Pricing and Hedging Callable Libor Exotics in Forward Libor Models

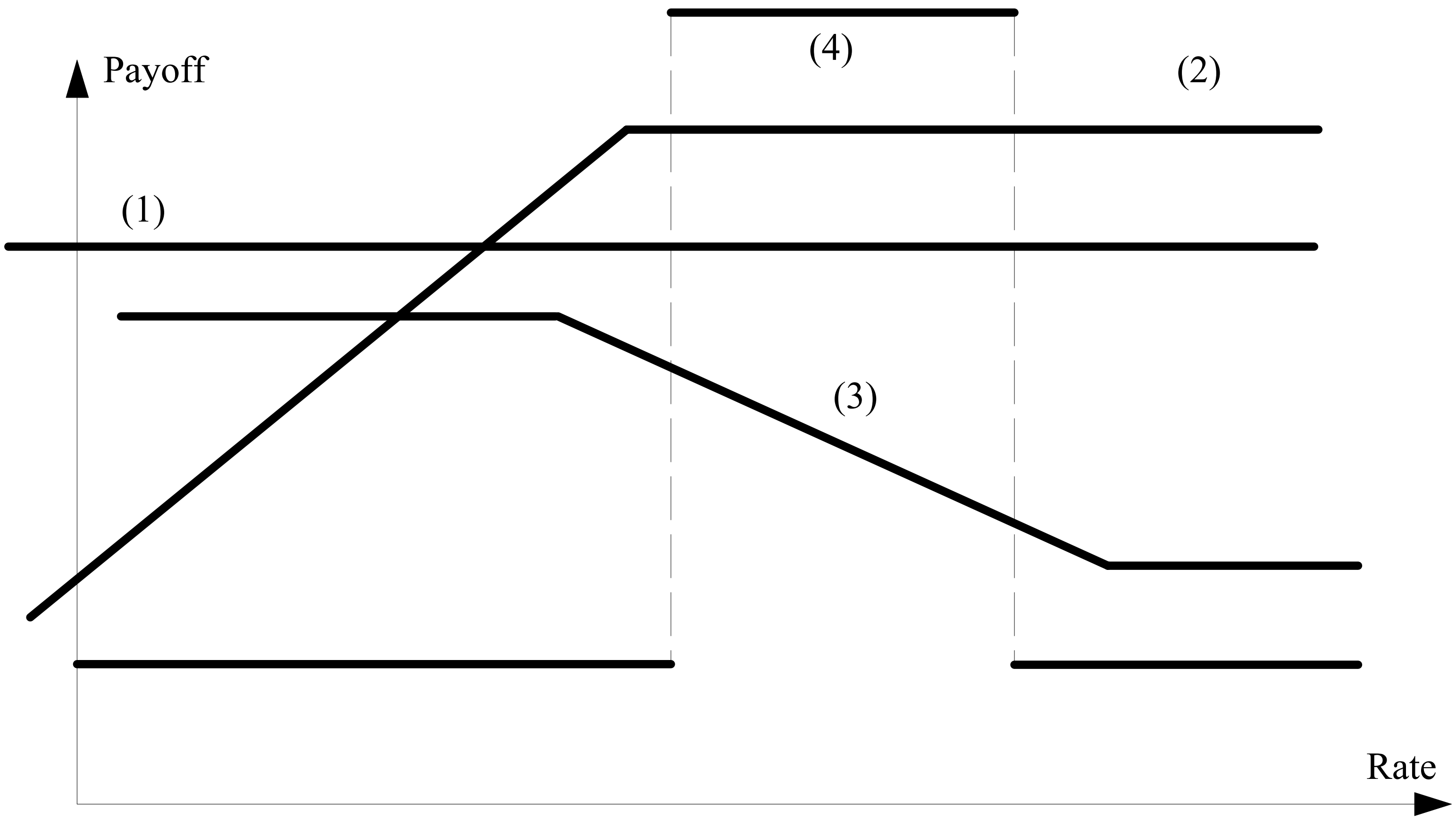
CAP Workshop

Vladimir Piterbarg
Bank of America

1 Market for callable Libor exotics

- Callable Libor exotic products (CLE): notes (bonds) with funky coupons
- Typical structure:
  - An investor buys a 10y note at par;
  - First year: high fixed coupon;
  - Years 2 – 10: coupon = Libor rate plus spread capped at certain level;
  - The bank has the right to call the note by returning the principal back to the investor, every year after the first.
- Investors motivation: Can get a large, above-market current coupon in year one, as well as a potential for above market returns in years 2 – 10
- Why would the bank pay high coupons? The option to call the note (cancel the deal after the first year) is valuable! In effect, the investor sells an option to the bank for current yield (premium). The bank can monetize the option by delta hedging ("gamma scalping") it.
- New structures get invented all the time. Make the coupon appealing to customers + make the option to cancel as valuable as possible

2 Coupon payoffs I



(1) Bermuda swaption
(2) Callable capped floater
(3) Callable inverse floater
(4) Callable range accrual

3 Coupon payoffs II

Let L be a Libor or a CMS rate.

- Bermuda swaptions. Coupon is fixed at some level c
- Callable capped floaters. If spread is s and cap is c, the payoff is

$$\min\left(L+s,c\right)$$

- Callable inverse floater. If strike is s, cap is c and floor is f then the payoff is

$$\max\left(\min\left(s-L,c\right),f\right).$$

- Callable range accrual. The coupon pays a fixed rate or a Libor rate accrued over the number of days a reference rate is within a certain range. If the payment rate is fixed and equal to c, and the lower/upper bound are l and u respectively, then the coupon pays

$$c\times\sharp\left\{t:L_t\in[l,u]\right\}.$$

- Callable CMS coupon diff. If S_1 and S_2 are two CMS rates with different tenors (i.e. 10 year and 2 year), g is the scaling factor, c and f are cap and floor, the coupon payment is equal to

$$\max\left(\min\left(g\times\left(S_1-S_2\right),c\right),f\right).$$

4 Formal definition of CLE

Tenor structure

$$\begin{aligned} 0 &= T_0 < T_1 < \ldots < T_N, \\ \delta_n &= T_{n+1} - T_n. \end{aligned}$$

Value of coupon that fixes at time T_i and pays at T_{i+1},

$$C_i(t).$$

Libor (bank funding) rate that fixes at time T_i,

$$L_i(t).$$

A callable Libor exotic is a Bermuda style option with exercise dates $T_1, \ldots, T_{N-1}$. Exercised at time T_n – get all coupons fixing on or after T_n.

The exercise value for exercise opportunity n, at time t, is equal to

$$E_n(t) = \sum_{i=n}^{N-1} B_t \mathbf{E}_t \left(B_{T_{i+1}}^{-1} \times (C_i(T_i) - L_i(T_i)) \times \delta_i \right).$$

So the bank can exercise the option, and if it does so on date T_n, the PV of the swap it gets is equal to $E_n(T_n)$.

Define

$$X_i(T_i) = (C_i(T_i) - L_i(T_i)) \times \delta_i.$$

5 Pricing and hedging challenges

- Highly nontrivial dependence of CLEs on volatility structure require BGM-type models
- Only Monte-Carlo available.
- Computing values of Bermuda-style options is hard.
- Risk parameters – even harder!
- Solutions to these problems are discussed in the talk.
- See [Pit03b], [Gla03]

6 Some useful notations

- Probability space $(\Omega, \mathbf{P}, \mathcal{F})$
- Zero-coupon bonds $P(t,T)$
- Numeraire B_t
- Filtration of sigma-algebras $\{\mathcal{F}_t\}$
- Pricing operator π. For arbitrary $X \sim \mathcal{F}_T$,

$$\pi_t(X) = B_t \mathbf{E}_t \left(B_T^{-1} X\right)$$

- Tenor structure

$$0 = t_0 < t_1 < \ldots < t_M$$

- Primary Libor rates

$$\bar{F}(t) = (F_0(t), \ldots, F_{M-1}(t)), \quad F_n(t) = \frac{P(t,t_n) - P(t,t_{n+1})}{(t_{n+1} - t_n) P(t,t_{n+1}).}.$$

7 BGM model and extensions

$W^{T_{n+1}}(\cdot)$ a Brownian motion under the T_{n+1}-forward measure

Lognormal BGM:

$$dF_n(t) = F_n(t)\,\lambda_n(t)\;dW^{T_{n+1}}(t)\,.$$

Skew-extended BGM:

$$dF_n(t) = \phi\left(F_n(t)\right)\lambda_n(t)\;dW^{T_{n+1}}(t)\,. \tag{1}$$

Stochastic Volatility BGM:

$$\begin{aligned} dz(t) &= \theta\left(z_0 - z(t)\right)\,dt + \varepsilon\sqrt{z(t)}\,dB(t)\,, \\ dF_n(t) &= \sqrt{z(t)}\phi\left(F_n(t)\right)\lambda_n(t)\;dW^{T_{n+1}}(t)\,, \end{aligned}$$

Use discrete money-market numeraire B_t by

$$\begin{aligned} B_{t_0} &= 1, \\ B_{t_{n+1}} &= B_{t_n} \times \left(1 + \left(t_{n+1} - t_n\right)F_n\left(t_n\right)\right), \quad 1 \le n < M, \\ B_t &= P\left(t, t_{n+1}\right)B_{t_{n+1}}, \quad t \in \left[t_n, t_{n+1}\right]. \end{aligned}$$

Spot Libor measure: use B as the numeraire.

8 Pricing callable Libor exotics in BGM

- Define $H_n(t)$ to be the value of a "sub-CLE" with the exercise dates $T_n, \ldots, T_{N-1}$. The same as the "hold" value of the original CLE if it has not been exercised up to and including the date T_n.
- Main recursion for exercise and hold values

$$\begin{aligned}
E_n(T_n) &= P(T_n, T_{n+1}) X_n(T_n) + \pi_{T_n} E_{n+1}(T_{n+1}), \\
H_{n-1}(T_{n-1}) &= \pi_{T_{n-1}} \max\{H_n(T_n), E_n(T_n)\}, \\
H_{N-1} &\equiv 0, \\
E_N &\equiv 0, \\
n &= N-1, \ldots, 1.
\end{aligned}$$

For $n = 1$ we obtain the value $H_0(0)$, the value of the CLE that we are after.

9 Pricing callable Libor exotics as barriers with an optimized barrier I

- Formally,

$$H_n(T_n) = \operatorname*{ess\,sup}_{\xi \in \mathcal{T}_n} \mathbf{E}_{T_n} B_{T_\xi}^{-1} E_\xi(T_\xi).$$

$\mathcal{T}_n$ is a set of all stopping times that exceed n.

- Supremum over all barrier options.
- The solution to this series of problems is given by the optimal exercise time index $\eta = \eta(\omega)$,

$$\eta(\omega) = \min\{n \geq 1 : \omega \in R_n\} \wedge N,$$

where R_n are exercise region at time T_n,

$$R_n = \{\omega \in \Omega : H_n(T_n, \omega) \leq E_n(T_n, \omega)\}, \quad 1 \leq n \leq N-1.$$

- The CLE value

$$H_0(0) = \mathbf{E}_0\left(B_{T_\eta}^{-1} E_\eta(T_\eta)\right) = \mathbf{E}_0\left(\sum_{n=\eta}^{N-1} B_{T_{n+1}}^{-1} X_n\right).$$

10 Pricing callable Libor exotics as barriers with an optimized barrier II

- Recall

$$
\begin{aligned}
H_0(T_0) &= \operatorname*{ess\,sup}_{\xi \in \mathcal{T}_0} \mathbf{E}_0 B_{T_\xi}^{-1} E_\xi(T_\xi) \\
&= \mathbf{E}_0 \left(B_{T_\eta}^{-1} E_\eta(T_\eta) \right).
\end{aligned}
$$

- Any stopping time gives a lower bound.
- Replacing optimal exercise regions with estimates $\tilde{R}_n$ and η with $\tilde{\eta}$ we get a lower bound on CLE value.
- The closer the estimated exercise region $\tilde{R}_n$ to the actual one, the tighter the lower bound on the value.
- Pricing in Monte-Carlo (lower bound):
  1. Pre-simulate some paths
  2. Estimate exercise regions for each exercise time
     (a) Optimization of parametrized boundaries (Anderson)
     (b) Replace expectations with regressions (Longstaff-Schwartz)
  3. Simulate additional paths (main simulation), and compute the CLE value as the value of a barrier option with a given set of exercise regions

11 Exercise boundary and risk sensitivities

- Risk sensitivities – bump one of the inputs and revalue.
- Keep the exercise boundary when bumping and revaluing!
- Risk to parameter x with current value x_0. For each x there is an optimal exercise boundary $\zeta(x)$. The value of a CLE for x is then given by $h(\zeta(x), x)$ where $h(\zeta, x)$ is the value of a barrier option for the boundary ζ and parameter value x. Then the risk number is equal to

$$\left.\frac{\partial}{\partial x} h(\zeta(x), x)\right|_{x=x_0} = \left.\frac{\partial}{\partial \zeta} h(\zeta, x_0)\right|_{\zeta=\zeta(x_0)} \times \left.\frac{\partial}{\partial x}\zeta(x)\right|_{x=x_0} + \left.\frac{\partial}{\partial x} h(\zeta(x_0), x)\right|_{x=x_0}.$$

  Important:

$$\left.\frac{\partial}{\partial \zeta} h(\zeta, x_0)\right|_{\zeta=\zeta(x_0)} = 0,$$

  because $\zeta(x_0)$ is the *optimal* one for the CLE.

- Faster (do not need to recompute)
- More accurate (no noise from boundary calculations)
- The full derivative of h with respect to x is equal to the partial one while keeping the exercise boundary constant.

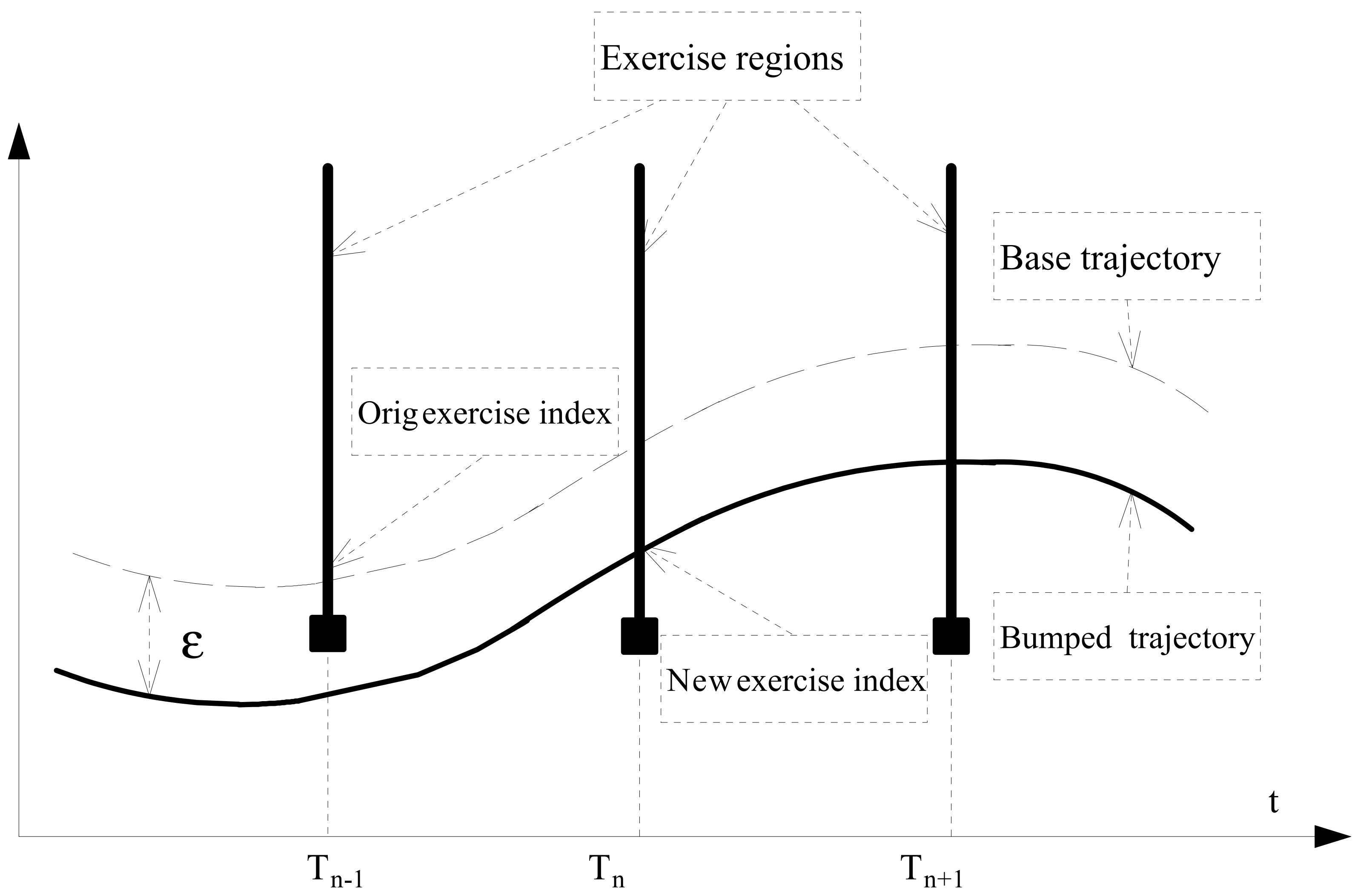
12 Deltas and why they are hard to obtain

- Interest rate deltas – changes in the value of the CLE with respect to $F_n(0)$, $n = 1, \ldots, M-1$. Natural bucketing.
- Recall that he value is computed as a sum over simulated paths ω_j, $j = 1, \ldots, J$,

$$\tilde{H}_0 = J^{-1} \sum_{j=1}^{J} \sum_{i=1}^{N-1} \left[B_{T_{i+1}}^{-1}(\omega_j) X_i(\omega_j) 1_{i \geq \tilde{\eta}(\omega_j)} \right]$$

- Two effects as we bump one of the rates slightly
  - Smooth: change in the values X_i or $B_{T_n}^{-1}$.
  - Jumpy: change in $\tilde{\eta}(\omega_j)$, can add/delete a whole coupon for a path affected. The quantities $1_{i \geq \tilde{\eta}(\omega_j)}$ do not depend smoothly on initial interest rate curve
- It it the second effect that makes simulation error large for Greeks (less smooth – higher error). See picture below

13 Non-smooth dependence of exercise index on initial conditions



14 Pathwise deltas I

- Define Δ_m to be the delta with respect to $F_m(0)$, *for each realization* ω,

$$\Delta_m X(\omega) = \frac{\partial X(\omega)}{\partial F_m(0)}.$$

- Valuation recursion:

$$B_{T_{n-1}}^{-1} H_{n-1}(T_{n-1}) = \mathbf{E}_{T_{n-1}} \max\left\{B_{T_n}^{-1} H_n(T_n), B_{T_n}^{-1} E_n(T_n)\right\}$$

Differentiate through (technical conditions to change the order of expectation and differentiation)

$$\begin{aligned}\Delta_m\left(B_{T_{n-1}}^{-1} H_{n-1}(T_{n-1})\right) &= \mathbf{E}_{T_{n-1}} 1_{\{H_n(T_n) > E_n(T_n)\}} \Delta_m\left(B_{T_n}^{-1} H_n(T_n)\right) \\ &\quad + \mathbf{E}_{T_{n-1}} 1_{\{E_n(T_n) > H_n(T_n)\}} \Delta_m\left(B_{T_n}^{-1} E_n(T_n)\right).\end{aligned}$$

- Unwrap the recursion to obtain

$$\begin{aligned}\Delta_m H_0(0) &= \sum_{n=1}^{N-1} \mathbf{E}_0\left(\prod_{i=1}^{n-1} 1_{\{H_i(T_i) > E_i(T_i)\}} \times 1_{\{E_n(T_n) > H_n(T_n)\}} \times \Delta_m\left(B_{T_n}^{-1} E_n(T_n)\right)\right) \\ &= \sum_{n=1}^{N-1} \mathbf{E}_0\left(1_{\{\eta = n\}} \times \Delta_m\left(B_{T_n}^{-1} E_n(T_n)\right)\right) \\ &= \mathbf{E}_0\left(\sum_{i=\eta}^{N-1}\left(\Delta_m\left(B_{T_{i+1}}^{-1} X_i\right)\right)\right).\end{aligned}$$

15 Pathwise deltas II

- Valuation:

$$H_0(0) = \mathbf{E}_0\left(\sum_{i=\eta}^{N-1} B_{T_{i+1}}^{-1} X_i\right).$$

- Can differentiate through, and keep the exercise boundary constant:

$$\Delta_m H_0(0) = \mathbf{E}_0\left(\sum_{i=\eta}^{N-1} \Delta_m\left(B_{T_{i+1}}^{-1} X_i\right)\right).$$

- Replace η with an estimate $\tilde{\eta}$, get an estimate of the delta,

$$\tilde{\Delta}_m H_0(0) = \mathbf{E}_0\left(\sum_{i=\tilde{\eta}}^{N-1} \Delta_m\left(B_{T_{i+1}}^{-1} X_i\right)\right).$$

- A converging estimate as $\tilde{\eta} \to \eta$.
- Significant reduction in noise (the problem discussed in the slide above eliminated). Time savings because deltas are computed in the same simulation as the value. Requires 1/16 - 1/32 the number of paths compared to the standard method.
- Valuation as barriers, yet pathwise deltas can be used (only for *optimal* boundary!)

16 Pathwise deltas III

- Apply chain rule to get $\Delta_m \left(B_{T_{i+1}}^{-1} X_i\right)$. For example

$$\begin{aligned}
\Delta_m P\left(t, t_m, t_{m+1}\right) &= \Delta_m \frac{1}{1+\tau_m F_m(t)} = \frac{\partial}{\partial F_m(t)} \left(\frac{1}{1+\tau_m F_m(t)}\right) \times \frac{\partial F_m(t)}{\partial F_m(0)} \\
&= -\frac{\tau_m}{\left(1+\tau_m F_m(t)\right)^2} \Delta_m F_m(t).
\end{aligned}$$

- The values $\Delta_m F_n(t)$ can be simulated in the same simulation as $F_n(t)$. Recall (under the spot measure)

$$dF_n(t) = \lambda_n(t)\, \phi\left(F_n(t)\right) \left(\mu\left(t, \bar{F}(t)\right) dt + dW(t)\right). \tag{2}$$

Differentiate through,

$$\begin{aligned}
d\Delta_m F_n(t) &= \lambda_n(t) \left[\sum_k \frac{\partial \phi\left(F_n(t)\right)}{\partial F_k(t)} \left(\Delta_m F_k(t)\right)\right] \left(\mu\left(t, \bar{F}(t)\right) dt + dW(t)\right) \\
&\quad + \lambda_n(t)\, \phi\left(F_n(t)\right) \left[\sum_k \frac{\partial \mu\left(t, \bar{F}(t)\right)}{\partial F_k(t)} \left(\Delta_m F_k(t)\right)\right] dt.
\end{aligned}$$

Initial conditions

$$\Delta_m F_n(0) = \delta_{mn}.$$

- Replace the drift with its "along the forward" approximation to speed up calculations as the derivatives can be precomputed.

See [GZ99], [Pit03a]

17 “Sausage” Monte-Carlo I

- Pathwise differentiation method not always applicable (discontinuity of coupons, system constraints, inaccurate exercise boundary estimates).
- “Digital” features in CLE payoff: use payoff smoothing via conditional expectations. Tailored to the callable structure.
- Main idea. Suppose we need to estimate

$$V = \int_0^1 f(x)\, dx.$$

- Regular Monte-Carlo (x_j are from the uniform distribution)

$$V_0 = J^{-1} \sum_{j=1}^{J} f(x_j).$$

- Pre-integration around sample points:

$$V_\varepsilon = J^{-1} \sum_{j=1}^{J} (2\varepsilon)^{-1} \int_{x_j-\varepsilon}^{x_j+\varepsilon} f(\xi)\, d\xi.$$

18 “Sausage” Monte-Carlo II

- For small ε, V_ε is close to V_0. But smoother:

$$\begin{aligned} V_\varepsilon &= J^{-1} \sum_{j=1}^{J} f_\varepsilon (x_j), \\ f_\varepsilon (x) &= (2\varepsilon)^{-1} \int_{x-\varepsilon}^{x+\varepsilon} f(\xi)\, d\xi. \end{aligned}$$

- Partial integration $f \to f_\varepsilon$ smooths the payoff. Smoother dependence on parameters!
- Smoother payoffs – smaller simulation error for Greeks.
- To compute integrals over $[x_j - \varepsilon, x_j + \varepsilon]$ use linearization/approximations/other tricks.

19 “Sausage” Monte-Carlo III

- Pre-integrate the payoff (approximately) along each simulated path of interest rates (“sausage”). Instead of

$$\tilde{H}_0 \approx J^{-1} \sum_{j=1}^{J} v_j$$
$$v_j = \sum_{i=1}^{N-1} \left[B_{T_{i+1}}^{-1} (\omega_j) X_i (\omega_j) 1_{i \geq \tilde{\eta}(\omega_j)} \right]$$

use

$$\tilde{H}_0 \approx J^{-1} \sum_{j=1}^{J} v_j^{\varepsilon},$$
$$v_j^{\varepsilon} = \mathbf{E} \left(\sum_{i=1}^{N-1} \left[B_{T_{i+1}}^{-1} (\omega) X_i (\omega) 1_{i \geq \tilde{\eta}(\omega)} \right] \middle| A_j^{\varepsilon} \right),$$

where (use $x(t, \omega)$ to denote a vector of Markovian state variables)

$$A_j^{\varepsilon} = \{ \omega : \| x(T_i, \omega) - x(T_i, \omega_j) \| < \varepsilon \quad \forall i = 1, \ldots, N-1 \}.$$

- Instead of “hard” exercise/no exercise rule, have a concept of a fuzzy, or partial, exercise (conditioned on being in the sausage).
- The probability of exercise on each date, conditioned on being in the sausage, can be analytically estimated (use approximate conditional independence/uniformity inside the sausage.)

- Final formula: For each path ω, instead of

$$v_j = \sum_{i=1}^{N-1} \left[B_{T_{i+1}}^{-1} (\omega_j) \, X_i (\omega_j) \, 1_{i \geq \tilde{\eta}(\omega_j)} \right]$$

we get

$$\begin{aligned} v_j &= \sum_{i=1}^{N-1} B_{T_{i+1}}^{-1} (\omega_j) \times X_i (\omega_j) \times (1 - q_i (\omega_j)) \, , \\ q_i &= q_{i-1} \times (1 - p_i) \, , \\ p_i &= \min \left(\max \left(\frac{\hat{E}_i - \hat{H}_i + \delta_i}{2\delta_i}, 0 \right), 1 \right) . \end{aligned}$$

- Here
  - $\hat{E}_i$, $\hat{H}_i$ are proxy exercise and hold values (from pre-simulation),
  - p_i is a marginal exercise probability for date T_i,
  - q_i is a probability of not exercising up to time T_i,
  - δ_i is the smoothing parameter, a known function of ε.
- Note that we get the standard formula if we set $\varepsilon = 0$.
- Instead of "all or nothing" for each path we get a weighted average of

21 Markovian projection of a BGM model I

- Idea that was popular for a while: Find an approximation to a BGM model that allows for a PDE formulation
- Two approximations:
  - BGM volatility structure need to be separable
  - The drifts in SDE's for Libor rates need to be dealt with
- Recall BGM under spot Libor measure

$$dF_n(t)/F_n(t) = \mu_n\left(t, \bar{F}(t)\right) dt + \lambda_n(t)\, dW(t).$$

- To extract Markovian factor, assume $\lambda_n(t)$ is of special "separable" form (Libor-specific scalings and common time-dependent component)

$$\lambda_n(t) = c_n v(t).$$

- For the drift, replace with deterministic one (more elaborate schemes possible)

$$\mu_n\left(t, \bar{F}(t)\right) \mapsto \mu_n\left(t, \bar{F}(0)\right).$$

- Define a (Markovian) factor

$$dX(t) = v(t)\, dW(t).$$

22 Markovian projection of a BGM model II

- Then each Libor rate can be represented as a deterministic function of the factor

$$F_n(t) = F_n(0) \exp\left(c_n X(t) + \int_0^t \mu_n\left(s, \bar{F}(0)\right) ds - \frac{1}{2} c_n^2 \int_0^t v^2(s)\, ds \right).$$

- All contracts are functions of the Markovian factor (state variable) $X(t)$
- A combination of BGM calibration process with speed and accuracy of a PDE-based scheme?
- Bad news: The approximation is too crude for all but the shortest (10y absolute max) CLE
- Good news: Good enough to use as a control variate for Monte-Carlo
- Typically the number of paths required reduced by a factor of 10 – 100

23 Model-based variance reduction

- Take a BGM model, approximate with a “separable” one
- Compute the value of a contract three ways:
  - Monte-Carlo with the original model (V_{orig})
  - Monte-Carlo with the Markovian projection (V_{MC}) *using the same paths*
  - PDE with the Markovian projection (V_{PDE}).
- The corrected value is equal to

$$V_{Corrected} = V_{orig} - \alpha \left(V_{MC} - V_{PDE} \right)$$

- The noise in $V_{orig} - V_{MC}$ is reduced (if the approximation of the vol/skew structure is good, i.e. if the correlation is high).
- The correction is unbiased,

$$\begin{aligned} \mathbf{E} V_{MC} &= V_{PDE}, \\ \mathbf{E} V_{Corrected} &= \mathbf{E} V_{orig}. \end{aligned}$$

24 Obtaining vegas I

- The BGM model is calibrated to the whole swaption grid. Need vegas to each point in the grid
- Direct approach:
  - Bump each point in the swaption grid in turn
  - Re-calibrate the BGM model
  - Compute the change in value of the CLE
- Does not work. Why? Calibration error usually of the same order of magnitude as a reasonable volatility bump (0.1 Black vol)
- Also too slow

25 Obtaining vegas II

- Indirect approach. Suppose v be the vector of swaption volatilities and σ be the vector of all model volatilities (call them BGM vols)
  - Bump all BGM vols in turn
  - Compute the change in value of the CLE for each bump
  - Recompute swaption vols in the "bumped" model (closed-form formulas available)
  - Re-express the vegas in terms of swaption vols using linear algebra. If h is the value of the CLE then we have matrix equation

$$\frac{\partial h}{\partial \sigma} = \frac{\partial h}{\partial v} \times \frac{\partial v}{\partial \sigma}.$$

    that we solve for $\frac{\partial h}{\partial v}$.

- Use methods described above (pathwise differentiation, "sausage" Monte-Carlo) to compute "model" vegas $\frac{\partial v}{\partial \sigma}$.

References

[Gla03] Paul Glasserman. *Monte Carlo Methods in Financial Engineering*. Springer, 2003.

[GZ99] Paul Glasserman and Xiaoling Zhao. Fast greeks in forward Libor models. *Journal of Computational Finance*, 3:5–39, 1999.

[Pit03a] Vladimir V. Piterbarg. Computing deltas of callable LIBOR exotics in a forward LIBOR model. To appear in The Journal of Computational Finance, 2003.

[Pit03b] Vladimir V. Piterbarg. A Practitioner's guide to pricing and hedging callable LIBOR exotics in forward LIBOR models. SSRN Working paper, 2003.