t^{bdic} \tag{3.18}$$

3.2.3 Data Requirements and Calibration

3.2.4 Pricing Example

3.3 Structural Approach

- following section suggests a structural model as third alternative to price CoCos
- concept of structural models has its roots in the seminal work of Merton (1974)
- the goal of the Merton model is to explain the default of a company based on its assets and liabilities under a Black-Scholes setting
- default event happens at maturity date of debt when the assets of the issuer are worth less than the face value of the debt (Duffie and Singleton, 2003)
- approach we use to price CoCos was developed by Pennacchi (2010)
- method has a bank's balance sheet as the main driver of a CoCo's price
- moreover, the approach modifies the idea of a structural model to incorporate equity, short-term deposits, subordinated debt and contingent capital
- it tries to model the stochastic evolution of a bank's balance sheet respectively of its components, with default when the institution is unable to meet its obligations (Duffie and Singleton, 2003)

3.3.1 Synthetic Balance Sheet

3.3.1.1 Bank Assets

- bank's assets are invested into a portfolio of loans, securities and off-balance sheet positions whose returns follow a mixed jump-diffusion process
- bank's asset value at time t is denoted A_t
- change in quantity of bank assets equals the assets' return plus changes due to cash inflows less cash outflows
- sources of inflows and outflows from bank assets to be specified shortly; but for now superscript $*$ is used to distinguish asset changes solely due to their rate of return, not including changes due to net cashflows

- instantaneous rate of return that the bank earns on its assets is denoted as dA_t^*/A_t^*
- under the risk-neutral probability measure, $\mathbb{Q}$, this rate of return follows the process

$$\frac{dA_t^*}{A_t^*} = (r_t - \lambda_t k_t)\, dt + \sigma dz + \left(Y_{q_{t-}} - 1\right) dq_t \tag{3.19}$$

- note that dz is a Brownian motion under the risk-neutral probability measure
- q_t is Poisson counting process that increases by one when a Poisson-distributed event occurs
- dq_t is either zero when no Poisson event occurs or it augments by one whenever a jump occurs
- risk-neutral probability that a jump occurs and that q_t increases by one is $\lambda_t dt$ where λ_t is the intensity of the jump process
- $Y_{q_{t-}}$ is an identically and independently distributed random variable drawn from $\ln(Y_{q_{t-}}) \sim \Phi\left(\mu_y, \sigma_y^2\right)$ at time t where μ_y is the mean jump size and σ_y the standard deviation of the jumps
- depending on whether the random variable $Y_{q_{t-}}$ is either greater or smaller than one there is an upward or downward jump in the bank's asset value
- risk-neutral expected proportional jump is defined as $E_t^{\mathbb{Q}}\left(Y_{q_{t-}} - 1\right) = k_t$ where $k_t = \exp(\mu_y + \frac{1}{2}\sigma_y^2) - 1$ and the jump intensity and the risk neutral jump probability are assumed to be independent, then the change in the return over the time interval dt caused by the jump element $(Y_{q_{t-}} - 1)dq_t$ is $\lambda_t k_t dt$
- path of A_t^* as described in equation 3.19 will be continuous most of the time, but can have finite jumps of differing signs and amplitudes at discrete points in time where the timing of jumps depends on the Poisson random variable q_t and the jump sizes depend on the random variable $Y_{q_{t-}}$
- Jumps may be interpreted as times when important information affecting the value of the assets is released (Duffie and Lando, 2001)

- risk-neutral process followed by the bank's assets equals the assets' risk-neutral rate of return less the payout of interest and premiums to depositors and, as long as contingent capital is unconverted, coupons to contingent capital investors c_t

$$dA_t = \left[\left(r_t - \lambda k\right) A_t - \left(r_t + h_t\right) D_t - c_t B\right] dt + \sigma A_t dz + \left(Y_{q_{t-}} - 1\right) A_t dq \tag{3.20}$$

- asset process of equation 3.20 can be rewritten as

$$\begin{aligned} \frac{dA_t}{A_t} &= \left[\left(r_t - \lambda k\right) - \left(r_t + h_t\right) \frac{D_t}{A_t} - c_t b_t \frac{D_t}{A_t}\right] dt + \sigma dz + \left(Y_{q_{t-}} - 1\right) dq_t \\ &= \left[\left(r_t - \lambda k\right) - \frac{r_t + h_t + c_t b_t}{x_t}\right] dt + \sigma dz + \left(Y_{q_{t-}} - 1\right) dq_t \end{aligned} \tag{3.21}$$

- making the change in variable $x_t = A_t / D_t$ and recalling the deposit growth process $g\left(x_t - \hat{x}\right)$ of equation 3.27, the risk neutral process for the asset to-deposit ratio is

$$\begin{aligned} \frac{dx_t}{x_t} &= \frac{dA_t}{A_t} - \frac{dD_t}{D_t} \\ &= \left[\left(r_t - \lambda k\right) - \frac{r_t + h_t + c_t b_t}{x_t} - g\left(x_t - \hat{x}\right)\right] dt + \sigma dz + \left(Y_{q_{t-}} - 1\right) dq_t \end{aligned} \tag{3.22}$$

- simple application of Itô's lemma for jump-diffusion process implies

$$\begin{aligned} d \ln\left(x_t\right) = &\left[\left(r_t - \lambda k\right) - \frac{r_t + h_t + c_t b_t}{x_t} - g\left(x_t - \hat{x}\right) - \frac{1}{2}\sigma^2\right] dt \\ &+ \sigma dz + \ln Y_{q_{t-}} dq_t \end{aligned} \tag{3.23}$$

3.3.1.2 Default-Free Term Structure

3.3.1.3 Deposits

- given the risk-neutral distribution of asset returns, it is possible to solve for the fair deposit insurance premium or deposit credit risk premium h_t as function of the current asset to deposit ratio x_t
- date t quantity of deposits is denoted D_t
- since the bank is assumed to be closed by the deposit insurer whenever $x_t \leq 1$, if x_t reaches 1 following continuous movement of the bank assets, the bank is closed with $A_{t_b} = D_t$ and depositors suffer no loss

- depositors experience losses only following downward jump in asset value that exceeds the bank's capital
- if such a jump does occur at date $\hat{t}$, the instantaneous proportional loss to deposits is $\left(D_t - Y_{q_{t-}} A_{\hat{t-}}\right) / D_t$
- credit risk premium on the instantaneous-maturity deposits, h_t, which depends on the asset-to-deposit ratio, equals:

$$h_t = \lambda \left[\Phi\left(-d_1\right) - x_{t^-} \exp\left(\mu_y + \frac{1}{2}\sigma_y^2\right) \Phi\left(-d_2\right) \right] \tag{3.24}$$

with

$$d_1 = \frac{\ln\left(x_{t^-}\right) + \mu_y}{\sigma_y} \tag{3.25}$$

$$d_2 = d_1 + \sigma_y \tag{3.26}$$

- bank pays interest to the insured depositors at the competitive, instantaneous-maturity default-free rate, r_t
- other deposits may be uninsured and are paid the competitive, default-free interest rate, r_t, plus the fair credit risk premium, h_t
- in either the case of insured or uninsured deposits, the bank is assumed to continuously pay out total interest and deposit premiums of $\left(r_t + h_t\right) D_t dt$
- with interest and insurance premiums paid out continuously, the bank's total quantity of deposits changes only due to growth in net new deposits (deposit inflows or outflows), which are no directly related to accrual of interest and premiums
- because empirical evidence such as Adrian and Shin (2010) finds that banks have target capital ratios and deposit growth expands when banks have excess capital, the model assumes that deposit growth is positively related to the bank's current asset-to deposit ratio, defined as $x_t = A_t / D_t$

$$\frac{dD_t}{D_t} = g\left(\hat{x} - x_t\right) dt \tag{3.27}$$

- g is a positive constant
- $\hat{x} > 1$ is a target asset-to-deposit ratio
- when the actual asset-to-deposit ratio exceeds its target, $x_t > \hat{x}$, the bank issues positive amounts of net new deposits
- when $x_t < \hat{x}$ the bank is gradually shrinking its balance sheet
- thus, deposit growth rate per unit time, $g\,(x_t - \hat{x})$, creates a mean-reverting tendency for the bank's asset-to-deposit ratio, x_t

3.3.1.4 Contingent Convertible Capital

- biggest challenge is the accurate estimation of the valuation parameters that drive the stochastic processes(De Spiegeleer et al., 2014) because input factors are often unobservable variables

3.3.2 Data Requirements and Calibration

3.3.3 Pricing Example

Chapter 4

Dynamics and Sensitivity Analysis

4.1 Credit Derivative Approach

4.2 Equity Derivative Approach

4.3 Structural Approach

Chapter 5

Empirical Analysis and Model Comparison

5.1 Data Description

5.1.1 Deutsche Bank

5.2 Model Parametrization

5.3 Model Comparison

5.3.1 Qualitative Analysis

5.3.2 Quantitative Analysis

Chapter 6

Conclusion

Appendix A

Sample Title

Appendix B

Code - Credit Derivative Approach

The following source code is an implementation of the Credit Derivative Approach (De Spiegeleer and Schoutens, 2011) written in R.

```
# Price of Contingent Convertible Bond
price_coco_cd <- function(t, T, S_t, S_star, C_p, c_i, r, N, q, sigma){

  spread_coco <- calc_spread_coco(t, T, S_t, S_star, C_p, r, q, sigma)
  V_t_coco <- N * exp(-(r + spread_coco) * (T - t))

  for (t in 1:T){
    V_t_coco <- V_t_coco + c_i * exp(-(r + spread_coco) * t)
  }
  return(V_t_coco)
}

# Calculation of Trigger Probability
calc_p_star <- function(t, T, S_t, S_star, r, q, sigma){
  p_star <- pnorm((log(S_star / S_t) - calc_mu(r, q, sigma) * (T - t)) /
    (sigma * sqrt(T - t))) + (S_star / S_t)^(2 * calc_mu(r, q, sigma) /
    sigma^2) * pnorm((log(S_star / S_t) + calc_mu(r, q, sigma) * (T - t
   )) / (sigma * sqrt(T - t)))
  return(p_star)
}

# Calculation of Drift of Underlying
calc_mu <- function(r, q, sigma){
  mu <- r - q - sigma^2 / 2
  return(mu)
}

# Spread of CoCo Bond
calc_spread_coco <- function(t, T, S_t, S_star, C_p, r, q, sigma){
  spread_coco <- - log(1 - calc_p_star(t, T, S_t, S_star, r, q, sigma))
    / (T - t) * (1 - S_star / C_p)
  return(spread_coco)
}

# Pricing Example
```

```
price_coco_cd(t <- 0, T <- 5, S_t <- 40, S_star <- 20, C_p <- 25, c_i <-
    7, r <- 0.03, N <- 100, q <- 0, sigma <- 0.3)
```

Appendix C

Code - Equity Derivative Approach

The following source code is an implementation of the Equity Derivative Approach (De Spiegeleer and Schoutens, 2011) written in R.

```
# Price of Contingent Convertible Bond
price_coco_ed <- function(t, T, S_t, S_star, C_p, c_i, r, N, q, sigma,
    alpha){
  V_t_ed <- price_cb(t, T, c_i, r, N) - price_dibi(t, T, S_t, S_star, c_
    i, r, q, sigma, alpha) + price_difwd(t, T, S_t, S_star, C_p, r, N, q
    , sigma, alpha)

  return(V_t_ed)
}

# Price of Corporate Bond
price_cb <- function(t, T, c_i, r, N){
  V_t_cb <- N * exp(-r * (T - t))

  for (t in 1:T){
  V_t_cb <- V_t_cb + c_i * exp(-r * t)
  }

  return(V_t_cb)
}

# Price of Binary Option
price_dibi <- function(t, T, S_t, S_star, c_i, r, q, sigma, alpha){
  V_t_dibi <- 0

  i <- t
  k <- T

  for (i in 1:k) {
  V_t_dibi <- V_t_dibi + c_i * exp(- r * i) * (pnorm(- calc_x_1_i(S_t, S
    _star, sigma, r, q, i) + sigma * sqrt(i)) + (S_star / S_t)^(2 * calc
    _lambda(r, q, sigma) - 2) * pnorm ( calc_y_1_i(S_t, S_star, sigma, r
    , q, i) - sigma * sqrt(i)))
  }

```

```
  V_t_dibi <- alpha * V_t_dibi

  return(V_t_dibi)
}

# Price of Down-And-In Forward
price_difwd <- function(t, T, S_t, S_star, C_p, r, N, q, sigma, alpha){
  V_t_difwd <- calc_conversion_rate(C_p, N, alpha) * (S_t * exp(- q * (T - t)) * (S_star / S_t) ^ (2 * calc_lambda(r, q, sigma)) * pnorm(calc_y_1(t, T, S_t, S_star, r, q, sigma)) - C_p * exp(- r * (T - t)) * (S_star / S_t)^(2 * calc_lambda(r, q, sigma) - 2) * pnorm(calc_y_1(t, T, S_t, S_star, r, q, sigma) - sigma * sqrt(T - t)) - C_p * exp(- r * (T - t)) * pnorm(- calc_x_1(t, T, S_t, S_star, r, q, sigma) + sigma * sqrt(T - t)) + S_t * exp(- q * (T - t)) * pnorm(- calc_x_1(t, T, S_t, S_star, r, q, sigma)))

  return(V_t_difwd)
}

# Calculation of Conversion Rate
calc_conversion_rate <- function(C_p, N, alpha){
  C_r <- alpha * N / C_p

  return(C_r)
}

# Calculation of additional Parameters
calc_x_1_i <- function(S_t, S_star, sigma, r, q, t_i){
  x_1_i <- log(S_t / S_star) / (sigma * sqrt(t_i)) + calc_lambda(r, q, sigma) * sigma * sqrt(t_i)

  return(x_1_i)
}

calc_y_1_i <- function(S_t, S_star, sigma, r, q, t_i){
  y_1_i <- log(S_star / S_t) / (sigma * sqrt(t_i)) + calc_lambda(r, q, sigma) * sigma * sqrt(t_i)

  return(y_1_i)
}

calc_lambda <- function(r, q, sigma){
  lambda <- (r - q + sigma^2 / 2) / sigma^2

  return(lambda)
}

calc_x_1 <- function(t, T, S_t, S_star, r, q, sigma){
  x_1 <- log(S_t / S_star) / (sigma * sqrt(T - t)) + calc_lambda(r, q, sigma) * sigma * sqrt(T - t)

  return(x_1)
}

```

```
calc_y_1 <- function(t, T, S_t, S_star, r, q, sigma){
  y_1 <- log(S_star / S_t) / (sigma * sqrt(T - t)) + calc_lambda(r, q,
    sigma) * sigma * sqrt(T - t)

  return(y_1)
}

# Pricing Example
price_coco_ed(t <- 0, T <- 5, S_t <- 40, S_star <- 20, C_p <- 25, c_i <-
    7, r <- 0.03, N <- 100, q <- 0, sigma <- 0.3, alpha <- 1)
```

Appendix D

Code - Structural Approach

The following source code is an implementation of the Structural Approach (Pennacchi, 2010) with support from Quantnet (2011) in translating the source code of Pennacchi (2010) from GAUSS to R.

```
# Price of Contingent Convertible Bond
price_coco_sa <- function(T , npath , rho , kappa , r_bar, r0, sigma_r,
    mu_Y, sigma_Y, lambda, g, x_hat, b0, p, e_bar, sigma_x, x0_low, x0_
    high, x0_nint, B, c_low, c_high, c_nint){
  n <- T * 250
  dt <- T / n

  result <- sim_corrProcess(T, npath, rho, n, dt)
  dW_1 <- result$dW_1
  dW_2corr <- result$dW_2corr

  r <- sim_interestrate(kappa, r_bar, r0, sigma_r, dW_2corr, n, npath,
    dt)

  V_t_sa <- get_price(npath, n, dt, dW_1, dW_2corr, r, mu_Y, sigma_Y,
    lambda, g, x_hat, b0, p, e_bar, sigma_x, x0_low, x0_high, x0_nint, B
    , c_low, c_high, c_nint) * 100
  return(V_t_sa)
}

sim_corrProcess <- function(T, npath, rho, n, dt){
  vect <- c(1, rho, rho, 1)
  RHO <- matrix(vect, nrow = 2)
  chol_RHO <- t(chol(RHO))

  # Create two Brownian Motions
  dW_1 <- matrix(1, n, npath)
  dW_2 <- matrix(1, n, npath)

  for(j in 1:npath)
  {
    dW_1[ , j] <- rnorm(n) * sqrt(dt)
    dW_2[ , j] <- rnorm(n) * sqrt(dt)
  }
```

```

  # Create Correlated Process based on Brownian Motions using Cholesky-Decomposition
  dW_2corr <- matrix(1, n, npath)
  for(j in 1:npath)
  {
    for(i in 1:n)
    {
      dW_2corr[i, j] <- dW_1[i, j] * chol_RHO[2, 1] + dW_2[i, j] * chol_RHO[2, 2]
    }
  }

  return(list("dW_1" = dW_1, "dW_2corr" = dW_2corr))
}

# Create Interest Rate Process
sim_interestrate <- function(kappa, r_bar, r0, sigma_r, dW_2corr, n, npath, dt){
  r <- matrix(r0, n + 1, npath)

  for(j in 1:npath)
  {
    for(i in 1:n)
    {
      r[i + 1, j] <- r[i, j] + kappa * (r_bar - r[i, j]) * dt + sigma_r * sqrt(r[i, j]) * dW_2corr[i, j]
    }
  }

  return(r)
}

get_price <- function(npath, n, dt, dW_1, dW_2corr, r, mu_Y, sigma_Y, lambda, g, x_hat, b0, p, e_bar, sigma_x, x0_low, x0_high, x0_nint, B, c_low, c_high, c_nint){

  c_fit_matrix <- matrix(0, x0_nint, length(lambda))

  for(w in 1:length(lambda))
  {
    # Create Parametres for Jump Process
    phi <- matrix(rbinom( n%*%npath, 1, dt * lambda[w]), n, npath)
    ln_Y <- matrix(rnorm(n%*%npath, mu_Y, sigma_Y), n, npath)

    b <- matrix(b0, n + 1, npath)
    x_bar0 <- 1 + e_bar + p * b0
    x_bar <- matrix(x_bar0, n + 1, npath)

    h <- matrix(1, n, npath)

    k <- exp(mu_Y + 0.5 * sigma_Y^2) - 1

    c <- seq(c_low, c_high, length = c_nint)
```

```
  x0 <- seq(x0_low, x0_high, length = x0_nint)

  for(l in 1:x0_nint)
  {
    for(m in 1:c_nint)
    {
      x <- matrix(x0[l],n+1,npath)
      ln_x0 <- matrix(log(x0[l]),n+1,npath)
      ln_x <- ln_x0
      binom_c <- matrix(1,n+1,npath)

      for(j in 1:npath)
      {
        for(i in 1:n)
        {
          d_1 <- (ln_x[i, j] + mu_Y) / sigma_Y
          d_2 <- d_1 + sigma_Y

          h[i, j] <- lambda[w] * (pnorm( - d_1) - exp(ln_x[i, j]) * exp(mu_Y + 0.5 * sigma_Y^2) * pnorm(-d_2))

          b[i + 1, j] <- b[i, j] * exp(- g[w] * (exp(ln_x[i, j]) - x_hat) * dt)

          ln_x[i + 1, j] <- ln_x[i, j] + ( (r[i, j] - lambda[w] * k) - (r[i, j] + h[i, j] + c[m] * b[i, j]) / exp(ln_x[i, j]) - g[w] * (exp(ln_x[i, j]) - x_hat) - 0.5 * sigma_x^2) * dt + sigma_x * sqrt(dt) * dW_1[i, j] + ln_Y[i,j] * phi[i, j]

          x[i + 1, j] <- exp(ln_x[i + 1, j])

          x_bar[i + 1, j] <- 1 + e_bar + p * b[i + 1, j]

          if(x[i + 1, j] >= x_bar[i + 1, j] && binom_c[i, j] > 0.5)
          {
            binom_c[i + 1, j] <- 1
          }else
          {
            binom_c[i + 1, j] <- 0
          }
        }
      }

      payments <- matrix(c(rep(c[m] * dt, n - 1), B), n, npath) * binom_c[1:n, ]

      for(j in 1:npath){
        for(i in 2:n){
          if(payments[i, j] == 0 && p * b[sum(binom_c[ , j]) + 1, j] <= x[sum(binom_c[ , j]) + 1, j] - 1 ){
            payments[i, j] <- p * B
            break
          }
```

```
            else if(payments[i, j] == 0 && 0 < x[sum(binom_c[ , j]) + 1, j] - 1 && x[sum(binom_c[ , j]) + 1, j] - 1 < p * b[sum(binom_c[ , j]) + 1, j]){
              payments[i, j] <- (x[sum(binom_c[ , j]) + 1, j] - 1) * B / b[sum(binom_c[ , j]) + 1, j]
              break
            }
            else{
              payments[i, j] <- payments[i, j]
            }
          }
        }
        vec_disc_v <- rep(0, npath)
        for(j in 1:npath)
        {
          disc_v <- 0
          int_r <- 0

          for(i in 1:n)
          {
            int_r <- int_r + r[i, j] * dt
            disc_v <- disc_v + exp(- int_r) * payments[i, j]
          }
          vec_disc_v[j] <- disc_v
        }

        V_t_sa <- mean(vec_disc_v)

        return(V_t_sa)
      }
    }
  }
}

# Pricing Example
price_coco_sa(T = 5, npath = 2, rho = - 0.2, kappa = 0.114, r_bar = 0.069, r0 = 0.035, sigma_r = 0.07, mu_Y = -0.01, sigma_Y = 0.02, lambda = c(1), g = c(0.5), x_hat = 1.1, b0 = 0.04, p = 1, e_bar = 0.02, sigma_x = 0.02, x0_low = 1.15, x0_high = 1.15, x0_nint = 10, B = 1, c_low = 0.05, c_high = 0.05, c_nint = 10)
```

References

Tobias Adrian and Hyun Song Shin. Liquidity and leverage. *Journal of Financial Intermediation*, 19(3):418–437, 2010.

Jan De Spiegeleer and Wim Schoutens. Pricing contingent convertibles: a derivatives approach. *Available at SSRN 1795092*, 2011.

Jan De Spiegeleer, Wim Schoutens, and Cynthia Van Hulle. *The Handbook of Hybrid Securities: convertible bonds, coco bonds and bail-in.* John Wiley & Sons, 2014.

Darrell Duffie and David Lando. Term structures of credit spreads with incomplete accounting information. *Econometrica*, 69(3):633–664, 2001.

Darrell Duffie and Kenneth J Singleton. Modeling term structures of defaultable bonds. *Review of Financial studies*, 12(4):687–720, 1999.

Darrell Duffie and Kenneth J Singleton. *Credit risk pricing, measurement, and management.* Princeton University Press, 2003.

Marc Erismann. *Pricing Contingent Convertible Capital-A Theoretical and Empirical Analysis of Selected Pricing Models.* PhD thesis, University of St. Gallen, 2015.

John C Hull. *Options, futures, and other derivatives.* Pearson Education, 2006.

Robert A Jarrow and Stuart M Turnbull. Pricing derivatives on financial securities subject to credit risk. *The Journal of Finance*, 50(1):53–85, 1995.

David Lando. *Credit risk modeling: theory and applications.* Princeton University Press, 2009.

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

George Pennacchi. A structural model of contingent bank capital. Working Paper 1004, Federal Reserve Bank of Cleveland, 2010. URL `https://ideas.repec.org/p/fip/fedcwp/1004.html`.

Quantnet. Translate gauss code to r code, 2011. URL `https://www.quantnet.com/threads/translate-gauss-code-to-r-code.7784/`.

Mark Rubinstein and Eric Reiner. Unscrambling the binary code. *Risk Magazine*, 4 (9):20, 1991.

Wolfgang Schmidt. Credit risk, default models and credit derivatives. Frankfurt School of Finance and Management, 2015.

Lujing Su and Marc Olivier Rieger. How likely is it to hit a barrier? theoretical and emperical estimates. Technical report, Technical Report Working Paper, 2009.