Politecnico di Milano

Financial Engineering

A.Y. 2022/2023

Assignment 8

Authors:

Roberto Carlo De Marco
Alice Iamoni
Rida Jafar
Filippo Lipari

1 Case study: certificate pricing

1.1 Point a)

Bank XX aims at hedging an issue by entering in a swap contract with an Investment Bank. In order to determine the upfront paid by the I.B. at time t_0, following the swap term sheet, we need to compute the NPV of the remaining flows of both parties and then subtracting the ones paid by the I.B. to the ones paid by the Bank.
In the two years case, Bank XX has to pay Euribor 3m plus a spread of 1.2% and receives the following coupons: 5% at $1Y$ and 2% at $2Y$. The payments of these coupons, however, are subject to some conditions. Namely, the one at the first year happens only if the underlying EURO STOXX is lower than the strike $K = 3000$. In addition to that we have Early Redemption Clause: before the payment of every coupon we check if the sum of past payments and the next payment is bigger than a trigger. If this is the case then the whole swap ends.
Having this in mind we simulated the EURO STOXX dynamics according to a normal mean variance mixture model with alpha equal to 0. The first step is to calibrate the model, we therefore obtained the following parameters:

σ	κ	η
0.1374	1.6922	4.9964

The second step is the implementation of the simulation:

1. We started by computing F0, obtained from S0 through this formula:
$$F0 = S0 \cdot e^{ttm*(r-d)} \tag{1}$$
2. We simulated the random variables needed: g standard normal and G gamma random variable
3. We computed the log increment with the following formula:
$$f_t = \sqrt{\delta} \cdot \sigma \cdot \sqrt{G} \cdot g - (1/2 + \eta) \cdot \delta \cdot \sigma^2 * G - laplaceexp(\eta) \tag{2}$$
4. We used the increment simulations to obtain a simulation for F at the reset dates of every year:
$$F(t,t) = F(t-1, t-1) \cdot e^{f_t} \tag{3}$$
5. In the end, we obtained S knowing that (from the no arbitrage relation):
$$S(t) = F(t,t) \tag{4}$$

At this point, for each realization of S we know which coupons will be paid, so we can discount back the flows of each case and then take the mean. In order to do this correctly we have to create two variables to do the checks: $St < Strike$, $CumulativeCouponAccrual > TriggerLevel$. In our case though, since the trigger level is the same as the coupon of the first year, the situation simplifies a lot - the I.B. in fact will only pay one of the flows - and we just need to check each time the first condition and it's complementary one.
The function described above returns an estimation for the upfront X obtained through an empirical mean, the value obtained is: **0.03349295615311**.
One last thing that is worth mentioning is that the upfront in this case could also be computed using the digital option formula (digital put for S at the first year for the first coupon and digital call for S at the first year for the second one). We preferred though to stick to the MC simulation since we also had to implement the case with maturity 3 years for point e, for which this simplification wasn't trivial.
Finally, since we used a Monte Carlo simulation, we also provided in the code a measure for the length of the CI of the upfront estimation, thanks to which we calibrated the number of MC simulations needed to obtain a length smaller than 1 bp, which in the end was 10^7. This leads to a not negligible computational time, but we deemed it to be an acceptable trade off, to change the value for a faster run it's enough to change the variable 'N' in the beginning of the code.

1.2 Point b)

It is possible to change the model that we used both for the calibration and for the simulation in order to obtain the upfront. To change the model we generalized the previously mentioned function, meaning that it could take different values of alphas. In this way the Laplace Exponent of the random variable G changes, so instead of simulating a gamma variable we have to simulate G as an Alpha-Stable random variable. In the following table and plot, we have reported the result we obtained for different values of alphas:

α	$UPFRONT$
0.1	0.03396527827041
0.2	0.03739730023273
0.3	0.03986468782633
0.4	0.04140646564210
0.5	0.04193881747968
0.6	0.04104122552583
0.7	0.03830489054908
0.8	0.03399995947244
0.9	0.02987808345191

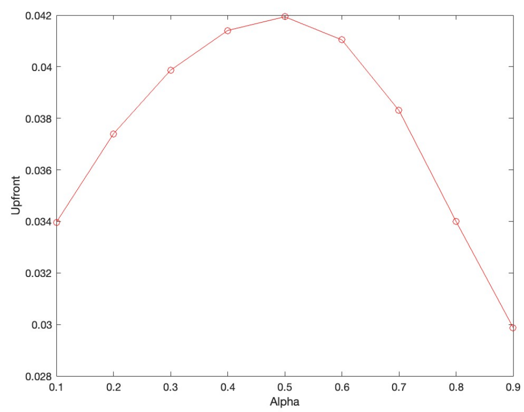

Figure 1: Upfront for different models

1.3 Point c)

We then built a new function for the Black model, which only has 1 parameter (σ), instead of the three we had before. The sigma we needed was the implied volatility, so it was obtained by interpolation on the implied volatility curve using our strike of interest. In this case we simulated directly the underlying using the Black Model:

$$S(t) = S(t-1)e^{(r-d-\frac{\sigma^2}{2})\delta(t-1,t)+\sqrt{\delta(t-1,t)}\sigma g} \tag{5}$$

where r is the forward rate between t-1 and t and g is Gaussian. The steps that follow are the same as the previous case. The result that we obtain with this procedure is: **0.02320632181310**. So the error with respect to the VG model is: **0.0102866343400**.

1.4 Point d)

Since we used a MC simulation in points a) and b), the alternative we proposed is still available for the three years expiry option. This is due to the fact that the function can simply be extended as we will describe in a better way in point e). If however we used, as mentioned before, the closed formula solution with the digital options, we couldn't have extended it to this case, since the option becomes path dependent and this solution can't be extended trivially anymore.

1.5 Point e)

We finally generalized again the function of point a) to work also for an option with maturity at 3 years. Namely, since we were already using a MC simulation, it was enough to extend it to another time step and check the conditions on the coupons for each year.
In particular, since the trigger level didn't change, the I.B. will only pay one of the three possible coupons: either 5% at 1 year, 5% at 2 years or 2.5% at 3 years.
By computing the NPVs and subtracting them as described before, we obtained in this final case an upfront of **0.04577932323023**.

2 Exercise: Swaption pricing via Hull-White

2.1 Price with Jamshidian formula

In order to price price a 3y7y and a 5y5y ATM swaptions (Strike equal to the corresponding Forward rate) within a single curve framework, and considering a 1 factor Hull-White model with $a = 12\%$ and $\sigma = 0.7\%$, we proceeded as follows.

First, starting from the formula of a Swaption payer we find that it's equivalent to a Put option on a Coupon bond with strike=1.

$$\begin{aligned} SP_{\alpha,\omega}(t_0) &= E_0\left[D(t_0,t_\alpha)\cdot BPV_{\alpha,\omega}(t_\alpha)\cdot[S_{\alpha,\omega}(t_\alpha)-K]^+\right] \\ &= E_0\left[D(t_0,t_\alpha)\cdot[1-P_{\alpha,\omega}(K,t_\alpha)]^+\right] \end{aligned} \tag{6}$$

where: $P_{\alpha,\omega}(K,t_\alpha)$ is coupon bond with coupons equal to K. Since the contract is ATM we have that:

$$K = S_{\alpha,\omega}(t_0) = \frac{1-B_{\alpha,\omega}(t_0)}{BPV_{\alpha,\omega}(t_0)} \tag{7}$$

We then proceeded to compute the strikes of the zero coupon bonds K_i:

$$K_i = P(x^*,t;T_\alpha,T_\omega) = \sum_{i=\alpha}^{\omega-1} c_i\cdot B_{\alpha,i+1}(x^*,t) \tag{8}$$

where:

$$B_{\alpha,i}(x^*,t) = B(t_0,T_\alpha,T_i)\cdot\exp\left(-x^*\cdot\frac{\sigma(t,T_i)-\sigma(t,T_\alpha)}{\sigma}-\frac{1}{2}\cdot\int_{t_0}^{t}[\sigma(u,T_i)^2-\sigma(u,T_\alpha)^2]du\right). \tag{9}$$

Lastly, thanks to the Jamshidian formula, we exploited the fact that the put on a coupon bond is the linear combination of puts on zero coupon bonds, according to the following formula:

$$Put_{\alpha,\omega}(t_0,K) = \sum_{i=\alpha}^{\omega-1} c_i\cdot Put^{ZC}_{\alpha,i+1}(t_0,K_i) \tag{10}$$

using this, we obtained the following results:

3y7y Swaption	5y5y Swaption
0.0165085766910	0.0148068907549

2.2 Price via tree

The tree approach consists in simulating the X dynamics (Ornstein Uhlenbeck process) over a grid of time steps. Then we exploit the relation between the process X and the forward discounts factor. At this point we can compute the payoff in each node of the tree thanks to the disocunt factors.

Let's analyze the implementation:

1. This is the zero-mean OU dynamic:

$$\begin{cases} dX_t = -aX_t + \sigma dW_t \\ X_{t_0} = 0 \end{cases} \tag{11}$$

   We know that this process has mean reverting property, and so in the creation of the grid we consider this characteristic using 2 symmetrical borders: l_{min}, l_{max}

2. We create the grid for the X using the following relations:

$$\begin{cases} \Delta x = \sqrt{3}\hat{\sigma} \\ x = l\Delta x \end{cases} \qquad \begin{cases} \hat{\sigma} = -\sigma\sqrt{\frac{1-\exp(-2adt)}{2a}} \\ \hat{\mu} = 1 - \exp(-adt) \\ l = -l_{max}, ..., -1, 0, 1, ..., +l_{max} \end{cases}$$

   We will have tree cases to face: *Case A*, *Case B*, *Case C* (explained by the pictures below).

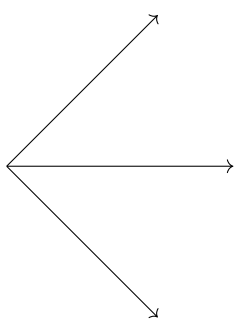

Figure 2: Case A

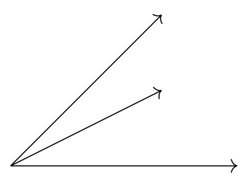

Figure 3: Case B

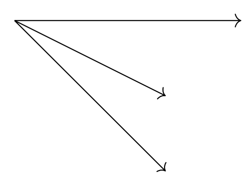

Figure 4: Case C

3. We had to compute the probabilities to arrive in each node, given the node at previous time step; this was done by solving linear systems:

$$\begin{cases} p_u = \frac{1}{2}(\frac{1}{3} - \hat{\mu}l + \hat{\mu}^2 l^2) \\ p_d = \frac{1}{2}(\frac{1}{3} + \hat{\mu}l + \hat{\mu}^2 l^2) \\ p_m = \frac{2}{3} - \hat{\mu}^2 l^2 \\ l = [-l_{max}+1, +l_{max}-1] \end{cases} \qquad \begin{cases} p_u = \frac{1}{2}(\frac{1}{3} + \hat{\mu}l + \hat{\mu}^2 l^2) \\ p_d = \frac{1}{2}(\frac{7}{3} + 3\hat{\mu}l + \hat{\mu}^2 l^2) \\ p_m = \frac{-1}{3} - \hat{\mu}^2 l^2 - 2l\hat{\mu} \\ (l = -l_{max}) \end{cases} \qquad \begin{cases} p_u = \frac{1}{2}(\frac{7}{3} - 3\hat{\mu}l + \hat{\mu}^2 l^2) \\ p_d = \frac{1}{2}(\frac{1}{3} - \hat{\mu}l + \hat{\mu}^2 l^2) \\ p_m = \frac{-1}{3} - \hat{\mu}^2 l^2 + 2l\hat{\mu} \\ (l = l_{max}) \end{cases}$$

4. We consider the payoff of the at-the-money (ATM) swaption, which can be treated as a put option (with strike $K = 1$) on a coupon bond with coupons equal to the strike of the swaption. Starting from the payoff at the final time step (the maturity of the option), we compute all possible payoffs (according to the probabilities of each step) by working backwards through the tree and discounting them with the stochastic discount given by (it is stochastic since it is dependent on the node we are discounting from):

$$D(t_i, t_{i+1}) \simeq B(t_i, t_{i+1}) \exp\left(-\frac{1}{2}\hat{\sigma^*}^2 - \hat{\sigma^*} g_i\right) = B(t_i, t_{i+1}) \exp\left(-\frac{1}{2}\hat{\sigma^*}^2 - \frac{\hat{\sigma^*}}{\hat{\sigma}}(\Delta x_{i+1} - \hat{\mu} x_i)\right) \tag{12}$$

These are the result we obtained with 4 steps per year (number of time instants: $M = 13$).

3y7y Swaption	5y5y Swaption
0.01617269448645	0.01476060303637

Below you can find the comparisons obtained using different methods and time grid density:

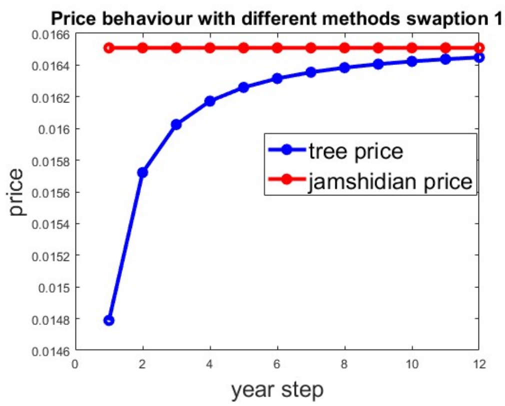

Figure 5: Swaption 1 price comparison

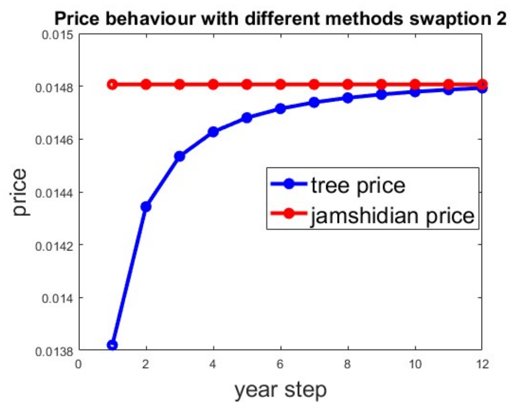

Figure 6: Swaption 2 price comparison

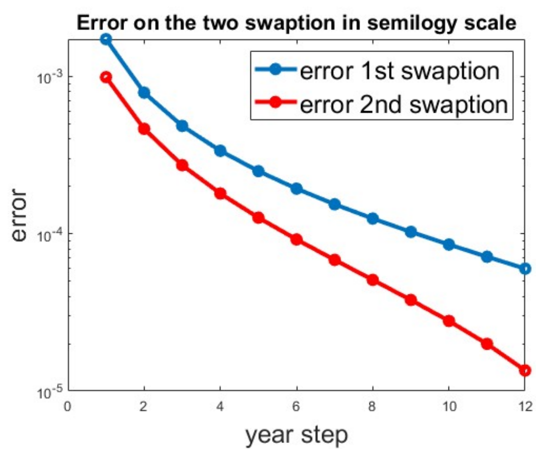

Figure 7: Error behaviour with respect to time grid density