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. They 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

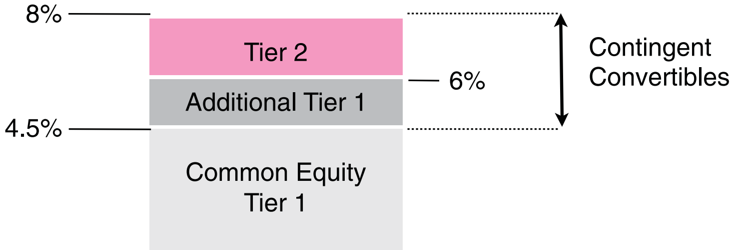

Figure 2.1: CoCos under Basel III (De Spiegeleer et al., 2014)

2.2 Payoff and Risk Profile

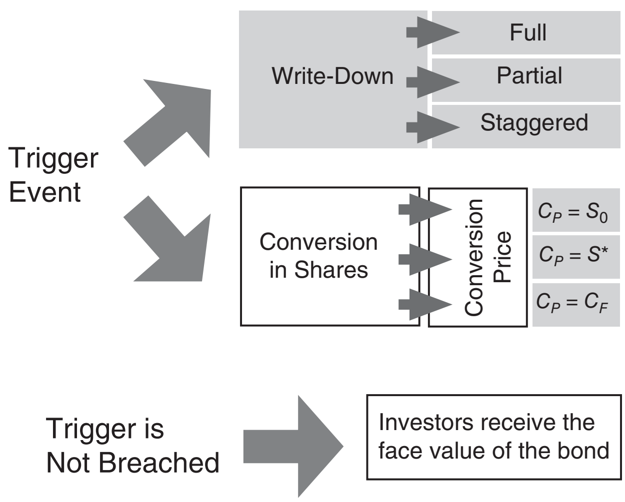

Figure 2.2: Anatomy of CoCos (De Spiegeleer et al., 2014)

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

The reduced-form approaches is widely used in financial markets in order to price credit risk. It was originally introduced by Jarrow and Turnbull (1995) respectively Duffie and Singleton (1999). In this context, the credit derivative approach applies the reduced-form approach to CoCos. With that said, the derivation of a pricing formular for CoCos follows mainly Lando (2009) and 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) But 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 context, 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 can be calculated 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

- another approach pursuant to De Spiegeleer and Schoutens (2011) and De Spiegeleer et al. (2014) uses equity derivatives to assess the theoretical value of CoCos

- equity derivative approach attempts to compensate for the apparent disadvantage of the credit derivative approach

  it takes into account that coupons may be knocked out if trigger is met

- pricing can be divided into two steps

  first step: values a CoCo without coupon payments; a so called Zero-Coupon CoCo

  second step: incorporates coupon payments in the pricing formula

- the closed-form formula of the equity derivative approach involves the use of standard Black-Scholes assumptions

3.2.1 First Step - Zero-Coupon CoCo

- underlying assumption behind this approach is, that the triggering of a Zero-Coupon CoCo is equivalent to the share price falling below the level S^*

- trigger indicator $\mathbb{1}_{\{\tau \leq T\}}$ equals one when the Zero-Coupon CoCo triggers before maturity T at default time τ and otherwise the indicator function is zero

- value of Zero-Coupon CoCo V^{zcoco} at maturity T, can be derived based on 2.6 as shown by Erismann (2015)

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

- financial payoff of equation 3.12 can be broken down into two components

  face value N of a Zero Bond

  long position in C_r shares generating a payoff only if CoCo materializes

- long position in shares can be approximated with a knock-in forward
- Hence, at time t the payoff profile V_t^{zcoco} can be replicated with a zero-coupon bond V_t^{zb} and a knock-in forward V_t^{kifwd}

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

- price of zero bond at V^{zb} time t can be calculated with the risk free rate r:

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

- long position in shares at time t can be approximated with a knock-in forward with price V^{kifwd}

  long position in a knock-in call

  short position in a knock-in put on the underlying shares

  both with strike K equal to conversion price C_p

  with barrier level equal to trigger price S^*

- intuition: if trigger is met at share price S^*, investor uses face value N to exercise forward which commits to buy the amount of C_r shares for the price of C_p at maturity T
- closed form solution exists for both knock-in options (Merton, 1973)
- price of knock-in call V^{kic} and knock-in put V^{kip} at time t can be calculated with:

$$\begin{aligned} V_t^{kic} &= S_t \exp\left[-q\left(T-t\right)\right]\left(\frac{S^*}{S_t}\right)^{2\lambda}\Phi\left(y\right) \\ &\quad - K\exp\left[-r\left(T-t\right)\right]\left(\frac{S^*}{S_t}\right)^{2\lambda-2}\Phi\left(y-\sigma\sqrt{T-t}\right) \end{aligned} \tag{3.15}$$

with

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

$$
\begin{aligned}
V_t^{kip} = {} & S_t \exp\left[-q\left(T-t\right)\right]\left(\frac{S^*}{S_t}\right)^{2\lambda}\left[\Phi\left(y\right) - \Phi\left(y_1\right)\right] \\
& - K \exp\left[-r\left(T-t\right)\right]\left(\frac{S^*}{S_t}\right)^{2\lambda-2}\left[\Phi\left(y - \sigma\sqrt{T-t}\right) - \Phi\left(y_1 - \sigma\sqrt{T}\right)\right] \\
& + K \exp\left[-r\left(T-t\right)\right]\Phi\left(x_1 + \sigma\sqrt{T-t}\right) \\
& - S_t \exp\left[-q\left(T-t\right)\right]\Phi\left(-x_1\right) \qquad (3.16)
\end{aligned}
$$

with

$$
\begin{aligned}
x_1 &= \frac{\log\left(\frac{S_t}{S^*}\right)}{\sigma\sqrt{T-t}} + \lambda\sigma\sqrt{T-t} \\
y_1 &= \frac{\log\left(\frac{S^*}{S_t}\right)}{\sigma\sqrt{T-t}} + \lambda\sigma\sqrt{T-t}
\end{aligned}
$$

- knock-in forward can be constructed with knock-in call and knock-in put (Hull, 2006)
- hence, price of knock-in forward V^{kifwd} at time t can be replicated using equation 3.15 and 3.16:

$$
\begin{aligned}
V_t^{difwd} = C_r \Bigg[& S_t \exp\left[-q\left(T-t\right)\right]\left(\frac{S^*}{S_t}\right)^{2\lambda}\Phi\left(y_1\right) \\
& - K \exp\left[-r\left(T-t\right)\right]\left(\frac{S^*}{S_t}\right)^{2\lambda-2}\Phi\left(y_1 - \sigma\sqrt{T-t}\right) \\
& - K \exp\left[-r\left(T-t\right)\right]\Phi\left(-x_1 - \sigma\sqrt{T-t}\right) \\
& + S_t \exp\left[-q\left(T-t\right)\right]\Phi\left(-x_1\right)\Bigg] \qquad (3.17)
\end{aligned}
$$

with

$$C_r = \frac{\alpha N}{C_p} \tag{3.18}$$

- constraint: subtle difference between actual economic payoff of 3.12 and replication with a knock-in forward

  knock-in forward replicates an economic ownership of shares at maturity T

  But: triggering of CoCo forces investor to accept conversion immediately leading to an economic ownership of shares at τ

- it may be argued: receiving a forward when the trigger is met disregards the dividends a shareholder would receive especially when a CoCo triggers early in its lifetime

- 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 Second Step - Adding Coupons

- As mentioned in the first step we did not include coupon payments in our valuation

- In this step we replace the zero bond with a straight bond with regular coupon payments c_i and a present value of V_t^{sb}

$$V_t^{sb} = \sum_{i=1}^{T} c_i \exp\left[-r(t_i - t)\right] + N \exp\left[-r(T - t)\right] \tag{3.19}$$

- However, we have to add a third component which takes into account the foregone coupon payments if the trigger is met

  investor receives coupon payments only if trigger is not met

  value of CoCo lower than straight bond of same issuer

  difference can be modeled with a short position in k binary down-and-in calls short position in a single binary down-and-in call with maturity t_i for each coupon payment at t_i

reduce present value of coupon payments of a straight bond

binary down-and-in option are knocked in if trigger S^* is met offsetting future coupon payments

short position model possibility of losing coupon payment

$$\begin{aligned} V_t^{bdic} = \alpha \sum_{i=1}^{k} c_i \exp\left(-r t_i\right) \Bigg[& \Phi\left(-x_{1i} + \sigma\sqrt{t_i}\right) \\ & + \left(\frac{S^*}{S_t}\right)^{2\lambda-2} \Phi\left(y_{1i} - \sigma\sqrt{t_i}\right) \Bigg] \end{aligned} \tag{3.20}$$

with

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

- theoretical price of CoCo V^{ed} pursuant the equity derivative approach equals (1) straight bond plus (2) knock-in-forward and (3) binary down-and-in calls

$$V_t^{ed} = V_t^{sb} + V_t^{kifwd} - V_{t_i}^{bdic} \tag{3.21}$$

3.2.3 Data Requirements and Calibration

3.2.4 Pricing Example

3.3 Structural Approach

"Structural credit pricing models are based on modeling the stochastic evolution of the balance sheet of the issuer, with default when the issuer is unable or unwilling to meet its obligations." (Duffie and Singleton, 2003)

3.3.1 Synthetic Balance Sheet

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.

```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)
  }
```

```r

  # 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

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 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. Theory of rational option pricing. *The Bell Journal of economics and management science*, pages 141–183, 1973.

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/`.

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.