Problems for the Quantathon

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

Up-and-out floor

Underlying floor:

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

U : the upper bound for the float (LIBOR) rate.

Assume that today is the issue time of the interest rate floor. Denote this time by t_0. The payment times of the floor are given by

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

The up-and-out floor generates the same cash flow as the interest rate floor up to (and including) the payment time, when the float rate is greater then the barrier. After this time, the option is terminated. In other words, if we denote by τ the first payment time t_m, when float rate $r(t_m, t_m + \delta t)$ between t_m and $t_m + \delta t$ is greater than U, then for a payment time t_j:

1. If $t_j \leq \tau$, then the holder gets standard floor payment

$$N \max(R\delta t - r(t_{j-1}, t_j)\delta t, 0).$$

2. If $t_j > \tau$, then the holder gets nothing.

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). We denote by $B(s,t)$ and $r(s,t)$ the discount factor and the float rate, respectively, at s for maturity t. We recall the identity:

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

From this identity we deduce that

$$\begin{aligned} U > r(t,t+\delta t) &\iff 1+U\delta t > 1 + r(t,t+\delta t)\delta t \\ &\iff B(t,t+\delta t)(1+U\delta t) > 1 \\ &\iff B(t,t+\delta t) > L, \end{aligned}$$

where $L = 1/(1+U\delta t)$ is the lower barrier for the discount factor.

Step 1 (Boundary condition).

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

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 floorets paid after t_{m+1} if possible barrier events happen after t_{m+1}).

We have that

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

and then that

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

and

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

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

American put on the futures price of a zero-coupon bond

N : the notional amount of the bond.

U : the maturity of the bond.

T : the maturity of the futures contract ($T < U$).

M : the number of futures times after the issue time.

K : the strike of the put option.

We denote by

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

the time difference between two adjacent futures times, where t_0 is the initial time. The set of the exercise times for the option has the form:

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

where $t_M = T$. Its intrinsic value is given by

$$V(t_m) = \max(K - F(t_m), 0),$$

where $F(t_m)$ is the futures price at t_m. We recall that a long position in the futures contract has zero entry cost and generates the cash flow:

1. $F(t_m) - F(t_{m-1})$ at t_m if $m = 1, \ldots, M-1$.
2. The purchase of the zero-coupon bond with maturity U and notional N at $t_M = T$ for $F(t_{M-1})$.

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

$$\begin{aligned}\underbrace{X(t_M)}_{>t_M} &= 0,\\ F(t_M) &= NB(t_M, U),\end{aligned}$$

where $B(s,t)$ is the discount factor 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:

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

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}) and $F(t_{m+1})$ is the futures price. We have that

$$\underbrace{X(t_{m+1})}_{>t_m} = \max\big(\underbrace{X(t_{m+1})}_{>t_{m+1}}, K - F(t_{m+1})\big).$$

Finally,

$$\begin{aligned}\underbrace{X(t_m)}_{>t_m} &= \mathcal{R}_{t_m}\big(\underbrace{X(t_{m+1})}_{>t_m}\big),\\ F(t_m) &= \frac{1}{B(t_m, t_{m+1})}\mathcal{R}_{t_m}\left(F(t_{m+1})\right),\end{aligned}$$

because

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

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

Cancellable interest rate swap, where LIBOR rate is set in arrears

N : the notional.

R : the fixed rate in the swap.

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.

Brief description: the holder of the option has the right to cancel the swap at any payment time (after the payments). The fixed payments are as in the usual swap. However, the float payments are set in *arrears*. This means that the float payment at a given payment time is determined by the LIBOR rate computed at *this* payment time (in the case of standard swap the float payment is set at the *previous* payment time).

We denote by $(t_m)_{m=1,\ldots,M}$ the payment times of the swap:

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

At payment time t_m, the following events take place:

1. One side pays "float" interest $NL(t_m, t_m + \delta t)\delta t$, where $L(t_m, t_m + \delta t)$ is the float (LIBOR) rate at t_m for the period of δt years.
2. Another side pays "fixed" interest $NR\delta t$.
3. After the payments, the option holder can terminate the contract.

Note that the first payment of the swap always takes place. In other words, the swap can not be canceled at issue time.

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 the payment times of the swap. We divide the algorithm into 3 steps.

We multiply on the notional at the end (Step 3). We set

$$\alpha_{\text{side}} = \begin{cases} 1, & \text{we pay fixed,} \\ -1, & \text{we pay float.} \end{cases}.$$

Step 1 (Boundary condition).

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

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

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

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

$$\underbrace{X(t_{m+1})}_{>t_{m+1},>t_m} = \max\big(\underbrace{X(t_{m+1})}_{>t_{m+1},>t_{m+1}}, 0\big)$$

and then that

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

where $Y(t_{m+1})$ is the value of the current swaplet:

$$Y(t_{m+1}) = \alpha_{\text{side}} \left(L(t_{m+1}, t_{m+1} + \delta t)\delta t - R\delta t\right)$$
$$= \alpha_{\text{side}} \left(\frac{1}{B(t_{m+1}, t_{m+1} + \delta t)} - (1 + R\delta t) \right).$$

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

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

Finally,

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

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

Double-up swap

Brief description : the interest rate swap, whose holder is given the right to double the swap payments as soon as the float rate falls below a barrier.

Parameters of underlying swap

N : the notional.

R : the fixed rate in the swap.

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.

L : the lower barrier for the float (LIBOR) rate.

We assume that today is the issue time of the swap and denote this time by t_0. The payment times are given by

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

Let τ be the first payment time the float (LIBOR) rate with period δt falls below L:

$$\tau = \min\{t_m : \; r(t_m, t_m + \delta t) \leq L, \quad m = 1, \ldots, M-1\}.$$

At and after time τ, that is, at payment times $t_n \geq \tau$, the holder is given the right to double the notional (like an American option). The exercise at t_n doubles the future swap payments (both float and fixed), that is, the payments at

$$t_n + \delta t, t_n + 2\delta t, \ldots, t_M.$$

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).

$$\underbrace{Y(t_{M-1})}_{>t_{M-1}} = \underbrace{X(t_{M-1})}_{>t_{M-1}} = 0,$$

where the notations are explained on the next step.

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:

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

where

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

is the value of to continue of the American swaption incorporated in the doubling of the notional, that is, the value of the additional future payments if the barrier has been crossed, but the option has not been exercised before or at t_{m+1}, and

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

is the value of the additional payments if the barrier has not been crossed before or at t_{m+1}. We have that

$$\begin{aligned}
\underbrace{Y(t_{m+1})}_{>t_m} &= \max(\underbrace{Y(t_{m+1})}_{>t_{m+1}}, \mathrm{swap}(t_{m+1}, M-(m+1), R, \delta t)), \\
\underbrace{X(t_{m+1})}_{>t_m} &= \underbrace{X(t_{m+1})}_{>t_{m+1}} + \big(\underbrace{Y(t_{m+1})}_{>t_m} - \underbrace{X(t_{m+1})}_{>t_{m+1}}\big) 1_{\{B(t_{m+1}, t_{m+1}+\delta t) \geq U\}},
\end{aligned}$$

where

- $\mathrm{swap}(t, m, R, \delta t)$ is the value of the swap with notional 1, period δt, fixed rate R, and m payments issued at time t.
- U is the upper barrier for the discount factors:

$$U = \frac{1}{1 + L\delta t}.$$

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

Finally,

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

Step 3 (After the loop). We add the original swap:

$$X(t_0) = X(t_0) + \mathrm{swap}(t_0, M, R, \delta t),$$

and return $NX(t_0)$.

Capped resettable floater with repayment option at reset times

$N:$ the notional.

$R:$ the repayment price (in percentage of the notional). Usually, $R > 1$.

C : the cap rate.

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

M : the total number of coupon payments.

K : the number of coupon payments before next reset. The ratio M/K should be an integer ≥ 2.

δL : the spread over Libor.

D_0 : the initial coupon rate. It determines the coupons paid up to and including the first reset time (the same as Kth payment time).

We assume that today is the issue time of the bond and denote this time by t_0. The payment times are given by

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

and the reset times for the coupon rates are given by

$$s_j = t_{j\times K} = t_0 + jK\delta t, \quad j = 1, \ldots, J,$$

where

$$J = \frac{M}{K} - 1.$$

1. At any coupon time t_m (whether it is a reset time or not) the holder of the bond receives the coupon

$$N\delta t \times D_{j-1},$$

   where D_{j-1} is the coupon rate determined at the previous reset time $s_{j-1} < t_m$. The initial rate D_0 is used if $t_m \leq s_1$.

2. If t_m is also a reset time s_j, that is,

$$m = j \times K \quad \text{for some } j = 1, \ldots, J,$$

   then

   (a) the issuer of the bond can cancel the contract by paying the repayment amount NR.

(b) the value of the coupon rate changes to

$$D_j = \min(L(s_j, s_j + K\delta t) + \delta L, C),$$

where $L(s,t)$ is the LIBOR rate computed at s for maturity t. The coupon rate D_j is then used for the computation of the coupons paid at times $s_j + \delta t$, $s_j + 2\delta t$, $\ldots$, $s_j + K\delta t$.

3. If the contract has not been terminated before, then at maturity t_M the holder receives the above coupon (set at the last reset time) and the notional.

Algorithm. The event times are given by

$$\big\{s_0 = t_0, \underbrace{s_1, \ldots, s_J}_{\text{reset times}}\big\},$$

where $s_0 = t_0$ is the initial time and $s_1, \ldots, s_J$ are the reset times:

$$s_j = s_0 + jK\delta t, \quad j = 1, \ldots, J.$$

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

We use the formulas for the reset coupon rates and float rates:

$$\begin{aligned} D_j &= \min(L(s_j, s_j + K\delta t) + \delta L, C), \\ L(s_j, s_j + K\delta t) &= \left(\frac{1}{B(s_j, s_j + K\delta t)} - 1\right)\frac{1}{K\delta t}, \end{aligned}$$

were $B(s,t)$ 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 1 (Boundary condition). We are at the last reset time. We get the notional at maturity and coupons up to and including the maturity.

$$X(s_{J-1}) = \underbrace{X(s_{J-1})}_{>s_J, >s_J} = B(s_J, s_J + K\delta t) + D_J\delta t \sum_{k=1}^{K} B(s_J, s_J + k\delta t).$$

Step 2 (Loop). We enter the loop at s_J (included) and exit at s_0 (not included):

$$\underbrace{s_0}_{\text{end}} \longleftarrow \underbrace{s_J}_{\text{begin}}.$$

We consider the iteration:

$$\underbrace{X(s_j)}_{?} \longleftarrow \underbrace{X(s_{j+1})}_{\text{known}},$$

where

$$X(s_{j+1}) = \underbrace{X(s_{j+1})}_{>s_{j+1},>s_{j+1}}$$

is the value to continue (the value of the payments made after s_{j+1} if no exercises were made before or at s_{j+1}). We have that

$$\begin{aligned}\underbrace{X(s_{j+1})}_{>s_{j+1},>s_j} &= \min\big(R, \underbrace{X(s_{j+1})}_{>s_{j+1},>s_{j+1}}\big), \\ \underbrace{X(s_j)}_{>s_{j+1},>s_j} &= \mathcal{R}_{s_j}\big(\underbrace{X(s_{j+1})}_{>s_{j+1},>s_j}\big).\end{aligned}$$

We add the coupons paid until and including the next reset time:

$$\underbrace{X(s_j)}_{>s_j,>s_j} = \underbrace{X(s_j)}_{>s_{j+1},>s_j} + D_j \delta t \sum_{k=1}^{K} B(s_j, s_j + k\delta t).$$

Step 3 (After the loop). We return $NX(s_0) = N \underbrace{X(s_0)}_{>s_0,>s_0}$.

Putable bond with resettable coupon

This is a variant of the so-called *ratchet bonds*.

N : the notional.

R_0 : the initial coupon rate.

R_1 : the reset coupon rate ($R_1 < R_0$).

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

M : the number of coupon payments.

L : the redemption price of the bond as the percentage of the notional; usually, $L < 1$.

Brief description: after a coupon payment, the issuer can reset the coupon rate from the original (higher) value R_0 to the reset (lower) value R_1. At any payment time greater or equal the reset time and smaller than the maturity, the holder can sell the bond back to the issuer for the redemption value LN.

We denote by t_0 the initial time and by $(t_m)_{m=1,\ldots,M}$ the coupon times:

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

Let t_m be the current time. Depending on the past, there are 3 possibilities:

1. If $t_m < t_M$ and the coupon rate has not been reset before, then
   (a) The holder receives original coupon $NR_0\delta t$.
   (b) The issuer can reset original coupon rate R_0 to lower coupon rate R_1. If he does so, then the holder has the right to sell the bond back to the issuer at redemption price LN.
2. If $t_m < t_M$ and the coupon rate has been already reset to R_1, but the bond has not been terminated, then
   (a) The holder receives reset coupon $NR_1\delta t$.
   (b) The holder can sell the bond back to the issuer for redemption amount LN.
3. If $t_m = t_M$ (the maturity) and the bond has not been terminated before, then the holder of the bond receives the coupon payment (original or reset) as well as notional amount N.

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).

$$X(t_{M-1}) = \underbrace{X(t_{M-1})}_{>t_{M-1},>t_{M-1}} = B(t_{M-1}, t_{M-1} + \delta t)(1 + R_0 \delta t),$$
$$Y(t_{M-1}) = \underbrace{Y(t_{M-1})}_{>t_{M-1},>t_{M-1}} = B(t_{M-1}, t_{M-1} + \delta t)(1 + R_1 \delta t).$$

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:

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

where

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

is the value to continue before the reset (the value of coupons paid after t_{m+1} if we reset after t_{m+1}) and

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

is the value to continue after the reset (the value of coupons paid after t_{m+1} if the bond has not been terminated before or at t_{m+1}).

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

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

We go backward in time and thus start at the end. First, we account for redemption and reset:

$$\underbrace{Y(t_{m+1})}_{>t_{m+1},>t_m} = \max(\underbrace{Y(t_{m+1})}_{>t_{m+1},>t_{m+1}}, L),$$
$$\underbrace{X(t_{m+1})}_{>t_{m+1},>t_m} = \min(\underbrace{X(t_{m+1})}_{>t_{m+1},>t_{m+1}}, \underbrace{Y(t_{m+1})}_{>t_{m+1},>t_m}).$$

Then we add the current coupon:

$$\underbrace{X(t_{m+1})}_{>t_m,>t_m} = \underbrace{X(t_{m+1})}_{>t_{m+1},>t_m} + R_0\delta t,$$
$$\underbrace{Y(t_{m+1})}_{>t_m,>t_m} = \underbrace{Y(t_{m+1})}_{>t_{m+1},>t_m} + R_1\delta t.$$

Finally, we rollback to the previous event time:

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

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