Breakable TRS with Monte-Carlo method

I – Model framework

This paper aims to introduce a proxy for the management of TRS that offers the customer the option to terminate the contract in advance at a given price by the sum of:

- Intrinsic value of performance leg
- Accrual of the financing leg
- Exit penalty (time function)

The framework by which we want to determine the best product proxy, both in terms of pricing and management, is to replicate with a portfolio consisting of a normal TRS and a Bermudian TRS-option.

Below we will take the customer's point of view (receiver performance TRS)

1/TRS pricing equation

The swap has the following market value (as a function of time), skipping the classical risk neutral expectation calculation steps:

$$TRS(t) = \left[\frac{P(t,T_n)}{P_{t_0}} - 1\right] ZC(t,T_n) - \sum_{k=1}^{n} \tau_k \cdot \left(l(t,T_{k-1},T_k) + s_{TRS}\right) \cdot ZC(t,T_k)$$

with following assumptions for the model :

- Bullet performance leg
- Floating rate upfront (Libor-like)
- Scaled nominal

2/TRSwaption payoff

This Bermudian option must:

- Cancel the swap
- Pay the customer for the performance of the equity underlying
- Provide an exit penalty

In the case of the exercise, then the payoff for the customer must be equal to:

$$H^{TRS}(t) = -\left[\frac{P(t,T_n)}{P_{t_0}} - 1\right] ZC(t,T_n) + \sum_{k=1}^{n} \tau_k \cdot \left(l(t,T_{k-1},T_k) + s_{TRS}\right) \cdot ZC(t,T_k) + \left[\frac{P(t)}{P_{t_0}} - 1\right] - fee(t)$$

To simplify the model, let's take following assumptions:

1. First, and to avoid useless calculation complexities for the target, let's assume the equity underlying won't deliver income (seen as an index then), in that case the forward price is:

$$P(t,T_n) \simeq P(t)\frac{\left(1+r_{repo}(T_n-t)\right)}{ZC(t,T_n)}$$

Where $\mathrm{r}_{\mathrm{repo}}$ is the repo margin of the index.

2. Let's also assumed that the libor rate forecasting curve is the same as discounting (no multi-curve framework). In that case, thanks to classical Libor rate definition we have:

$$\sum_{k=1}^{n} \tau_k \cdot l(t,T_{k-1},T_k) \cdot ZC(t,T_k) \;=\; 1 - ZC(t,T_n)$$

With these approximations the payoff in case of exercise can be written as:

$$H^{TRS}(t) \simeq \sum_{k=1}^{n} \tau_k \cdot s_{TRS} \cdot ZC(t,T_k) \; - \frac{P(t)}{P_{t_0}} \cdot r_{repo} \cdot (T_n - t) - fee(t)$$

$$\Leftrightarrow H^{TRS}(t) \simeq s_{TRS} \cdot A_{1,n}(t) \; - \frac{P(t)}{P_{t_0}} \cdot r_{repo} \cdot (T_n - t) - fee(t)$$

With the annuity notation $A_{1,n}(t) \;=\; \sum_{k=1}^{n} \tau_k \cdot ZC(t,T_k)$

$fee(t)$ function will be assumed to be deterministic and linked to time to maturity :

$$\begin{cases} fee(t) \;=\; f \cdot \dfrac{(T_n - t)}{T_n} \\ f \; as \; user - defined \; factor \end{cases}$$

3/Risk factors framework

Risk factors of this specific option are:

1. The repo margin.
2. The underlying equity of the TRS
3. Interest rates

The third interest rate factor is assumed to be less impacting, in first approach, than two previous two factors. This assumption will be discussed in next sections but this simplified model will assume a constant and deterministic interest rate for annuity calculation.

The model for this proxy option estimation is based on the random evolution of the first two factors according to the following SDE system:

$$\begin{cases} dr_{repo} = \sigma_{repo}\left(r_{repo} + \gamma\right) dW_t^{r_{repo}} \\ dS = r_{repo} S dt + \sigma_S S dW_t^S \end{cases}$$

$$\Rightarrow \begin{cases} r_{repo}(t) \;=\; (r_{repo}(0) \;+\; \gamma) e^{-\frac{\sigma_{repo}{}^2}{2}t \,+ \sigma_{repo} W_t^{r_{repo}}} \; - \gamma \\ S(t) = \; S(0) e^{(r_{repo}(t) \,-\, -\frac{\sigma_S{}^2}{2})t \,+ \sigma_S W_t^S} \end{cases}$$

With γ specific shift (sometimes useful to reflect negatives repo rates) and $(W_t^{r_{repo}}, W_t^S)$ uncorellated brownian motions.

4/TRSwaption pricing

The value of the bermudean TRSwaption is :

$$TRSwaption(t_0) = \max_{\tau\in\mathbb{T}} \mathbb{E}^{\mathbb{Q}}\left(\frac{H^{TRS}(\tau)}{B(\tau)}\right)$$

where τ is a stopping time taking values in $\mathbb{T} = \{\mathrm{T}_0 \ldots \mathrm{T}_\mathrm{M} = \mathrm{T}\}$, with T as TRS maturity

To find the option price at time t_0 we apply backward induction. Starting from the option value at maturity T_M, we know the option value is equal to the payoff:

$$TRSwaption(T_M) = max(H^{TRS}(T_M), 0)$$

For any other time T_m with 0 ≤ m < M we assume by induction that $TRSwaption(\mathrm{T}_{\mathrm{m}+1})$ is known. If we define the continuation value at time T_m by:

$$\begin{cases} CV^{TRSwaption}(T_m) = \mathbb{E}^{\mathbb{Q}}\left(\dfrac{B(T_m)}{B(T_{m+1})} \cdot TRSwaption(T_{m+1}) \middle| \mathcal{F}_{T_m}\right) \\ \mathcal{F}_{T_m} \; = \; \sigma(r_{repo}(T_m), S(T_m), A_{1,n}(T_m)) \end{cases}$$

then the value of the Bermudan option at time T_m is given by the following formula, following non-arbitrage condition for american/bermudean option and of course if the option is still alive at T_m:

$$TRSwaption(T_m) = max\left(H^{TRS}(T_m), CV^{TRSwaption}(T_m)\right)$$

Given $H^{TRS}(T_m)$, the main computation goal is solving $CV^{TRSwaption}(\mathrm{T}_\mathrm{m})$, which will be done through Monte-Carlo method and especially using with Longstaff-Schwartz algorithm.

5/Algorithm path

Following n paths is generated for each option's components:

$$\begin{cases} r_{repo}^k(T_m) \\ S^k(T_m) \\ A_{1,n}(T_m), one\ unique\ path \end{cases} , k = \ 1 \ldots n; m = 0 \ldots M$$

For each sample path k the option value at maturity $TRSwaption(T_M)$ can be computed

Assume now that the set of option values $TRSwaption\left(T_{m+1}, r_{repo}^k(T_{m+1}), S^k(T_{m+1}), A_{1,n}(T_{m+1})\right) = TRSwaption^{\mathrm{k}}(T_{m+1})$, for k = 1, . . . , n, is known.

To compute $CV^{TRSwaption}(T_m)$, we regress the discounted option values at time T_{m+1} on a set of polynomial functions $P_j(T_m, r_{repo}^k(T_m), S^k(T_m), A_{1,n}(T_m)) = P_j^k(T_m)$, for j = 1, . . . , J, where J is a fixed number, that we'll choose ≤ 4.

The solving process follows the next linear regression scheme:

$$\begin{cases} independent\ variables\ x_{kj} = P_j^k(T_m) \\ dependant\ variables\ y_k = \dfrac{B(T_m)}{B(T_{m+1})} \cdot TRSwaption^k(T_{m+1}) \end{cases}$$

$$\Rightarrow\ y_k = \sum_{j=1}^{J} \beta_j x_{kj} + \varepsilon_k, with\ regression\ parameters\ \beta_j\ and\ independent\ error\ term\ \varepsilon_k$$

With the "mandatory" assumption $\frac{B(T_m)}{B(T_{m+1})} \cdot TRSwaption(T_{m+1}) \approx \sum_{j=1}^{J} \beta_j P_j(T_m)$, the option continuation value can be approximated :

$$CV^{TRSwaption,k}(T_m) = \mathbb{E}^{\mathbb{Q}}\left(\frac{B(T_m)}{B(T_{m+1})} \cdot TRSwaption(T_{m+1}) \middle| \mathcal{F}_{T_m}^k\right)$$

$$\Rightarrow\ CV^{TRSwaption,k}(T_m) \approx \mathbb{E}^{\mathbb{Q}}\left(\sum_{j=1}^{J} \beta_j(T_m) P_j(T_m) \middle| \mathcal{F}_{T_m}^k\right)$$

$$\Rightarrow\ CV^{TRSwaption,k}(T_m) \approx \sum_{j=1}^{J} \beta_j(T_m) P_j^k(T_m)$$

That's the end of the (mathematical) road : thanks to this proxy continuation value, we can compute the option value itself :

$$TRSwaption^k(T_m) = max\left(H^{TRS,k}(T_m), CV^{TRSwaption,k}(T_m)\right)$$

II – Implementation and numerical results

- Program will be split between risk factors modeling framework and option pricing itself.
- Python program will be coded first, followed by C++ program.