Assignment7_AY23_24
Group 14

Idda, Raffaeli, Riondato, Stillo

Contents

1 Certificate Pricing

In this initial phase, we aims to hedge the bank's position (as outlined in Annex 1) by entering into a swap contract. First of all, by calibration of assignment 5, we found the value of the parameters that we need (σ, η, k). Then we implemented the simulation of the forward using Monte Carlo method, considering the same F_0 of lab 5 ($F_0 = 2.97 \times 10^3$) :

1. Simulate g as a normal standard random variable, and G depending on the model considered:
2. Compute the log increment as: $f_t = \sqrt{\delta} \cdot \sigma \cdot G \cdot g - \left(\frac{1}{2} + \eta\right) \cdot \delta \cdot \sigma^2 \cdot G - \text{LaplaceExp}(\eta)$
3. Get S: $S(t) = F(t,t) = F(t-1,t-1) \cdot e^{f_t}$

Because we find probabilities with Monte Carlo, we did not calculate the confidence interval.

To compute the contract's value at t_0, we discounted the cash flows and determined the Net Present Value of the contract. Subsequently, by equating the NPV to zero, we derived the upfront value. However, the cash flows described in Annex 2 indicate that the presence of coupons is directly linked to the underlying asset's values two days prior to the payment dates. Hence, two scenarios are plausible, as follows:

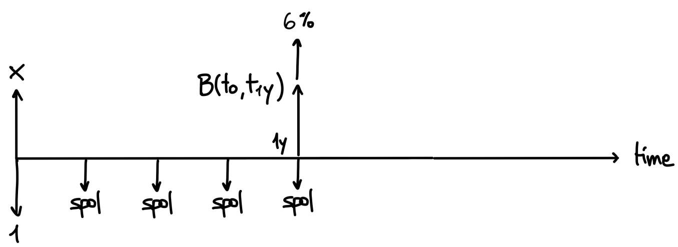


Figure 1: if $S_N \leq K$

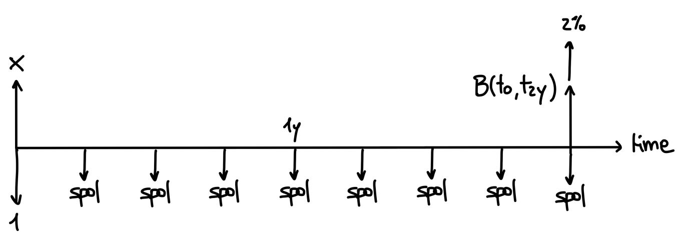


Figure 2: if $S_N >$ K

We also need to check if CumulativeCouponAccrual > TriggerLevel, but since the trigger level is the same as the coupon of the first year (6%), we just need to check each time if $S_N \leq$ K. Therefore, the formula for the upfront in case S_N is smaller or equal than the strike (N=1 year) is:

$$X = 1 - B(t_0, t_N) + \sum_{i=1}^{N} [s^{\text{spol}} \cdot \delta(t_{i-1}, t_i) B(t_0, t_i)] - 0.06 \cdot \delta(t_0, t_N) B(t_0, t_N) \tag{1}$$

On the other hand, if S_N is bigger (N=2 years), the upfront is:

$$X = 1 - B(t_0, t_N) + \sum_{i=1}^{N} [s^{\text{spol}} \cdot \delta(t_{i-1}, t_i) B(t_0, t_i)] - 0.02 \cdot \delta(t_0, t_N) B(t_0, t_N) \tag{2}$$

1.1 NIG model

To compute the probabilities of each scenario, we exploited the Normal Inverse Gaussian model running one million MonteCarlo simulations and computing G following this procedure:

- Generate independent $u \sim uniform(0,1)$ and $z \sim \chi_1^2$
- $\hat{G} = 1 - \frac{k}{2}\left(\sqrt{z^2 + \frac{4z}{k}} - z\right)$
- If $(1+\hat{G}) \cdot u > 1$ then $G = \frac{1}{\hat{G}}$, else $G = \hat{G}$

And the parameters are:

α	0.5
Σ	0.1012
η	13.6159
k	1.1540

Table 1: NIG parameters

Merging both (1) and (2), we obtained $\boldsymbol{X} = \mathbf{2.49\%}$.

1.2 VG model

Furthermore, in order to price the same issue we can exploit also the Variance Gamma method. Indeed, that is the same model of NIG but making the parameter alpha tend towards zero. To compute G, we used the built-in matlab function `gamrnd`. Calibrating with $\alpha = 0$ we obtain:

α	0
Σ	0.1385
η	4.6170
k	1.7705

Table 2: VG parameters

In this framework, merging both (1) and (2), we obtained $\boldsymbol{X} = \mathbf{2.49\%}$. The error from the upfront of NIG model is 0.06%, so it's a good choice of an alternative model.

1.3 Black model

However, it is possible to price this issue also exploiting the Black model; in this case we have only one free parameter (σ), that we can acquire with spline interpolation on the implied volatility curve using our strike.
Also in this case we simulated the forward using Monte Carlo technique. The difference from the previous cases lies in the log increment calculation:

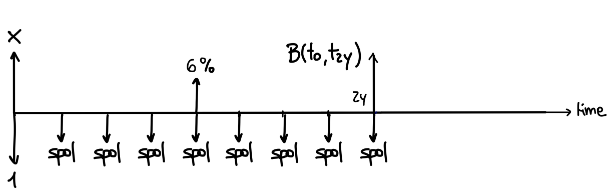
$$f_t = -0.5 \cdot \text{blk\_vol}^2 \cdot \text{ttm}(1) + \text{blk\_vol} \cdot \sqrt{\text{ttm}(1)} \cdot g$$

Always exploiting equations (1) and (2), the upfront is $\boldsymbol{X} = \mathbf{1.54\%}$.

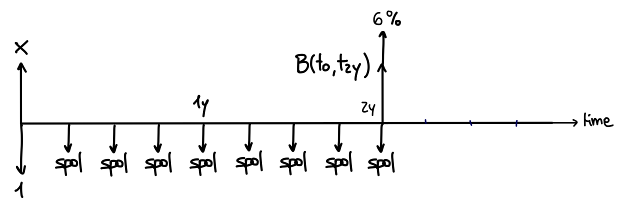
The error with respect to the upfront using NIG model is of 38.1% because the digital risk is not properly taken into account, so this model isn't the best alternative.

1.4 3 years maturity upfront

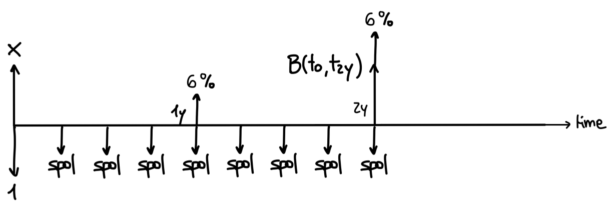
Since we use a Monte Carlo simulation, and not a closed formula, we can extend our analysis also to the structured bond with a three-year expiry using both NIG and VG model. With a maturity of 3 years we have four different scenarios:



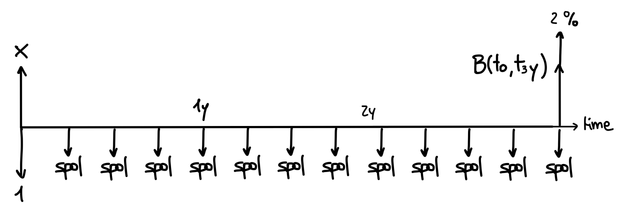
(a) case 1



(b) case 2



(c) case 3



(d) case 4

We calculated the upfront using the same procedure as described above, finding $\boldsymbol{X} = \mathbf{3.69\%}$ with both models. The error between models is 0.01%.

Note: in calculating the probability of all cases, it is important to remember that they are not independent.

2 Bermudian Swaption Pricing via Hull-White

The objective of this exercise was to evaluate the price of a Bermudan yearly Payer Swaption (Strike 5%, non-call 2) with an underlying swap maturing in 10 years (Start Date: February 15^{th}, 2008) employing a Trinomial tree. Subsequently, we were asked to verify the correctness of the tree's design and implementation.

2.1 Swaption pricing using Trinomial tree

In a Trinomial tree structure, every node branches into three paths, each linked to a probability indicating how the state at that node might evolve. In the context of the Hull-White framework, we initially employed the tree, after appropriately discretizing the time horizon, to simulate the Ornstein-Uhlenbeck process. In this context, since such a process is a "mean-reverting process," the tree is characterized by a maximum limit beyond which the process no longer grows/declines but remains constant.

Model tools:

$$\begin{cases} \Delta X = X_{i+1} - X_i \quad = -\hat{\mu} \cdot X_i + \hat{\sigma} \cdot g_i \\ X_0 = 0 \end{cases} \tag{3}$$

where:

$$\begin{cases} g_i \sim N(0,1) \\ \hat{\mu} = 1 - e^{-a*\Delta t} \\ \hat{\sigma} = \sigma * \sqrt{\dfrac{1 - e^{-2a*\Delta t}}{2a}} \end{cases} \tag{4}$$

and moreover:

$$\begin{cases} X_i = l * \Delta \\ \quad l = -l_{max} : 1 : l_{max} \\ l_{max} = ceil(1 - \sqrt{\frac{2}{3}} * \frac{1}{\hat{\mu}}) \\ \quad \Delta = \sqrt{3} * \hat{\sigma} \end{cases} \tag{5}$$

Below is a graphic representation of the tree structure:

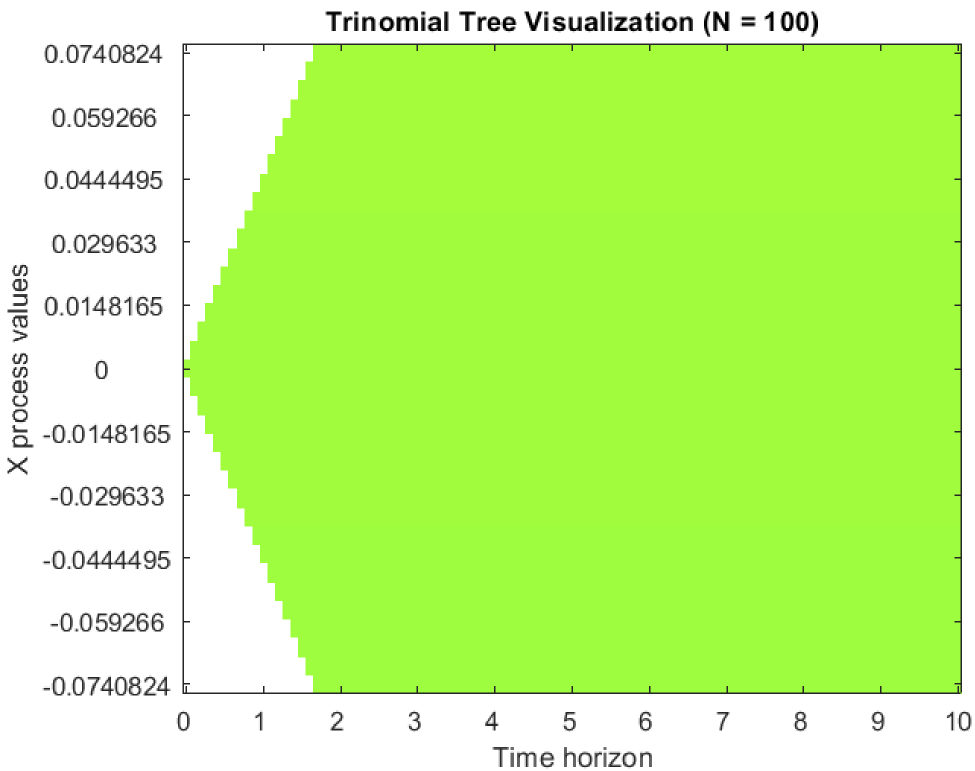


Figure 4: Tree

Note: The maximum value that the underlying process can reach is determined by how dense the temporal grid created earlier to discretize is, since there exists a dependence.

After obtaining the values of the OU process (thanks to the customized function `compute_forward_tree`), we applied the following formula (corollary of Lemma 1 (HJM)) to find the discount factor curves:

$$\begin{cases} \frac{B(s; T_\alpha, T_i)}{B(t_0; T_\alpha, T_i)} = e^{-X_s \frac{\sigma(s,T_i) - \sigma(s,T_\alpha}{\sigma} - \frac{1}{2} \int_{t_0}^{s} [\sigma(u,T_i)^2 - \sigma(u,T_\alpha)^2] \, du} \\ \qquad T_i = T_{\alpha+1}, ..., T_\omega \end{cases} \tag{6}$$

where

- t_0 is the February 19$^{\text{th}}$, 2008, i.e. the date on which to price the Swaption.

- t_α is the expiry of the Swaption and since we are considering a Bermudan Swaption it can vary between 2 years (in the first year the option is not callable) and 9 years (we cannot exercise the expiry in the last year since there would not be any swap).
- t_ω is the maturity of the underlying swap (fixed at 10 years).

Note: To compute $B(t_0; T_\alpha, T_i)$ we used the well-known "spot discounts - forward discounts relationship" by obtaining spot discounts on the desired dates with bootstrapping.

Since we needed to price a Bermudan Swaption, we found the option's value at t_0 by calculating the expected value of the discounted payoffs, making sure in the nodes where it is possible to early exercise to take the maximum between the payoff and the continuation value ($Price = E_0[D(t_0, T : \alpha) * Payoff]$). The payoffs depend on the discounts ($Payoff = BPV_{\alpha\omega}[S_{\alpha\omega}(T_\alpha) - K]^+$ with $S_{\alpha\omega}(T_\alpha) = \frac{1-B(T_\alpha,T_\omega)}{BPV_{\alpha\omega}[T_\alpha]}$), and since we discounted them later, our "**s**" in the forward discount formula above became T_α, with T_α varying depending on when the option is exercised. This way, the calculated forward curves effectively became spot curves.
The expected value was practically computed by exploring the tree backwards.
To calculate stochastic discounts with which to discount payoffs we used the approximated formula for backwards methods coming from Lemma 2 (HJM):

$$\begin{cases} D(t_i, t_{i+1}) = B(t_i, t_{i+1}) * e^{-\frac{1}{2}\hat{\sigma}^{*2} - \frac{\hat{\sigma}^*}{\hat{\sigma}}[\Delta X_{i+1} e^{-a*\Delta t} + \hat{\mu} X_{i+1}]} \\ \hat{\sigma}^* = \dfrac{\sigma}{a}\sqrt{\Delta t - \dfrac{2}{a}(1 - e^{-a*\Delta t}) + \dfrac{1}{2a}(1 - e^{-2a*\Delta t})} \end{cases} \tag{7}$$

After creating appropriate customized functions to simplify what is explained above, we then estimated the option price, obtaining the following result: **PRICE = 0.022 €**

Note: This value is already obtained for a very small number of time steps (e.g. N=20). Increasing N shows that the price actually converges to this value.

2.2 Checking the correct implementation of the tree

To test whether the tree algorithm we created works correctly, we thought we would use it to price an instrument for which we know a closed formula. The "Jamshidian Approach" offers a pseudo closed formula (□) to price European Swaptions due to the equivalence that exists between the price of a Plain Vanilla Swaption and a Coupon Bond Put option with appropriate parameters.

Note (□): The Jamshidian formula is considered an exact formula because it is a closed formula minus the calculation of X^*, which is computed numerically, but whose computational cost is very low.

CRITICALITIES: We applied Jamshidian's formula to price an EU Payer Swaption on a Plain Vanilla Swap (not Forward Start) as a Put Coupon Bond option with strike $k = 1$. We further decomposed the problem by considering the Coupon Bond option as the sum of Zero Coupon Bond Put options with the appropriate parameters.
To evaluate the prices of Zero Coupon Bond options, we used the closed formula of the HJM model.
The result we obtained is: **PRICE = 0.013 €**.
However, trying to replicate the same calculation using the tree we obtained a result of **PRICE = 0.048 €**.

In both cases we found a price of the EU Swaption lower than the price of the Bermudan Swaption, which is reasonable since the second instrument also offers the possibility of exercising the option in advance with respect to expiry and therefore costs more.

We tried our best to figure out what wasn't working but, unfortunately, we could not figure out whether we made a mistake in the type of tool with which to check the correctness of the tree, whether we got confused in our choice of parameters to apply the Jamshidian Approach, or whether we readjusted the tree constructed to price the Bermudan in the wrong way.