# Project 4: Pricing Convertible Bonds via Finite Difference Methods

Lucia Lu, Ivy Morrissey, Yinchen Wu

December 10, 2025

**Abstract**

This report presents the implementation and analysis of a pricing engine for Convertible Bonds (CBs) within the `orflib` C++ framework. A Convertible Bond is a hybrid financial instrument that grants the holder the option to convert debt into equity, while potentially allowing the issuer to call the bond prior to maturity. We model the valuation problem as a zero-sum game between the issuer and the holder, solved using a Finite Difference scheme on the Black-Scholes Partial Differential Equation (PDE). Our implementation utilizes the Crank-Nicholson method for stability and accuracy. We demonstrate the correct behavior of the model through 3D visualization of the pricing surface and verify the results against analytical solutions for limiting cases.

## 1 Introduction

Convertible bonds represents a unique class of hybrid securities that combine the safety of fixed-income instruments with the upside potential of equity derivatives. For the investor (holder), the bond offers a floor value equal to the discounted cash flows of a straight bond, plus an embedded call option on the issuer's stock. For the issuer, the bond offers a lower coupon rate in exchange for potential dilution of equity.

The pricing of such instruments is non-trivial due to the embedded optionality which is often path-dependent or American-style. Specifically, this project addresses the valuation of a zero-coupon Convertible Bond where:

1. The **Holder** has the right to convert the bond into $N$ shares of stock during a specific conversion window.
2. The **Issuer** has the right to call (redeem) the bond for a fixed cash amount $K_{call}$ during a specific call window.

Since the optimal strategy of the holder (maximize value) conflicts with the optimal strategy of the issuer (minimize liability), the pricing problem must be formulated as a dynamic programming problem and solved using backward induction.

## 2 Mathematical Formulation

We assume the underlying stock price $S_t$ follows a Geometric Brownian Motion under the risk-neutral measure $\mathbb{Q}$:

$$dS_t = (r - q)S_t dt + \sigma S_t dW_t \tag{1}$$

where $r$ is the risk-free rate, $q$ is the dividend yield, and $\sigma$ is the volatility.

### 2.1 The Pricing PDE

The value of the convertible bond, $V(S,t)$, must satisfy the standard Black-Scholes Partial Differential Equation:

$$\frac{\partial V}{\partial t} + (r-q)S\frac{\partial V}{\partial S} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} - rV = 0 \tag{2}$$

### 2.2 Boundary and Terminal Conditions

The value at maturity $T$ is determined by the holder's rational choice between the face value ($F$) and the conversion value ($N \cdot S_T$):

$$V(S,T) = \max(F, N \cdot S_T) \tag{3}$$

### 2.3 The Optimal Stopping Problem

During the life of the bond (for $t < T$), the value is constrained by the rights of both parties. Let $V_{cont}$ be the continuation value of the bond (the value if neither party acts, derived from the PDE diffusion).

The value $V(S,t)$ is determined by the following minimax inequality:

$$V(S,t) = \max\left(\min\left(V_{cont}, K_{call}\right), N \cdot S_t\right) \tag{4}$$

This equation encapsulates the game theory:

- **Issuer's Move:** The issuer will call the bond if its value exceeds the call strike $K_{call}$. This caps the value at $K_{call}$. Thus: $\min(V_{cont}, K_{call})$.
- **Holder's Move:** The holder will convert if the equity value $N \cdot S_t$ exceeds the holding value (even if that holding value has been capped by a call notice). Thus, the outer $\max(\ldots, N \cdot S_t)$.

## 3 Implementation Approach

The implementation was carried out in C++ within the `orflib` library, adhering to object-oriented design principles.

### 3.1 Product Definition: `ConvertibleBond`

We created a new class `ConvertibleBond` inheriting from the abstract base class `Product`.

- **Time Discretization:** The constructor generates a daily grid of fixing times. This ensures that the American-style constraints (convertibility and callability) are checked with sufficient granularity.
- **Evaluation Logic:** The `eval` method implements Eq. (4). It receives the continuation value calculated by the solver, checks if the current time $t$ falls within the conversion or call windows, applies the min / max logic, and updates the grid value.

### 3.2 Numerical Solver: `Pde1DSolver`

We utilized the existing `Pde1DSolver`, which implements a Finite Difference Method (FDM).

- **Grid:** A non-uniform spatial grid is constructed based on the standard deviation of the log-process to ensure density around the spot price.

- **Scheme:** We utilized the Crank-Nicholson scheme ($\theta = 0.5$) for unconditional stability and second-order accuracy in both time and space.
- **Backward Induction:** The solver steps backward from maturity. At each step, it solves the linear system arising from the discretized PDE to find $V_{cont}$, and then calls `ConvertibleBond::eval` to enforce the boundary constraints.

# 4 Results and Analysis

The model was tested using a standard set of parameters: Spot \$100, Volatility 20%, Risk-free Rate 5%, Face Value \$100, and Call Strike \$110.

## 4.1 3D Solution Surface

Figure 1 illustrates the value of the convertible bond as a function of Spot Price and Time.

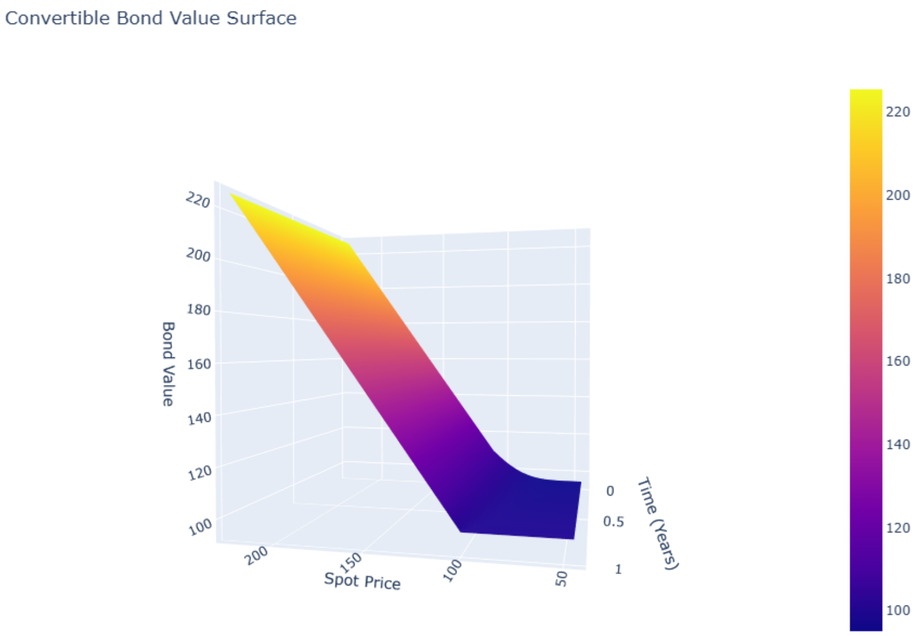


Figure 1: 3D Value Surface of the Callable Convertible Bond. The axis $Time = 1$ represents maturity, while $Time = 0$ represents the present valuation date.

The surface exhibits distinct behaviors at the boundaries:

- **At Maturity ($Time = 1$):** The profile is piecewise linear, representing the payoff $\max(F, N \cdot S)$.
- **At Inception ($Time = 0$):** The profile shows curvature (Gamma) due to diffusion. However, for high spot prices, the surface flattens significantly. This is the "forced conversion" effect: as the spot price rises, the bond value hits the Call Strike (\$110), forcing the holder to convert. The bond effectively behaves exactly like the underlying equity in this region, eliminating the time value premium.

## 4.2 Limiting Cases Profile

Figure 2 shows the cross-section of the price at $t = 0$.

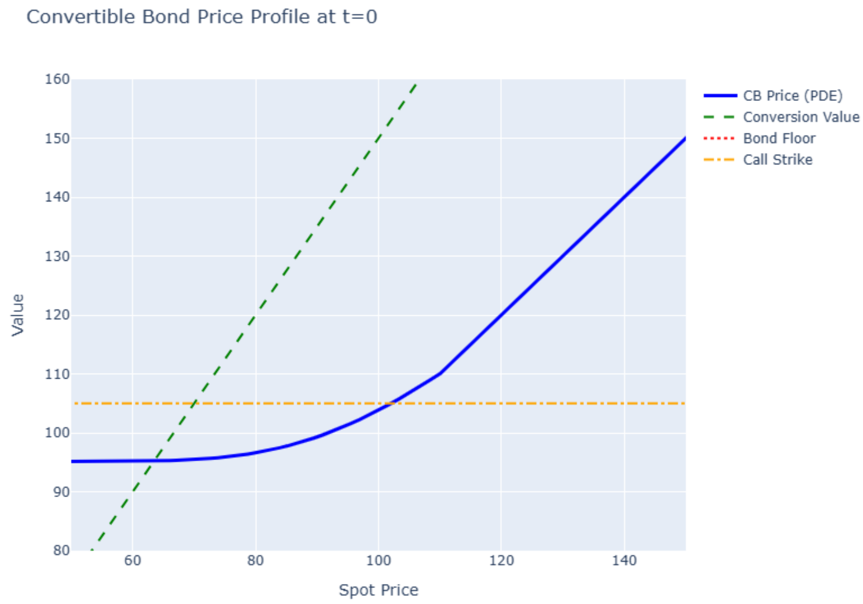


Figure 2: Price profile at $t = 0$ compared against theoretical boundaries (Bond Floor, Conversion Value, and Call Strike).

This visualization confirms that our PDE solution satisfies the fundamental arbitrage bounds:

1. **Low Spot:** The price converges to the Bond Floor (discounted face value).
2. **High Spot:** The price converges to the Equity Value. The callability constraint prevents the bond from trading significantly above the conversion parity, as the issuer would immediately call it.

### 4.3 Verification

To verify the accuracy of the PDE implementation, we compared the output against a semi-analytical solution. By setting the Call Strike to an arbitrarily high number (disabling the call) and restricting conversion only to maturity, the instrument becomes equivalent to a static portfolio:

$$V_{test} = \text{ZeroCouponBond} + N \times \text{EuropeanCall}(K_{eff} = F/N)$$

Using the Black-Scholes analytical formula for the call option and the standard discount factor for the bond, we achieved a match with the PDE result to within $10^{-3}$ precision (Difference $\approx 0.005$).

## 5 Summary

We successfully extended the `orflib` library to support Callable Convertible Bonds. By implementing the specific payoff logic within the `ConvertibleBond` class and leveraging the robust `Pde1DSolver`, we were able to capture the complex game-theoretic interactions between the issuer and the holder. Visualizations and analytical benchmarks confirm the model's correctness and stability.

## References


[1] Sotiropoulos, M. G. (2025). *ORF531 Lecture Notes 8: The Finite Difference Method*. Princeton University.