Problems for the Quantathon: 1

The issue time for all options coincides with the initial time. The maturities, barrier, and exercise times are strictly greater than the initial time.

Forward Swap Lock

T : the maturity.

Parameters of the underlying swap:

N : the notional.

R : the fixed rate.

δt : the interval of time between the payments given as year fraction.

M : the total number of payments.

side : this parameter defines the side of the swap contract, i.e. whether one pays "fixed" and receives "float" or otherwise.

At maturity T a holder of the contract enters into the interest rate swap with the parameters defined above and the issue time T. Write the algorithm with only one event time t_0.

Algorithm. We need only one event time:

$$\{t_0\},$$

which is the initial time. We multiply on the notional at the end.

Assume first that we receive float LIBOR $L(s,t)$ and pay fixed rate R. Then

$$\underbrace{X(t_0)}_{\text{pay fixed}} = \text{bank account} - \text{coupon bond} = Y(t_0) - Z(t_0),$$

where

1. The bank account pays float interest

$$L(t_{m-1}, t_m)\delta t \quad \text{at} \quad t_m = T + m\delta t, \quad m = 1, \dots, M,$$

and notional 1 at maturity t_M. By replication, its value at T given by the notional:

$$Y(T) = 1.$$

The value at t_0 is given by the discount factor:

$$Y(t_0) = B(t_0, T).$$

2. The coupon bond pays fixed coupons

$$NR\delta t \quad \text{at} \quad t_m = T + m\delta t, \quad m = 1, \ldots, M,$$

and notional 1 at maturity t_M. Its value is given by

$$Z(t_0) = R\delta t \sum_{m=1}^{M} B(t_0, t_m) + B(t_0, t_M).$$

If we pay float, then

$$\underbrace{X(t_0)}_{\text{pay float}} = - \underbrace{X(t_0)}_{\text{pay fixed}} .$$

We return $NX(t_0)$.

Interest rate collar

N : the notional.

C : the cap rate.

F : the floor rate, $F < C$.

δt : the time interval between the payments given as year fraction.

M : the total number of payments.

t_0 : the initial time.

The payment times are given by

$$t_m = t_0 + m\delta t, \quad m = 1, \ldots, M.$$

At payment time t_m, we receive cap payment

$$N \max(L(t_{m-1}, t_m)\delta t - C\delta t, 0)$$

and make floor payment

$$N \max(F\delta t - L(t_{m-1}, t_m)\delta t, 0).$$

Here, $L(t_{m-1}, t_m)$ is the market interest rate computed at t_{m-1} for maturity t_m.

Algorithm. The event times are

$$\{t_0, (t_m)_{m=1,\dots,M-1}\},$$

where t_0 is the initial time and $(t_m)_{m=1,\dots,M-1}$ are all payment times except the last one.

We divide the algorithm into 3 steps. We multiply on the notional at the end (Step 3).

Step 1 (Boundary condition).

$$\begin{aligned}\underbrace{X(t_{M-1})}_{>t_{M-1}} = &\max(1 - B(t_{M-1}, t_M)(1 + C\delta t), 0) \\ &- \max(B(t_{M-1}, t_M)(1 + F\delta t) - 1, 0).\end{aligned}$$

Step 2 (Loop). We enter the loop at t_{M-1} (included) and exit at t_0 (not included):

$$\underbrace{t_0}_{\text{end}} \longleftarrow \underbrace{t_{M-1}}_{\text{begin}}.$$

We consider the iteration:

$$\underbrace{X(t_m)}_{?} \longleftarrow \underbrace{X(t_{m+1})}_{\text{known}},$$

where

$$X(t_{m+1}) = \underbrace{X(t_{m+1})}_{>t_{m+1}}$$

is the value to continue (the value of caplets paid after t_{m+1}). We have that

$$\underbrace{X(t_m)}_{>t_{m+1}} = \mathcal{R}_{t_m}\big(\underbrace{X(t_{m+1})}_{>t_{m+1}}\big)$$

and then that

$$\begin{aligned}\underbrace{X(t_m)}_{>t_m} = \underbrace{X(t_m)}_{>t_{m+1}} &+ B(t_m, t_{m+1}) \max(L(t_m, t_{m+1})\delta t - C\delta t, 0) \\ &- B(t_m, t_{m+1}) \max(F\delta t - L(t_m, t_{m+1})\delta t, 0) \\ = \underbrace{X(t_m)}_{>t_{m+1}} &+ \max(1 - B(t_m, t_{m+1})(1 + C\delta t), 0) \\ &- \max(B(t_m, t_{m+1})(1 + F\delta t) - 1, 0),\end{aligned}$$

where $B(s,t)$ is the discount factor at s for maturity t and we used the identity:

$$B(s,t)(1 + L(s,t)(t - s)) = 1.$$

Step 3 (After the loop). We return $NX(t_0) = N \underbrace{X(t_0)}_{>t_0}$.

American put on forward rate agreement

N : the notional amount.

R : the fixed rate.

δt : the time interval for the loan as year fraction.

$(t_m)_{m=1,\ldots,M}$: the exercise times.

The owner of the option has the right to sell the forward rate agreement at any exercise time t_m. In this case,

1. at time t_m he receives notional N;
2. at time $t_m + \delta t$ he pays notional plus fixed interest, that is, the amount $N(1 + R\delta t)$.

Algorithm. The event times are given by

$$\{t_0, (t_m)_{m=1,\ldots,M}\}.$$

We divide the algorithm into 3 steps. We multiply on the notional at the end (Step 3).

Step 1 (Boundary condition). We have that

$$\underbrace{X(t_M)}_{>t_M} = 0,$$

as there are no exercises after the maturity.

Step 2 (Loop). We enter the loop at t_M (included) and exit at t_0 (not included):

$$\underbrace{t_0}_{\text{end}} \longleftarrow \underbrace{t_M}_{\text{begin}}.$$

We consider the iteration:

$$\underbrace{X(t_m)}_{?} \longleftarrow \underbrace{X(t_{m+1})}_{\text{known}},$$

where

$$X(t_{m+1}) = \underbrace{X(t_{m+1})}_{>t_{m+1}}$$

is the value to continue (the value of the potential payments if we exercise after t_{m+1}). We have that

$$\underbrace{X(t_{m+1})}_{>t_m} = \max(\underbrace{X(t_{m+1})}_{>t_{m+1}}, 1 - B(t_{m+1}, t_{m+1} + \delta t)(1 + R\delta t)),$$

where $B(s,t)$ is the discount factor at s for maturity t. Finally,

$$\underbrace{X(t_m)}_{>t_m} = \mathcal{R}_{t_m}\big(\underbrace{X(t_{m+1})}_{>t_m}\big).$$

Step 3 (After the loop). We return $NX(t_0) = N \underbrace{X(t_0)}_{>t_0}$.

American swaption

$(t_m)_{m=1,\ldots,M}$: the exercise times.

Parameters of underlying swap

N : the notional.

R : the fixed rate.

δt : the interval of time between the payments.

M : the total number of payments.

side : the side of the swap contract, that is, whether one pays "fixed" and receives "float" or otherwise.

A holder of the option can enter into the underlying swap agreement at any exercise time t_m. This time then becomes the issue time of the swap.

Algorithm. The event times are given by

$$\{t_0, (t_m)_{m=1,\ldots,M}\}.$$

We divide the algorithm into 3 steps.

Step 1 (Boundary condition). We have that

$$\underbrace{X(t_M)}_{>t_M} = 0,$$

as there are no exercises after the maturity.

Step 2 (Loop). We enter the loop at t_M (included) and exit at t_0 (not included):

$$\underbrace{t_0}_{\text{end}} \longleftarrow \underbrace{t_M}_{\text{begin}}.$$

We consider the iteration:

$$\underbrace{X(t_m)}_{?} \longleftarrow \underbrace{X(t_{m+1})}_{\text{known}},$$

where

$$X(t_{m+1}) = \underbrace{X(t_{m+1})}_{>t_{m+1}}$$

is the value to continue (the value of the potential payments if we exercise after t_{m+1}). We have that

$$\underbrace{X(t_{m+1})}_{>t_m} = \max(\underbrace{X(t_{m+1})}_{>t_{m+1}}, V(t_{m+1})),$$

where $V(t_{m+1})$ is the value of the swap issued at t_{m+1}. This value has been already computed elsewhere. Finally,

$$\underbrace{X(t_m)}_{>t_m} = \mathcal{R}_{t_m}\big(\underbrace{X(t_{m+1})}_{>t_m}\big).$$

Step 3 (After the loop). We return $X(t_0) = \underbrace{X(t_0)}_{>t_0}$.

Putable and callable bond

Coupon bond:

> N : the notional.
>
> R : the coupon rate.
>
> δt : the interval of time between the payments given as year fraction.
>
> M : the total number of coupon payments.

U : the repurchase price of the bond as percentage of the notional. After the coupon payment the issuer of the bond can repurchase the bond from the holder by paying NU. Typically, this payment is greater than the notional ($U > 1$).

L : the redemption price of the bond as percentage of the notional. After the coupon payment the holder of the bond can sell it back to the issuer for amount LN. Typically, this amount is less than the notional ($L < 1$).

Denote by t_0 the current time and by $(t_m)_{m=1,\dots,M}$ the future coupon times:

$$t_i = t_0 + m\delta t, \quad m = 1, \dots, M.$$

1. At maturity $T = t_M$, if the bond has not been terminated before, the owner of the bond receives coupon $RN\delta t$ and notional N.

2. At a coupon time t_m other than the maturity, if the bond has not been terminated before,

   (a) The owner of the bond receives coupon $RN\delta t$.

(b) The owner of the bond has the right to redeem the bond. In this case he receives amount LN from the issuer of the bond and the bond is terminated.

(c) The issuer of the bond has the right to repurchase the bond. In this case the holder of the bond receives amount UN and the bond is terminated.

Note that the events take place in their respective order:

$$\text{coupon} \longrightarrow \text{redemption} \longrightarrow \text{repurchase}.$$

Algorithm. The event times are

$$\{t_0, (t_m)_{m=1,\ldots,M-1}\},$$

where t_0 is the initial time and $(t_m)_{m=1,\ldots,M-1}$ are all payment times except the last one. We divide the algorithm into 3 steps. We multiply on the notional at the end (Step 3).

Step 1 (Boundary condition).

$$X(t_{M-1}) = \underbrace{X(t_{M-1})}_{>t_{M-1},>t_{M-1}} = B(t_{M-1}, t_{M-1} + \delta t)(1 + R\delta t).$$

Step 2 (Loop). We enter the loop at t_{M-1} (included) and exit at t_0 (not included):

$$\underbrace{t_0}_{\text{end}} \longleftarrow \underbrace{t_{M-1}}_{\text{begin}}.$$

We consider the iteration:

$$\underbrace{X(t_m)}_{?} \longleftarrow \underbrace{X(t_{m+1})}_{\text{known}},$$

where

$$X(t_{m+1}) = \underbrace{X(t_{m+1})}_{>t_{m+1},>t_{m+1}}$$

is the value to continue (the value of coupons paid after t_{m+1} if we exercise after t_{m+1}).

The sequence of events at t_{m+1}:

$$\text{coupon} \longrightarrow \text{redemption} \longrightarrow \text{repurchase}.$$

We go backward in time and thus start at the end.

$$\begin{aligned}
\underbrace{X(t_{m+1})}_{>t_{m+1},>t_m} &= \max(\min(\underbrace{X(t_{m+1})}_{>t_{m+1},>t_{m+1}}, U), L), \\
\underbrace{X(t_{m+1})}_{>t_m,>t_m} &= \underbrace{X(t_{m+1})}_{>t_{m+1},>t_m} + R\delta t,
\end{aligned}$$

and then that

$$\underbrace{X(t_m)}_{>t_m,>t_m} = \mathcal{R}_{t_m}\big(\underbrace{X(t_{m+1})}_{>t_m,>t_m}\big).$$

Step 3 (After the loop). We return $NX(t_0) = N\underbrace{X(t_0)}_{>t_0}$.

Futures on cheapest bond to deliver

This problem is motivated by the existing futures contract on US treasury bonds.

Input: the parameters of the futures contract.

- T : the maturity of the futures contract.
- M : the number of settlement times between today and the maturity.
- **Bonds to deliver** with indexes $j = 1, \dots, J$. We assume that all the bonds are issued at T (the maturity of the futures contract). The parameters of the bond with index j have the form:
  - N_j : the notional.
  - R_j : the coupon rate.
  - $(\delta t)_j$: the interval of time between the payments given as year fraction.
  - M_j : the number of coupon payments.

Output: futures price $F(t_0)$ computed at the initial time.

We assume that the settlement times are given by

$$t_m = t_0 + m\delta t, \quad m = 1, \dots, M,$$

where t_0 is the initial time and

$$\delta t = \frac{T - t_0}{M}.$$

Notice that the settlement times include T, but do not contain t_0.

The futures contract involves the following transactions:

1. It costs nothing to enter into either a long or a short position in the futures contract at t_0.
2. At time t_m before maturity, $m = 1, \dots, M-1$,
   (a) the buyer (long position) pays futures price $F(t_{m-1})$ established at the previous trading day,
   (b) the seller (short position) pays futures price $F(t_m)$ established at the current trading day.
3. At maturity $T = t_M$
   (a) the buyer (long position) pays futures price $F(t_{M-1})$ established at previous trading day,
   (b) the seller (short position) delivers one of the available coupon bonds. Note that the seller has the right to choose which bond to deliver.

Algorithm. The event times are

$$\{t_0, (t_m)_{m=1,\dots,M}\},$$

where t_0 is the initial time and $(t_m)_{m=1,\dots,M}$ are all futures times. We divide the algorithm into 3 steps.

Step 1 (Boundary condition).

$$F(t_M) = \min_{j=1,\dots,J} V_j(t_M),$$

where $V_j(t_M)$ is the price of jth bond at the issue time t_M:

$$V_j(t_M) = N_j \left\{ \left(\sum_{m=1}^{M_j} B(t_M, t_M + m(\delta t)_j) \right) R_j(\delta t)_j + B(t_M, t_M + M_j(\delta t)_j) \right\},$$

and $B(s,t)$ is the discount factor computed at s for maturity t.

Step 2 (Loop). We enter the loop at t_M (included) and exit at t_0 (not included):

$$\underbrace{t_0}_{\text{end}} \longleftarrow \underbrace{t_M}_{\text{begin}}.$$

We consider the iteration:

$$\underbrace{F(t_m)}_{?} \longleftarrow \underbrace{F(t_{m+1})}_{\text{known}}.$$

We have that

$$F(t_m) = \frac{1}{B(t_m, t_{m+1})} \mathcal{R}_{t_m}(F(t_{m+1})),$$

because

$$0 = \mathcal{R}_{t_m}(F(t_{m+1}) - F(t_m)) = \mathcal{R}_{t_m}\big(F(t_{m+1})\big) - B(t_m, t_{m+1})F(t_m).$$

Step 3 (After the loop). We return $F(t_0)$.