

A stochastic volatility model for callable CMS swaps and translation invariant path dependent derivatives*

Claudio Albanese[†] Manlio Trovato[‡]

May 22, 2006

Abstract

We present a stochastic volatility term structure model based on a continuous time lattice which allows for a numerically stable and quite efficient methodology to price fixed income exotics. We present numerical applications to bermudan swaptions and callable CMS swaps. We then extend the model to translation invariant path dependent payoffs by means of an efficient dimensional reduction technique and show its application to callable range accruals and target redemption notes.

1 Introduction

The history of interest rate models is characterized by a long series of turns. The Black formula for caplets and swaptions was designed to take as underlying a single forward rate under the appropriate forward measure. This approach has the advantage to lead to a simple pricing formula for European options but also the limitation of not being extendable to callable contracts. To have a more consistent model, short rate models were introduced in (Cox, Ingersoll & Ross 1985), (Vasicek 1977), (Black & Karasinski 1991) and (Hull & White 1993). Next came LIBOR market models, also known as correlation models. First introduced in (Brace, Gatarek & Musiela 1996) and (Jamshidian 1997), forward LIBOR models affirmed themselves as a flexible Montecarlo pricing methodology providing non-parametric fits to both term structures for interest rates

*This research work was carried out in partial fulfillment of the requirements for doctoral studies at Imperial College London and was supported in part by a research grant of Merrill Lynch, which is gratefully acknowledged. The content of this article is not necessarily related to any opinions that might be held or practices that might be followed at Merrill Lynch. The authors would also like to thank Oliver Chen and Laurent Borredon for their useful comments.

[†]DEPARTMENT OF MATHEMATICS, IMPERIAL COLLEGE LONDON. *E-mail address:* claudio.albanese@imperial.ac.uk

[‡]GLOBAL MARKETS DERIVATIVES ANALYTICS, MERRILL LYNCH LONDON, AND DEPARTMENT OF MATHEMATICS, IMPERIAL COLLEGE LONDON. *E-mail address:* manlio_trovato@ml.com

and at-the-money Black volatilities for either caplets or a family of swaptions of fixed tenor. Various extensions of forward LIBOR models aim at incorporating volatility smiles, calibrating thus to the so called "volatility cube", given by the Black volatility of swaptions as a function of strike, maturity and tenor. Local volatility extensions were pioneered in (Andersen & Andreasen 2000). A stochastic volatility variation is proposed in (Andersen & Brotherton-Ratcliffe 2001), and is further extended in (Andersen & Andreasen 2002). A different approach to stochastic volatility forward LIBOR models is described in (Joshi & Rebonato 2003). Jump-diffusion forward LIBOR models are treated in (Glasserman & Merener 2001), (Glasserman & Kou 1999). A calibration framework is proposed in (Piterbarg 2003).

Modeling stochastic volatility within the framework of LIBOR market models is a challenging task from an implementation viewpoint due to the intrinsic simulation noise of Montecarlo methods which make the calculation of hedge ratios a particularly arduous task. In a strive to achieve a better understanding of the volatility dynamics and vega sensitivities, in recent years we witnessed the emergence and recognition as a market standard in the fixed income domain of the SABR model by (Hagan, Kumar, Lesniewski & Woodward 2002). Although probably not the ultimate solution, SABR is an important stepping stone which allows one to better understand the stochastic volatility process from an analytical viewpoint. Likewise to the Black formula approach, SABR models as underlier a single forward rate under the corresponding forward measure. A limitation of this framework is that it is not suitable to price callable swaps and Bermuda swaptions. Pricing inconsistencies may also emerge with SABR due to the fact that the model is solved by means of expansions which are asymptotic (as opposed to convergent) and are thus characterized by irreducible approximation errors.

Within the standard SABR model one also observes calibration difficulties with European options. Implied swaption volatilities with very large strikes are probed by constant maturity swaps (CMS). These exotic structures receive a floater, the coupon being a spread over LIBOR, and pay the equilibrium swap rate of a fixed tenor prevailing at each coupon date. An analysis of the CMS leg coupon structure leads to the conclusion that CMS contracts are particularly sensitive to the asymptotic behavior of implied volatilities for very large strikes. Market CMS rates actually drive the option market in extreme strike regions and indicate that implied volatilities flatten out and converge asymptotically to a constant. This behavior is not consistent with the rapidly diverging asymptotics which are implied by SABR.

In this article, we attempt to go beyond SABR by introducing a stochastic volatility short rate model which has the correct asymptotic behavior for implied swaption volatilities and can thus be used for callable CMS swaps as well as Bermuda swaptions. Our model is solved on continuous time lattices of a new type. Discretization schemes of similar type have been previously discussed in (Albanese & Kuznetsov 2005) and (Albanese & Kuznetsov 2003), and applied to models for which the spectrum of the Markov generator can be computed in analytically closed form. While retaining the insights provided by the spectral

analysis treatment in the previous papers, here we renounce to analytic solvability and follow instead a non-parametric approach. The new idea is to leverage not on the ability to evaluate special functions but instead on numerical linear algebra routines. In particular, no part of our calculations require the use of Montecarlo or asymptotic methods and prices and hedge ratios are very stable, even for extreme strike levels.

This technique yields a gain in model flexibility and allows us to extend the application domain and refine the calibration procedure. The criterion guiding our methodology is time-stationarity and calibration is carried out in three separate stages: in the first we achieve an approximate fit to the term structure of interest rates and the term structure of implied at-the-money volatilities for a wide range of market cap and swaption contracts. In the second we make the fit to interest rates and implied at-the-money volatility by introducing a minimum amount of time dependence through a time-dependent coordinate change. This provides a very reasonable but not completely accurate fit to out-of-the-money implied volatilities. This issue is addressed in a third calibration step aimed at achieving perfect consistency with out-of-the-money skews, but this time only in the region of the volatility cube spanned by a selected family of hedging vehicles. For this purpose, we make use of a sequence of measure changes which resemble a technique introduced in the context of the Markov functional model, see (Hunt, Kennedy & Pelsser 2000), except that we work consistently in the risk neutral measure as opposed to a terminal bond as numeraire asset. As a result of the construction, our model is nearly stationary over time horizons in excess of 30 years.

Having constructed a model for interest rates with volatility regimes, we present novel numerical techniques for the pricing of path dependent payoffs. Typically, in order to evaluate path dependent derivatives, one would be tempted to resort to Montecarlo methods, which allow for an easy way to describe the path dependent nature of a payoff. However Montecarlo methods are prone to evaluation biases for callable structures and present numerical challenges to achieve stable greeks. The naive extension of lattice methods to path dependent payoffs requires the increase in dimensionality of the problem and the discretization of path states, as discussed in text books like (Hull 2005). However, although this conceptual extension is valid in its nature, this approach may prove too onerous from a numerical performance point of view. Also, care must be taken in the way the discretization is performed, as this may lead to numerical instabilities. The application to finance of numerical techniques for the valuation of path dependent payoffs on a lattice is a field which lay fairly unexplored, with only few contributions available in the literature, as the domain decomposition approach presented by (Hu 2005). In this paper we propose an extension of our stochastic volatility model to a particular class of path dependent payoffs. We show its application to translation invariant payoffs, where the path dependent state does not depend on the previous history of the path state itself, both in the case where the path dependency can be evaluated in continuous increments, like callable range accruals, and in the case where the path dependency has a discrete nature, like Target Redemption Notes (TARNs).

The structure of this article is as follows: in the first part we present the construction of the stochastic volatility model, a technique largely based on our previous work (Albanese & Trovato 2005) but with a refinement of the stochastic volatility regimes process. We also present useful numerical techniques for a stable matrix inversion computation, crucial for the regularity of our time continuous lattice. We then show how the model can be extended to translation invariant path dependent payoffs, by means of a lifting operation on the Markov generator and present an efficient algorithm for the diagonalization of large matrices in blocks, by means of partial Fourier transforms. We present a practical application to the pricing of callable range accruals, with continuous path state increments, and TARNs, with discrete path state increments.

2 The model

2.1 The conditional local volatility processes

We introduce $M + 1$ states of volatility. The process conditioned to stay in one of such states $a \in \{0, \dots, M\}$ is related to the solution r_{at} of the following equation:

$$dr_{at} = \kappa_a(\theta_a - r_{at})dt + \sigma_a r_{at}^{\beta_a} dW. \quad (1)$$

In the functional analysis formalism we use, these SDEs are associated to the Markov generators

$$\mathcal{L}_a^r = \kappa_a(\theta_a - r_{at}) \frac{\partial}{\partial r} + \frac{\sigma_a^2 r_{at}^{2\beta_a}}{2} \frac{\partial^2}{\partial r^2}. \quad (2)$$

To build a continuous time lattice, we discretize the short rate variable and constrain it to belong to a finite lattice Ω_a containing $N + 1$ points $r(x) \geq 0$, where $x = 0, \dots, N$, $r(0) = 0$ and the following ones are in increasing order, not necessarily equally spaced. The discretized Markov generator $\mathcal{L}_{\Omega_a}^r$ is defined as the operator represented by a tridiagonal matrix such that

$$\begin{aligned} \sum_y \mathcal{L}_{\Omega_a}^r(x, y) &= 0 \\ \sum_y \mathcal{L}_{\Omega_a}^r(x, y)(y - x) &= \kappa_a(\theta_a - r(x)) \\ \sum_y \mathcal{L}_{\Omega_a}^r(x, y)(y - x)^2 &= \sigma_a^2 r(x)^{2\beta_a}. \end{aligned}$$

The above equations impose probability conservation and ensure that first and second moment of the r_{at} conditional local volatility process are correctly recovered. In our example, we select an inhomogenous grid of $N = 70$ points spanning short rates from 0% to 50%, in such a way to obtain a finer grid in the region around the spot rate level. We also choose to work with 8 states for volatility. Although our model is more complex than a simple local volatility

process, it is convenient to describe our resolution method in the specific case of the operator $\mathcal{L}_{\Omega a}^r$ with constant a . This method can then be generalized and is at the basis of other extensions such as the introduction of jumps, as we show in the sequel. We write the diagonalization problem of the Markov generator $\mathcal{L}_{\Omega a}^r$, conditioned on a fixed value of a , in the following matrix form:

$$\mathcal{L}_{\Omega a}^r = U\Lambda U^{-1} \quad (3)$$

where U is the matrix having as columns the right eigenvectors and Λ is the diagonal matrix having the eigenvalues λ_i as elements. We set boundary conditions in such a way that there is absorption at the endpoints, and in particular at the point corresponding to zero rates $r(0) = 0$. The diagonalization problem and its numerical implementation are discussed in the sequel, as well as our choice of boundary conditions. Taking this for granted, we now focus on our construction and note that the key to our methodology is the remark that, if the Markov generator is diagonalizable, we can apply an arbitrary function F to it by means of the following formula:

$$F(\mathcal{L}_{\Omega a}^r) = UF(\Lambda)U^{-1} \quad (4)$$

This formula is at the basis of the so-called "functional calculus". As Ito's formula regarding functions of stochastic processes is central in the mathematical Finance for diffusion processes, functional calculus for Markov generators plays a pivotal role in our framework for stochastic volatility models. This formula has several applications. An immediate one allows us to express the pricing kernel $u(r(x), t; r(y), T)$ of the process as follows:

$$u(r(x), t; r(y), T) = (e^{(T-t)\mathcal{L}_{\Omega a}^r})(x, y) = \sum_n e^{\lambda_n(T-t)} u_n(x) v_n(y) \quad (5)$$

where $u_n(x)$ and $v_n(y)$ are the right and left eigenvectors respectively.

2.2 Introducing jumps

At this stage of the construction one has the option to add jumps. Although in the examples discussed in this paper we are mostly focused on long dated callable exotics for which we find that the impact of jumps can be safely ignored, adding jumps involves negligible additional complexities and is thus worth considering and implementing in other situations. To add jumps, one can follow the following procedure which accounts for the need to assign different intensities to up-jumps and down-jumps. Jump processes are associated to a special class of stochastic time changes given by monotonously non-decreasing processes T_t with independent increments. These time changes are known as Bochner subordinators and are characterized by a Bochner function $\phi(\lambda)$ such that

$$E_0 [e^{-\lambda T_t}] = e^{-\phi(\lambda)t} \quad (6)$$

For example, the case of the variance gamma process which received much attention in Finance corresponds to the function

$$\phi(\lambda) = \frac{\mu^2}{\nu} \log \left(1 + \lambda \frac{\nu}{\mu} \right) \quad (7)$$

where μ is the mean rate and ν is the variance rate of the variance gamma process. The generator of the jump process can be expressed using functional calculus as the operator $-\phi(-\mathcal{L}_{\Omega a}^r)$. To produce asymmetric jumps, we specify the two parameters in (7) differently for the up and down jumps and compute separately two Markov generators

$$\mathcal{L}_{\pm} = -\phi_{\pm}(-\mathcal{L}_{\Omega a}^r) = -U_{\pm} \phi_{\pm}(-\Lambda) U_{\pm}^{-1}$$

where:

$$\phi_{\pm}(\lambda) = \frac{\mu_{\pm}^2}{\nu_{\pm}} \log \left(1 + \lambda \frac{\nu_{\pm}}{\mu_{\pm}} \right)$$

The new generator for our process with asymmetric jumps is obtained by combining the two generators above

$$\mathcal{L}_{j\Omega a}^r = \begin{pmatrix} 0 & \dots & \dots & \dots & 0 \\ \mathcal{L}_-(2,1) & d(2,2) & \mathcal{L}_+(2,3) & \dots & \mathcal{L}_+(2,n) \\ \vdots & \vdots & \ddots & \dots & \vdots \\ \mathcal{L}_-(n-1,1) & \mathcal{L}_-(n-1,2) & \dots & d(n-1,n-1) & \mathcal{L}_+(n-1,n) \\ 0 & 0 & \dots & \dots & 0 \end{pmatrix}$$

Here the element of the diagonal are chosen in such a way to satisfy probability conservation:

$$d(x, x) = - \sum_{y \neq x} \mathcal{L}_{j\Omega a}^r(x, y) \quad (8)$$

Also notice that we have zeroed out the elements in the matrix at the upper and lower boundary: this ensures that there is no probability leakage in the process. A justification of this boundary condition choice is reported in the sequel.

In our setting, we choose a short rate as a modeling primitive and we thus do not need to impose a martingale condition. Otherwise, were we working with a forward rate instead, the appropriate method of restoring the martingale condition would be to modify the matrix elements of the resulting generator on the first sub-diagonal and the first super-diagonal.

At this stage of the construction, we have therefore obtained a generator $\mathcal{L}_{j\Omega a}^r$ for the short rate process, whose dynamics is characterized by a combination of state dependent local volatility and asymmetric jumps. We note that the addition of jumps has not increased the dimensionality of the problem and is therefore computationally efficient.

2.3 Modeling the dynamics of stochastic volatility regimes

The construction for the volatility regime process proceeds with a similar methodology to the one used for the short rate process, but in this case we model the dynamics of the stochastic volatility regime process α by means of a composition of a mean reverting diffusion process and asymmetric jumps only. The process on the regimes is typically much slower than the process on the state variable r , and it is responsible for driving the long term behaviour of the term structure of the yield curve. The short rate is allowed to transition from one regime to another with low frequency: the diffusion component of the regime process α drives the transition between contiguous regimes, whereas the presence of jumps allows for a non-zero probability to move between non contiguous regimes. Typically, down jumps in the regime process occur with higher frequency than up jumps, a feature captured by the asymmetric nature of jumps in the model.

In the first step of our construction we define the discretized Markov generator for the α process as follows:

$$\mathcal{L}^\alpha = k^\alpha(\theta^\alpha - \alpha_t)\nabla^\alpha + \frac{\nu^2 \alpha_t^{2\gamma}}{2}\Delta^\alpha \quad (9)$$

where ∇^α is the nabla operator, or the discretized first derivative operator, and Δ^α is the Laplacian operator, or the second derivative discretized operator. The α variable is discretized and constrained to belong to a finite lattice Σ containing $M + 1$ points $\alpha(a) = \alpha_a$, where $a = 0, \dots, M$. The discretized Markov generator $\mathcal{L}_\Sigma^\alpha$ is defined as the operator represented by a tridiagonal matrix such that

$$\begin{aligned} \sum_b \mathcal{L}_\Sigma^\alpha(a, b) &= 0 \\ \sum_b \mathcal{L}_\Sigma^\alpha(a, b)(b - a) &= k^\alpha(\theta^\alpha - \alpha(a)) \\ \sum_b \mathcal{L}_\Sigma^\alpha(a, b)(b - a)^2 &= \nu^2 \alpha(a)^{2\gamma} \end{aligned}$$

Next we overlay asymmetric jumps to the process described by the Markov generator $\mathcal{L}_\Sigma^\alpha$. Two Markov generators are computed for the up and down jumps of the volatility regime process

$$\mathcal{L}_{j\pm}^\alpha = -\phi_{\alpha\pm}(-\mathcal{L}_\Sigma^\alpha) = -U_{\alpha\pm}\phi_{\alpha\pm}(-\Lambda_\alpha)U_\alpha^{-1}$$

where:

$$\phi_{\alpha\pm}(\lambda_\alpha) = \frac{\mu_{\alpha\pm}^2}{\nu_{\alpha\pm}} \log\left(1 + \lambda_\alpha \frac{\nu_{\alpha\pm}}{\mu_{\alpha\pm}}\right)$$

and U_α is the matrix having as columns the right eigenvectors of the diagonalization problem $\mathcal{L}_\Sigma^\alpha = U_\alpha \Lambda_\alpha U_\alpha^{-1}$. The new generator for our process with

asymmetric jumps is obtained by combining the two generators above

$$\mathcal{L}_j^\alpha = \begin{pmatrix} d^\alpha(1,1) & \mathcal{L}_+^\alpha(1,2) & \cdots & \cdots & \mathcal{L}_+^\alpha(1,M) \\ \mathcal{L}_-^\alpha(2,1) & d^\alpha(2,2) & \mathcal{L}_+^\alpha(2,3) & \cdots & \mathcal{L}_+^\alpha(2,M) \\ \vdots & \vdots & \ddots & \cdots & \vdots \\ \mathcal{L}_-^\alpha(M-1,1) & \mathcal{L}_-^\alpha(M-1,2) & \cdots & d^\alpha(M-1,M-1) & \mathcal{L}_+^\alpha(M-1,M) \\ \mathcal{L}_-^\alpha(M,1) & \mathcal{L}_-^\alpha(M,2) & \cdots & \mathcal{L}_-^\alpha(M,M-1) & d^\alpha(M,M) \end{pmatrix} \quad (10)$$

Here the element of the diagonal are chosen in such a way to satisfy probability conservation:

$$d^\alpha(a,a) = - \sum_{a \neq b} \mathcal{L}_j^\alpha(a,b) \quad (11)$$

Note that, in this case, we impose reflecting boundary conditions and we ensure that the process has positive drift at state zero and negative drift at state M , so that the process never leaks outside the boundaries.

Out of the two generators we just defined, one for the short rate and one for the stochastic volatility, we form a Markov generator \mathcal{L} acting on functions of both the rate state variable x and the volatility state variable α . This generator has matrix elements given as follows:

$$\mathcal{L}(x, a; y, b) = \mathcal{L}_{j\Omega_a}^r(x, y) \delta_{ab} + \mathcal{L}_j^\alpha(a, b) \delta_{xy} \quad (12)$$

where δ indicates the Kronecker delta.

To gain an intuition on the meaning of the matrix element $\mathcal{L}(x, a; y, b)$ we note that, for each element off the main diagonal, $\mathcal{L}(x, a; y, b)dt$ represents the transition probability to move from a state $r(x)$ in the a volatility regime to a state $r(y)$ in the b volatility regime in an infinitesimal time dt . We are therefore describing a two-dimensional problem: one for the underlying short rate and one for the stochastic volatility regimes.

Since our underlier is a short rate though, we are not interested in the pricing kernel but rather in the discounted transition probabilities given by

$$G(x, a, t; y, b, T) = E \left[e^{-\int_t^T r_s ds} \delta(r_T - r(y)) \delta(\alpha_T - b), \| r_t = r(x), \alpha_t = a \right] \quad (13)$$

This kernel satisfies the following backward equation

$$\frac{\partial}{\partial t} G(x, a, t; y, b, T) + (\mathcal{L}G)(x, a, t; y, b, T) = r(x)G(x, a, t; y, b, T) \quad (14)$$

In functional calculus notations, the solution is given by

$$G(x, a, t; y, b, T) = e^{\mathcal{G}(T-t)}(x, a; y, b) \quad (15)$$

where $\mathcal{G}(x, a; y, b) \equiv \mathcal{L}(x, a; y, b) - r(x)\delta_{xy}$.

The same diagonalization method illustrated above for the local volatility problem can also be applied in this case. One can represent the matrix \mathcal{G} in the form

$$\mathcal{G} = U\Lambda U^{-1} \quad (16)$$

where Λ is diagonal, and write the matrix of discounted transition probabilities as follows

$$e^{\mathcal{G}(T-t)} = U e^{\Lambda(T-t)} U^{-1}. \quad (17)$$

2.4 Measure changes

For a perfect fit to the volatility cube of a selected family of hedging vehicles, we overlay a series of measure changes to fine-tune the value of out-of-the-money implied volatility skews up to fairly remote strike regions. This introduces a degree of time dependence in the model which however is fairly limited, due to the fact that the first calibration step typically produces a very reasonable first approximation with the correct asymptotic behavior, as we discuss below.

Our sequence of measure changes does not correspond to changes of numeraire asset, as is typically done with interest rate derivatives, see (Jamshidian 1993) and (Geman, Karoui & Rochet 1995), but is rather of a slightly different nature, as we work exclusively under the risk neutral measure. Consider an increasing sequence of maturities for European style options for which we wish to obtain a perfect price fit: t_1, t_2, \dots, t_n . The desired measure change is specified by a sequence of functions $H_i(x, a) > 0$ for $i = 1, \dots, n$ such that, for $t \in (t_{i-1}, t_i)$ we have that

$$\frac{\partial}{\partial t} H_i(x, a, t; y, b, t_i) + (\mathcal{L}H_i)(x, a, t; y, b, t_i) = 0 \quad (18)$$

For each $t \in (t_{i-1}, t_i)$, the measure-changed discounted transition probabilities are given by

$$G_{H_i}(x, a, t; y, b, t_i) = \frac{1}{H_i(x, a, t)} G(x, a, t; y, b, t_i) H_i(y, b, t_i) \quad (19)$$

and they satisfy to the following backward equation:

$$\frac{\partial}{\partial t} G_{H_i}(x, a, t; y, b, t_i) + ((\mathcal{L}H_i - r)G_{H_i})(x, a, t; y, b, t_i) = 0 \quad (20)$$

where

$$(\mathcal{L}H_i - r)(x, a, t; y, b, t_i) = \frac{1}{H_i(x, a, t)} (\mathcal{L} - r)(x, a, t; y, b, t_i) H_i(y, b, t_i) + \frac{1}{H_i(x, a, t)} \frac{\partial}{\partial t} H_i(x, b, t) \quad (21)$$

Composing the measure-changed process across time intervals, one obtains the measure changed kernel for the discounted transition probabilities, for $0 \leq i < j \leq n$:

$$G_H(x_i, a_i, t_i; x_j, a_j, t_j) = \sum_{x_{i+1}, \dots, x_{j-1}} \sum_{a_{i+1}, \dots, a_{j-1}} \prod_{k=i+1}^j \frac{H_k(x_k, a_k, t_k)}{H_k(x_{k-1}, a_{k-1}, t_{k-1})} G(x_{k-1}, a_{k-1}, t_{k-1}; x_k, a_k, t_k) \quad (22)$$

where H_k is the measure change function defined at time t_k .

The measure change functions H_k provide the time dependent degree of freedom we need to calibrate to the region of the volatility cube spanned by our selected hedging vehicles. The defined measure-change will depend on the particular family of hedging vehicles we select. The fit to the other regions of the volatility cube, comprising swaptions of different tenors and maturities is qualitatively correct but not quantitatively precise. However, so long as the hedging vehicles of interest to our pricing problem are correctly priced, this discrepancy may not be of substantial importance. Further reducing this discrepancy will require augmenting the number of regimes and refining the model.

3 Numerical implementation

3.1 Boundary conditions

Boundary conditions may be set in such a way to either have absorption or reflection at the endpoints. With both choices we guarantee probability conservation for the process, and as a consequence there exists a zero eigenvalue. We choose to impose absorbing boundary conditions for the short rate process as a way to be conservative in our model implementation. In fact, our objective is to calibrate model parameters in such a way that the probability for the short rate to hit zero is minimal. With absorbing boundary conditions it is possible to have a direct assessment of the probability of this unwanted event, as this probability accumulates at the boundary. With reflecting boundary conditions, instead, the probability of zero or negative rates accumulates at very low interest rate levels, but in this case it can not be directly evaluated: in this region the probability of this unwanted event is mixed with the probability of low short rate values. We find that absorbing boundary conditions give rise to noticeable pathologies only in fairly extreme strikes of the 10y and 20y into 5y JPY swaptions.

For the process on the state variable α , reflecting boundary conditions are a natural choice: the short rate moves from one regime to another according to the dynamics described by the Markov generator $\mathcal{L}_\Sigma^\alpha$ of eq. (10). The short rate should be allowed to reach the boundary regimes $M = 0$ and $M = 7$ and reflect back into the lattice space Σ , when one of these regimes is attained. No attempt is made, in this case, for the short rate to reach the boundaries with minimal probability.

3.2 The diagonalization problem

Given a discretized Markov generator \mathcal{L}_Ω defined on a finite lattice Ω containing $N + 1$ points, we consider the following pair of eigenvalue problems:

$$\mathcal{L}_\Omega u_n = \lambda_n u_n \qquad \mathcal{L}_\Omega^T v_n = \lambda_n v_n$$

where the superscript T denotes matrix transposition, u_n and v_n are the right and left eigenvectors of \mathcal{L}_Ω , respectively, whereas λ_n are the corresponding eigenvalues. Except for the simplest cases, the Markov generator \mathcal{L}_Ω is not a symmetric matrix, hence u_n and v_n are different. Also, in general, the eigenvalues are not real. We are only guaranteed that their real part is non-positive $\text{Re}\lambda_n \leq 0$ and that complex eigenvalues occur in complex conjugate pairs, in the sense that if λ_n is an eigenvalue then its conjugate $\bar{\lambda}_n$ is also an eigenvalue.

There is no guarantee, in the most general case, that there exists a complete set of eigenfunctions. However, the chance that such a complete set does not exist for a Markov generator specified non-parametrically is zero, so we can safely assume that this is the case. In the unlikely case that this assumption is not correct, the numerical linear algebra routines needed to solve our lattice model will identify the problem and a small perturbation of the given operator will suffice to rectify the situation. Assuming completeness, the diagonalization problem can be rewritten in the following matrix form:

$$\mathcal{L}_\Omega = U\Lambda U^{-1} \tag{23}$$

where U is the matrix having as columns the right eigenvectors and Λ is the diagonal matrix having the eigenvalues λ_i as elements.

In most cases, the numerical implementation of the diagonalization problem can be efficiently carried out by means of routines such as `geev` in LAPACK. However, care must be taken in analysing the stability and accuracy of the spectrum. It is possible to evaluate the circles of uncertainty for each eigenvalue by means of perturbation techniques on the Markov generator or with alternative algorithms, as described in (Threfethen 1999).

3.3 Matrix inversion algorithm

Given a matrix S , of dimension $[m \times n]$, the computation of its inverse may generate numerical instabilities due to loss of accuracy when computing the difference between two very large values, a scenario which may occur when large eigenvalues are solutions of the diagonalization problem of the matrix S . An improved numerical algorithm for the computation of the matrix inverse S^{-1} makes use of *PLU* factorization and rebalancing of the matrix elements U . We write the matrix inversion problem in the form:

$$S^{-1} = (PLU)^{-1} = U^{-1}L^{-1}P^{-1}$$

where P is a $[m \times n]$ permutation matrix, L is a $[m \times n]$ or $[m \times m]$ lower triangular matrix with unit diagonals and U is a $[n \times n]$ or $[m \times n]$ upper triangular matrix. By definition the eigenvalues of P and L are all located on the unitary circle, hence the computation of L^{-1} and P^{-1} can be directly performed by calling matrix inversion routines such as `inversez` in LAPACK without incurring in instabilities. We rebalance the elements of matrix U , in such a way to obtain an upper triangular matrix with unit diagonals, by multiplying each column by a vector of constant elements. In particular, for each column y , we set:

$$\bar{U}(x, y) = \frac{U(x, y)}{U(y, y)}$$

This transformation does not change the nature of the problem as the spectrum of matrix \bar{U} is simply a linear combination of the spectrum of matrix U . However, matrix \bar{U} can be inverted with minimal risk of incurring in numerical instabilities.

4 Calibration

Two different approaches are typically adopted in the calibration of interest rates models: global or local calibration. The aim of the former is to solve for a model that, on average, is able to capture the dynamics of the whole market, across all tenors, maturities and strikes, whereas with the latter one can reprice exactly a small set of hedging instruments. A global calibration is, by its nature, a very ambitious problem and therefore is typically only solved in approximation, for example in least square sense. It has the advantage that the same model parameterization can be applied to a wide variety of pricing problems, hence it achieves consistency for the pricing of all derivatives held in a trading book, across tenors, maturities and strikes. The price to pay is the inaccuracy with which individual hedging instruments are repriced. Viceversa, a locally calibrated model is able to capture exactly the dynamics of the hedging vehicles of interest for a given pricing problem, but it needs to be recalibrated on a trade by trade basis.

We adopt an hybrid approach: we look for a stationary model parameterization that, on average, captures the dynamics of the market for a large set of instruments, similarly to what a globally calibrated model would do. However, for each pricing problem, we fit exactly a small set of hedging vehicles, by means of time dependent general measure changes.

4.1 Global calibration step

In a first step (the global step), we calibrate to the yield curve and ATM volatilities of a large set of instruments, spanning the whole volatility surface across all expiries and tenors. This calibration set need not change frequently and it is performed by means of a least square minimization algorithm.

Once we have solved for a best fit using the model above without introducing any explicit time dependency, we then introduce time dependency to achieve a perfect fit to the term structure of forward rates and at-the-money swaption volatilities. As a consequence of this procedure, the degree of time variability of model parameters is kept to a bare minimum and the yield curve is exactly recovered. As we calibrate the model to at-the-money Black volatilities, we find

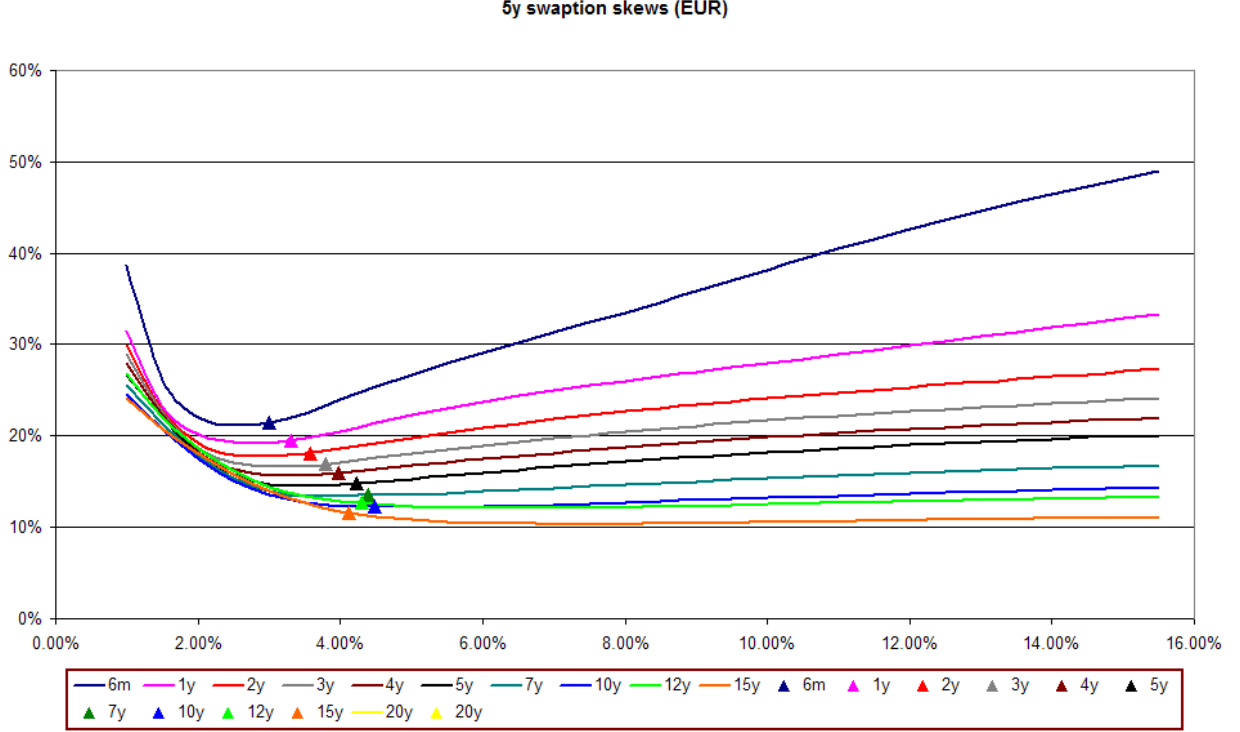


Figure 1: BlackScholes implied volatility smile for 5y EUR tenor swaption across different maturities and strikes

that the high degree of stationarity ensures a qualitatively correct asymptotic behavior for implied swaption volatilities even in the region of extreme strike values. Fig. 1 and Fig. 2 show our results for EUR and JPY denominated swaptions. This qualitative agreement also ensures that the subsequent series of measure changes adds only a small amount of explicit time dependence.

For an exact fit of the term structure of interest rates, we introduce explicit time dependence by combining two operations: a shift of the short rate by a time varying, deterministic function of time and a deterministic time change, i.e.

$$r_t \rightarrow \tilde{r}_t = \dot{f}(t)r_{f(t)} + c(t). \quad (24)$$

Here $f(t)$ is monotonously increasing and $\dot{f}(t)$ denotes its time derivative. Using the new process, discounted transition probabilities can be computed as follows:

$$E \left[e^{-\int_t^T \tilde{r}_s ds} \delta(\tilde{r}_T - \tilde{r}(y)) \delta(\alpha_T - b), \|\tilde{r}_t = \tilde{r}(x), \alpha_t = a \right] = e^{-\int_t^T c(s) ds} G(x, a, f(t); y, b, f(T)) \quad (25)$$

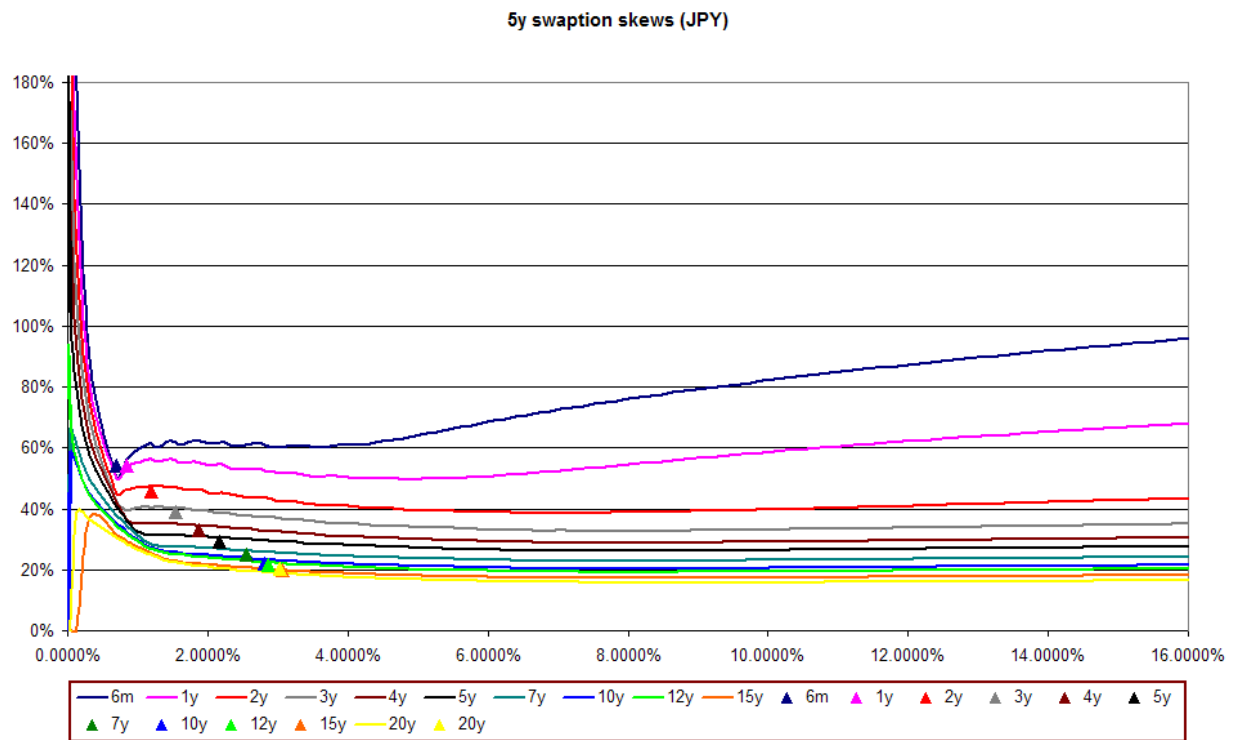


Figure 2: BlackScholes implied volatility smile for 5y JPY tenor swaption across different maturities and strikes

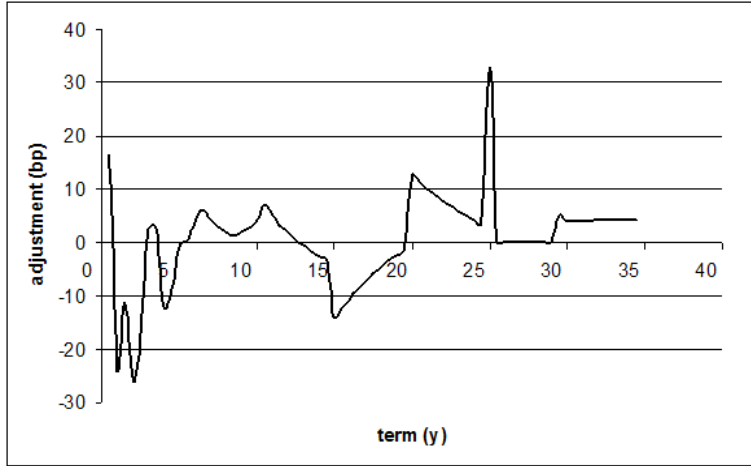


Figure 3: Time dependent short rate adjustment for a perfect fit to the yield curve

where G is the discounted probability kernel for the stationary process defined above.

Our choice in the working example is $f(t) = 1.095t + 0.008t^2$. The function $c(t)$ is then defined in such a way to match the term structure of forward swap rates. This adjustment is given in Fig. 3. As one can see from this picture, the short rate adjustment is typically less than 10 basis points in absolute value. This ensures that the probability of the modified short rate process \tilde{r}_t to attain negative values is very small. In contrast, in a typical implementation of the Hull-White model along similar lines, the short rate adjustment is typically of a few percent. The discrepancy is linked to the fact that the richer stochastic volatility model we construct is capable of explaining most of the salient features of the zero curve even with the constraint that the process be stationary.

An additional advantage of having a nearly stationary model is that the shapes of yield curves that one obtains depend on the short rate and the volatility state but are largely independent of time. Fig. 4 shows the yield curves corresponding to different initial volatility states and different starting values for the short rate. As the graphs indicate, yield curves are sensitive to the initial volatility state as they raise if initial volatilities raise. Moreover graphs show that curves invert for high values of the short rate. In our model, this behavior is consistent over all time frames except for corrections of the order of 10 basis points.

4.2 Local calibration step

In a second step (the local step), we ensure that the volatility smile of a subset of instruments is recovered. This step is performed systematically on the fly and

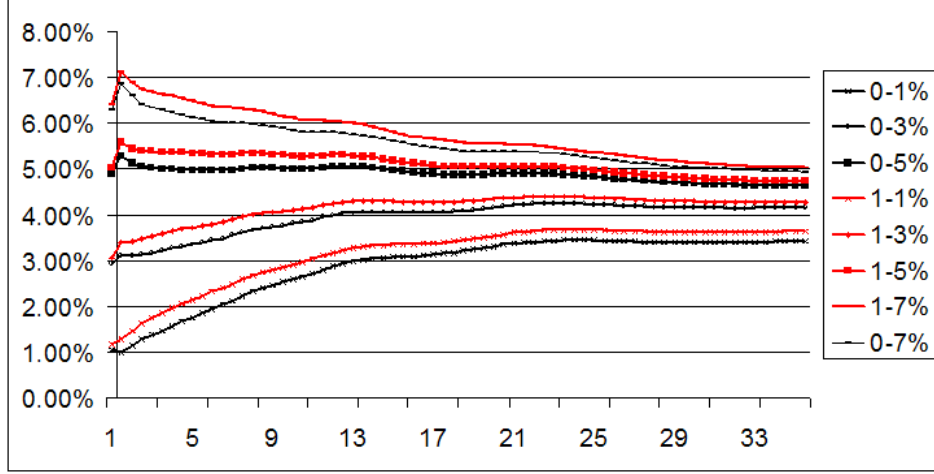


Figure 4: Yield curve shapes implied by the model as a function of the initial state

ensures that we accurately fit our model to the smile of the swaption contracts that represent the hedging vehicles of our pricing problem. Consider a sequence of times $t_i, i = 0, \dots, n$ where we want to locally calibrate our model. In order to solve for the desired functions $H_i(x, a, t_i) > 0$ necessary to fit the volatility cube one must ensure that the measure change process recovers the market value of the selected swaption contracts, where the strike varies across all lattice discretization points $k_j, j = 0, \dots, N$:

$$SO_{mkt}(k_j, t_i, \tau_i) = \sum_y G_H(x_0, a_0, t_0; y, b, t_i) V(y, b, k_j, t_i, \tau_i) \quad (26)$$

where:

- t_o is today's date
- x_0 is the spot state for the short rate lattice
- a_0 is the spot state for volatility regime lattice
- t_i is the expiry of the swaption contract
- τ_i is the term of the swaption contract
- k_j is the strike of the swaption contract
- $SO_{mkt}(k_j, t_i, \tau_i)$ is the interpolated market value of the swaption contract struck at k_j
- $V(y, b, k_j, t_i, \tau_i)$ is the payoff of a swaption contract

It should be noted that an interpolation on the input market value should be performed first, in order to obtain a target smile for the calibration of the measure change functions, which need to be evaluated at each lattice node point k_j . Alternatively, one could define the calibration problem for the available swaption contracts on the market and interpolate on the H function. However, this second approach would lead to an ill conditioned problem and it is therefore best avoided. The job of the sequence of functions $H_i(x, a, t_i)$ is to alter the discounted transition density in such a way to recover the implied swaption smile. However, as the initial discounted transition density has been obtained via a first global calibration step and it is therefore close to our desired solution, the time dependency introduced by the $H_i(x, a, t_i)$ is kept to a minimum. Proceeding forward in time, at time t_1 one can solve for $H_1(y, b, t_1)$, rearranging eq. (26) into a system of linear equations, which could be efficiently solved by means of numerical routines like *gels* of LAPACK:

$$\sum_{y,b} \left\{ \begin{array}{l} SO_{mkt}(k_j, t_1, \tau_1) e^{(t_1-t_0)\mathcal{L}}(x_0, a_0; y, b) - \\ V(y, b, k_j, t_1, \tau_1) e^{(t_1-t_0)\mathcal{L}}(x_0, a_0; y, b) \end{array} \right\} H_1(y, b, t_1) = 0 \quad (27)$$

The key observation which allows to obtain the above equation is that one can express the measure changed discounted transition probability kernel G_H as a function of H_1 only. In fact, from eq. (18) we have that:

$$H_1(x_0, a_0, t_0) = \sum_{y,b} e^{(t_1-t_0)\mathcal{L}}(x_0, a_0, t_0; y, b, t_1) H_1(y, b, t_1) \quad (28)$$

therefore substituting eq. (28) in:

$$\begin{aligned} SO_{mkt}(k_j, t_1, \tau_1) &= \sum_{y,b} G_H(x_0, a_0, t_0; y, b, t_1) V(y, b, k_j, t_1, \tau_1) \\ &= \sum_{y,b} \frac{1}{H_1(x_0, t_0)} G(x_0, a_0, t_0; y, b, t_1) H_1(y, b, t_1) V(y, b, k_j, t_1, \tau_1) \end{aligned}$$

and rearranging terms, we obtain:

$$\begin{aligned} &\sum_{y,b} e^{(t_1-t_0)\mathcal{L}}(x_0, a_0, t_0; y, b, t_1) H_1(y, b, t_1) SO_{mkt}(k_j, t_1, \tau_1) \\ &= \sum_{y,b} G(x_0, a_0, t_0; y, b, t_1) H_1(y, b, t_1) V(y, b, k_j, t_1, \tau_1) \\ &= \sum_{y,b} e^{(t_1-t_0)\mathcal{L}}(x_0, a_0, t_0; y, b, t_1) H_1(y, b, t_1) V(y, b, k_j, t_1, \tau_1) \end{aligned}$$

For each k_j , this is a linear equation in $H_1(y, b, t_1)$, from which one obtains eq. (27) by factorizing out $H_1(y, b, t_1)$. For each tenor τ , we have hence obtained a

system of $(N+1) = 71$ equations, one per node point on the lattice with $k_j = x_j$, $j = 0, \dots, N$, for the solution of a set of functions $H_1(y, b, t_1)$, $y = 0, \dots, N$; $b = 0, \dots, M$, with $(N+1) \cdot (M+1)$ unknowns. Depending on the application of interest, one can complete the specification of the calibration problem either by imposing an exact fit to the implied swaption smile of more than one tenor τ per expiry, or by matching a correlation structure between two rates. Also, some regularity conditions should be imposed to our system of equations in order to achieve a smooth solution. We suggest two possible approaches, which are based on the Tichonov regularization.

If one defines:

$$H_1(y, b, t_1) = 1 + \delta H_1(y, b, t_1)$$

then a smooth solution can be achieved by solving the system of equations:

$$\begin{cases} \sum_{y,b} h(k_j, t_1, \tau_1; y, b) \delta H_1(y, b, t_1) = - \sum_{y,b} h(k_j, t_1, \tau_1; y, b) \\ \min \sum_{y,b} (\delta H_1(y, b, t_1))^2 \end{cases}$$

where

$$h(k_j, t_1, \tau_i; y, b) = SO_{mkt}(k_j, t_1, \tau_i) e^{(t_1 - t_0)\mathcal{L}}(x_0, a_0; y, b) - V(y, b, k_j, t_1, \tau_i) e^{(t_1 - t_0)\mathcal{L}}(x_0, a_0; y, b)$$

Or, alternatively, one can define:

$$H_1(y, b, t_1) = 1 + \sum_{\tilde{b} \leq b} \sum_{\tilde{y} \leq y} \delta H_1'(\tilde{y}, \tilde{b}, t_1)$$

then a smooth solution can be achieved by solving the system of equations:

$$\begin{cases} \sum_{y,b} h(k_j, t_1, \tau_1; y, b) \sum_{\tilde{b} \leq b} \sum_{\tilde{y} \leq y} \delta H_1'(\tilde{y}, \tilde{b}, t_1) = - \sum_{y,b} h(k_j, t_1, \tau_1; y, b) \\ \min \sum_{y,b} (\delta H_1'(y, b, t_1))^2 \end{cases}$$

Proceeding forward in time, at time t_i , the calibration problem is still given by eq. (26), but in this case we need to compose the previous solution for all $H_j(y, b, t_j)$, $0 < j < i$. The measure changed discounted kernel, at time t_i , $t_0 < t_i$ is given by:

$$G_H(x_0, a_0, t_0; x_i, a_i, t_i) = \sum_{x_1, \dots, x_{j-1}} \sum_{a_1, \dots, a_{j-1}} \prod_{k=i+1}^j \frac{H_k(x_k, a_k, t_k)}{H_k(x_{k-1}, a_{k-1}, t_{k-1})} G(x_{k-1}, a_{k-1}, t_{k-1}; x_k, a_k, t_k) \quad (29)$$

which we can also write as:

$$\begin{aligned} & G_H(x_0, a_0, t_0; x_i, t_i) \quad (30) \\ = & G_H(x_0, a_0, t_0; x_{i-1}, a_{i-1}, t_{i-1}) \frac{1}{H_i(x_{i-1}, a_{i-1}, t_{i-1})} G(x_{i-1}, a_{i-1}, t_{i-1}; x_i, a_i, t_i) H_i(x_i, a_i, t_i) \end{aligned}$$

where the measure changed kernel $G_H(x_0, a_0, t_0; x_{i-1}, x_{i-1}, t_{i-1})$ is known from previous calibration iterations. With this modification, the calibration procedure for H_i proceeds in a similar manner to the one described for H_1 . The final solution for all functions H_i is obtained by performing a few iterations of the local calibration steps, so that to ensure that the both the term structure of interest rates, Black ATM volatility and the selected slices of the volatility cubes are correctly recovered.

The advantage of this hybrid calibration approach is that, as the starting point of the local calibration problem is already close to the desired solution, it is possible to keep to a minimum the amount of time dependency introduced in the model. It should be noted that one may be able to attempt a calibration to a subset of the volatility cube without introducing time dependent measure changes and increasing the number of regimes. This is also a possible and viable alternative, however it has the disadvantage of a performance penalty in the pricing, due to an increase in the size of the discounted transition probability matrix.

5 Pricing Bermuda swaptions and callable CMS swaps

We apply our model to the pricing of Bermuda swaption and callable CMS swaps. The latter are of particular interest, as the pricing of these contracts is very sensitive to the asymptotic behaviour of the implied volatility smiles of the model. Our numerical results refer to EUR denominated CMSs. We find that the model reproduces the correct asymptotic behaviour of the implied volatility smiles and gives rise to numerically stable exercise boundaries and hedge ratios.

Implied volatilities for European swaptions are given in Fig. 1 and 2, for EUR and JPY respectively. Here we graph extreme out of the money strikes of up to 15% for swaptions of varying maturities where the deliverable is a 5Y swap. One can notice that implied volatilities naturally flatten out at long maturities, a behavior consistent with what observed in the European swaption market and also inferred from the CMS market where such extreme strike levels are probed. Adopting this as a starting point, a perfect non-parametric fit can be achieved using the sequence of measure changes described above, without introducing a significant amount of explicit time dependence.

Exercise boundaries for 10Y Bermuda swaptions are given in Fig. 11 and Fig. 12. The first graph refers to payer swaptions and the second to receiver swaptions. The corresponding graphs for callable CMSs are given in Fig. 13 and Fig. 14. Notice that the exercise boundaries depend on the volatility state.

Sensitivities for Bermuda swaptions are given in Fig. 15 and 16. These sensitivities are computed by holding the volatility state variable fixed and are defined as the derivative of the price for a 10Y payer Bermuda swaption, struck at the rate reported on the abscissa, with respect to the rate of the 10Y swap. Numerically, these sensitivities are obtained by moving the spot rate and

computing the change in value of both the Bermuda swaption contract and the underlying hedging instrument.

Sensitivities of callable constant maturity swaps are given in Fig. 17, 18 and 19. The delta and gamma are computed similarly to what done for Bermuda swaptions, while the vega is calculated with respect to the 10Y into 5Y European swaption.

6 Pricing callable range accruals

In a typical range accrual note, the fixed coupon rate accrues contingent on the LIBOR rate being range bounded during the same coupon period. At payment date, the unit notional payment is calculated as follows:

$$p = k \frac{\sum_{i=1}^N \mathbf{1}_{\{f < L_i < c\}}}{N} \delta_{acc} \quad (31)$$

where $\mathbf{1}$ is the indicator function such that:

$$\mathbf{1}_{\{f < L_i < c\}} = \begin{cases} 1 & \text{if } f < L_i < c \\ 0 & \text{otherwise} \end{cases}$$

and:

- k : the range accrual fixed rate
- N : the total number of rate set observations in the coupon period
- f : the range floor
- c : the range cap
- L : the LIBOR rate
- δ_{acc} : the accrual fraction for the coupon period

Many variations exist on this theme. One refers to CMS range accruals when the coupon is contingent on a range bounded CMS rate, rather than a LIBOR rate; for CMS spreads range accruals, or twister notes, the coupon is contingent on a CMS spread instead. Also, k may be a fixed rate as well as a LIBOR rate index, a CMS rate or even a CMS spread index. Finally, if the range cap or floor level vary between accrual periods, the range accrual note is denoted as step-up. Callable range accrual structures are popular with investors, seeking to receive a higher coupon than the coupon they would receive if they were to enter into a plain vanilla fixed versus floating swap. Typically, in a range accrual swap, financial institutions pay a stream of structured coupon payments (the structured leg) to the investor in return of a stream of LIBOR

payments (the funding leg). The higher the probability that the reference index sets outside the range, the higher the nominal coupon rate. Therefore, the shape of the forward curve is crucial for the pricing of range accrual notes, and the range caps and floors are usually tailored to match the investor's view on interest rates. Also, as range accruals can be seen as a portfolio of digital options, volatility risk is an important component to be taken into account in the pricing and risk management of this product. As the investor is short the volatility of the reference index, the higher the implied volatility the higher is the yield enhancement that the investor can achieve. Typically, the higher the implied volatility compared to historical realized volatility of the range index, the more the investor is willing to take advantage of this structure.

In this paper we focus on the pricing of the most traditional range accrual note, where the coupon is defined as in eq. (31), however the mathematical framework we propose can be extended and applied to most trade typologies described above.

For easy of notation, from now on we refer to the state variable x as the multidimensional state variable driving the short rate process and the stochastic volatility process. In particular, in presence of stochastic volatility, the x variable $\in [0, \dots, \tilde{N} = (N \times M)]$ is a pair (ξ, a) , where $\xi \in [0, \dots, N]$ discretizes the state space of the short rate r , whereas $a \in [0, \dots, M]$ discretizes the state space of the stochastic volatility regime process. In the continuum time limit, we define the cumulative range accrual fraction, on an *Act/Act* basis, as:

$$I_t = \int_0^t \mathbf{1}_{\{f < L(x_s) < c\}} ds \quad (32)$$

The process I_t is the integral of a stochastic process, function of the short rate r_t . An important property of I_t is that the process does not depend on I_t itself, therefore it is translation invariant: if we were to shift the initial condition by a constant, the process I_t would only change by a constant shift. We approximate I_t in the discrete space as:

$$I_t = \alpha t + \beta m_t \quad (33)$$

where m_t is integer valued in $[0, 2L]$ and α and β are two constants.

Our objective is to find the joint transition probability between the short rate r and the cumulative range accrual fraction I_t . We seek to compute a lifted Markov generator $\tilde{\mathcal{L}}(x, m; y, n)$, a function of both the state space x , associated to the short rate process r and the regime process α , and the state space m , associated to the cumulative range accrual fraction process I_t . Having calculated $\tilde{\mathcal{L}}(x, m; y, n)$, the value at time t_i of the effective range accrual unit notional cash flow paid at time t_{i+1} is computed on the lattice as follows:

$$\sum_y \sum_n e^{(t_{i+1}-t_i)\tilde{\mathcal{L}}}(x, 0; y, n) \Delta I(n) G(x, t_i; y, t_{i+1}) k = C(x, t_i)$$

The generator $G(x, t_i; y, t_{i+1})$ is the discounted probability kernel of the short rate, discretized in the state variables (ξ, a) . $\Delta I(n)$ represents the expected range accrual fraction in the period (t_i, t_{i+1}) at each state n , therefore the term $\sum_n e^{(t_{i+1}-t_i)\tilde{\mathcal{L}}(x, 0; y, n)} \Delta I(n)$ evaluates the probability of accruing interest in the period (t_i, t_{i+1}) , for any given (x, y) state. Hence, the value of the range accrual coupon is given by the expected value of a fixed rate k multiplied by the expected range accrual fraction.

We define the lifted Markov generator:

$$\tilde{\mathcal{L}}(x, m; y, n) = \mathcal{L}_x(x; y) \delta_{mn} + \mathcal{L}_m(m; n | x) \delta_{xy}$$

We proceed with the construction of the Markov generator $\mathcal{L}_m(m; n | x)$. As the cumulative range accrual fraction is a quantity that grows smoothly in time, only up and down moves to contiguous sites are allowed in the model, and in particular no jumps are allowed. One can then note that, at each state n , a non zero transition probability can only be associated either to an up or a down move. Contingent on a state x , the matrix $\mathcal{L}_m(m; n | x)$ has non zero elements either on the diagonal and super- or sub-diagonal, respectively. Hence drift matching and probability conservation conditions are sufficient to uniquely determine the Markov generator $\mathcal{L}_m(m; n | x)$:

$$\sum_n \mathcal{L}_m(m; n | x) (m - n) = E[dm] \quad (34)$$

$$\sum_n \mathcal{L}_m(m; n | x) = 0 \quad (35)$$

Given the discretization choice for I , the drift of m can be computed as follows:

$$dI = \mathbf{1}_{\{f < L(x_s) < c\}} ds \quad (36)$$

$$E[dI] = \mathbf{1}_{\{f < L(x) < c\}} \quad (37)$$

$$dI = \alpha dt + \beta dm \quad (38)$$

$$E[dI] = \alpha + \beta E[dm] \quad (39)$$

$$E[dm] = \beta^{-1} (\mathbf{1}_{\{f < L(x) < c\}} - \alpha) \quad (40)$$

We then define the following two quantities, associated with an up and down move of the state variable m respectively:

$$a_+(x) = \beta^{-1} (\mathbf{1}_{\{f < L(x) < c\}} - \alpha)_+ \quad (41)$$

$$a_-(x) = \beta^{-1} (\mathbf{1}_{\{f < L(x) < c\}} - \alpha)_- \quad (42)$$

From eq. (34), eq. (35), eq. (41) and eq. (42) it follows that:

$$\begin{aligned}
\mathcal{L}_m(m; n \mid x) = & a_+(x)\delta_{n, m+1 \bmod [2L+1]}\delta_{x,y} \\
& + a_-(x)\delta_{n, m-1 \bmod [2L+1]}\delta_{x,y} \\
& - (a_+(x) + a_-(x))\delta_{n,m}\delta_{x,y}
\end{aligned} \tag{43}$$

We now take a closer look at how the matrix $\mathcal{L}_m(m; n \mid x)$ has been constructed and explain the meaning of its terms. The first element in eq. (43) is located on the super-diagonal and it is associated with the transition probability of an up move in the state variable m . The second element is on the sub-diagonal and it is related to the transition probability of a down move in the state m . The third element, on the diagonal, guarantees probability conservation and it is associated with an invariant move in m . By construction, either $a_+(x)$ or $a_-(x)$ can be non zero therefore, for each state x , the matrix $\mathcal{L}_m(m; n \mid x)$ is either sub- or super-diagonal. Special attention should be paid to boundary conditions. As the process I_t does not depend on I_t itself, the Markov generator $\mathcal{L}_m(m; n \mid x)$ is translation invariant, therefore the transition probability in the state m is only a function of the difference $(m - n)$. Also, the process I_t , continuous and unbounded in continuous time, is approximated with a discrete and bounded process on the lattice. However, by imposing periodic boundary conditions, we ensure that any transition in m going beyond the discrete space $[0, 2L]$ reappears on the opposite side of the boundary. Hence, the unbounded nature of the translation invariant process I_t is well captured by the periodic boundary conditions. Periodic boundary conditions are imposed by the term $\bmod[2L + 1]$ in the equation. Given the discrete space $[0, 2L]$, the parameters β and α control the centering and the width of the distribution of I_t in the space $[0, 2L]$. In practise, it is important to ensure that the distribution is properly centred so that near-zero transition probabilities would be found at the edges of the space. With enough discretization points, this can be achieved for a fixed β and α , across different maturities t .

The lifted Markov generator $\tilde{\mathcal{L}}(x, m; y, n)$ can therefore be constructed as follows:

$$\begin{aligned}
\tilde{\mathcal{L}}(x, m; y, n) = & \mathcal{L}_x(x; y)\delta_{m,n} \\
& + a_+(x)\delta_{n, m+1 \bmod [2L+1]}\delta_{x,y} \\
& + a_-(x)\delta_{n, m-1 \bmod [2L+1]}\delta_{x,y} \\
& - (a_+(x) + a_-(x))\delta_{n,m}\delta_{x,y}
\end{aligned}$$

Having computed $\tilde{\mathcal{L}}(x, m; y, n)$ one can evaluate the joint density:

$$Q(x, m, t; y, n, T) = e^{(T-t)\tilde{\mathcal{L}}}(x, m; y, n)$$

This requires the diagonalization of the matrix $\tilde{\mathcal{L}}(x, m; y, n)$, which is of dimension $(\tilde{N} \times (2L + 1), \tilde{N} \times (2L + 1))$. The naive diagonalization of this matrix would

be an onerous computational task, with unacceptable performance for practical applications. However, we proceed with a block diagonalization algorithm, which allows for an efficient and accurate solution of this problem.

6.1 Block diagonalization

The algorithm is based on the observation that, due to the translation invariant nature of $\mathcal{L}_m(m; n | x)$, it is possible to define a non singular linear transformation in terms of Fourier series such that, given a generic function $u(x, j)$, we have that:

$$(\mathcal{F}^{-1}\tilde{\mathcal{L}}\mathcal{F})u = (\oplus_{j=-L}^L \mathcal{L}_{p_j})u$$

where \mathcal{L}_{p_j} are a set of matrices of dimension $(\tilde{N} + 1, \tilde{N} + 1)$, function of (x, y) only and defined further below, and \mathcal{F} is the partial Fourier transform matrix defined as follows:

$$\mathcal{F} = \frac{1}{\sqrt{2L+1}} \begin{pmatrix} e^{ip_{-L}(L)} & 0 & 0 & \dots & \dots & e^{ip_{-L}(-L)} & 0 & 0 \\ 0 & \ddots & 0 & \dots & \dots & 0 & \ddots & 0 \\ 0 & 0 & e^{ip_{-L}(L)} & \dots & \dots & 0 & 0 & e^{ip_{-L}(-L)} \\ e^{ip_{-L+1}(L)} & 0 & 0 & \dots & \dots & e^{ip_{-L+1}(-L)} & 0 & 0 \\ 0 & \ddots & 0 & \dots & \dots & 0 & \ddots & 0 \\ 0 & 0 & e^{ip_{-L+1}(L)} & \dots & \dots & 0 & 0 & e^{ip_{-L+1}(-L)} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ e^{ip_L(L)} & 0 & 0 & \dots & \dots & e^{ip_L(-L)} & 0 & 0 \\ 0 & \ddots & 0 & \dots & \dots & 0 & \ddots & 0 \\ 0 & 0 & e^{ip_L(L)} & \dots & \dots & 0 & 0 & e^{ip_L(-L)} \end{pmatrix}$$

The matrix \mathcal{F} is composed of $((2L+1) \times (2L+1))$ diagonal blocks of dimension $(\tilde{N} + 1, \tilde{N} + 1)$ each with constant elements on the diagonal. If we define each block:

$$\mathcal{F}_{j_1, j_2} = \begin{pmatrix} e^{ip_{j_1}(j_2)} & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & e^{ip_{j_1}(j_2)} \end{pmatrix}$$

with $j_1 = -L, \dots, L$ and $j_2 = L, \dots, -L$, the matrix \mathcal{F} can be described with the compact notation:

$$\mathcal{F} = \frac{1}{\sqrt{2L+1}} \begin{pmatrix} \mathcal{F}_{-L, L} & \mathcal{F}_{-L, L-1} & \dots & \mathcal{F}_{-L, -L} \\ \mathcal{F}_{-L+1, L} & \mathcal{F}_{-L+1, L-1} & \dots & \mathcal{F}_{-L+1, -L} \\ \vdots & \vdots & \vdots & \vdots \\ \mathcal{F}_{L, L} & \mathcal{F}_{L, L-1} & \dots & \mathcal{F}_{L, -L} \end{pmatrix}$$

The joint density function can be computed by calculating first the transformed kernel of $\tilde{\mathcal{L}}$, for which a block diagonalization is possible, and then by applying the inverse transformation:

$$\begin{aligned} Q &= e^{(T-t)\tilde{\mathcal{L}}} = \mathcal{F}^{-1} \left(\mathcal{F} e^{(T-t)\tilde{\mathcal{L}}} \mathcal{F}^{-1} \right) \mathcal{F} = \mathcal{F}^{-1} \left(e^{(T-t)(\mathcal{F}\tilde{\mathcal{L}}\mathcal{F}^{-1})} \right) \mathcal{F} \\ &= \mathcal{F}^{-1} \left(e^{(T-t)\oplus_{j=-L}^L \mathcal{L}_{p_j}} \right) \mathcal{F} = \mathcal{F}^{-1} \left(\oplus_{j=-L}^L e^{(T-t)\mathcal{L}_{p_j}} \right) \mathcal{F} \end{aligned}$$

where, for simplicity of notation, we have dropped the state and time dependencies.

We now show that the matrix $\mathcal{F}\tilde{\mathcal{L}}\mathcal{F}^{-1}$ can be computed as the direct sum of a set of matrices $\mathcal{L}_{p_j}(x; y)$, for $j = -L, \dots, L$, of dimension $(\tilde{N} + 1, \tilde{N} + 1)$. This result allows to solve the diagonalization problem of $\tilde{\mathcal{L}}(x, m; y, n)$ with a series of smaller matrix diagonalizations, which can be efficiently performed by means of numerical routines like *geev* of LAPACK. First we observe that, as by construction the Markov generator $\mathcal{L}_m(m; n | x)$ is translation invariant, its eigenfunctions are of the form e^{ipm} , and if $u(x)$ is a generic function, we have that:

$$\begin{aligned} \sum_y \sum_n \tilde{\mathcal{L}}(x, m; y, n) e^{ipm} u(y) &= e^{ipm} \left\{ \begin{array}{c} \sum_y \mathcal{L}_x(x; y) u(y) \\ + e^{ip} a_+(x) u(x) + e^{-ip} a_+(x) u(x) \\ - (a_+(x) + a_-(x)) u(x) \end{array} \right\} \\ &= e^{ipm} \sum_y \mathcal{L}_p(x; y) u(y) \end{aligned}$$

where $\mathcal{L}_p(x; y) \stackrel{def}{=} \mathcal{L}_x(x; y) + [e^{ip} a_+(x) + e^{-ip} a_+(x) - (a_+(x) + a_-(x))] \delta_{x,y}$. Also, given a generic function $u(x, m)$, one can calculate the Fourier transform $\hat{u}(x, j)$:

$$\hat{u}(x, j) = (\mathcal{F}u)(x, j) = \frac{1}{\sqrt{2L+1}} \sum_{m=-L}^L u(x, m) e^{-ip_j m}$$

In the discrete space, the index p spans the values $p_j = \frac{2\pi j}{2L+1}$, $j = -L, \dots, L$, whereas $k = 0, \dots, \tilde{N}$ and the vector $\hat{u}(x, j)$ has the form:

$$\widehat{u}(x, j) = \begin{pmatrix} \widehat{u}(0, -L) \\ \widehat{u}(1, -L) \\ \vdots \\ \widehat{u}(\widetilde{N}, -L) \\ \widehat{u}(0, -L + 1) \\ \widehat{u}(1, -L + 1) \\ \vdots \\ \widehat{u}(\widetilde{N}, -L + 1) \\ \vdots \\ \vdots \\ \widehat{u}(0, L) \\ \vdots \\ \widehat{u}(\widetilde{N}, L) \end{pmatrix}$$

Applying the operator $\mathcal{F}\widetilde{\mathcal{L}}\mathcal{F}^{-1}$ to $\widehat{u}(x, j)$ we obtain:

$$\begin{aligned} (\mathcal{F}\widetilde{\mathcal{L}}\mathcal{F}^{-1}\widehat{u})(x, k) &= \frac{1}{2L+1} \sum_{m=-L}^L e^{-ip_k m} \sum_{j=-L}^L \sum_y \sum_n \widetilde{\mathcal{L}}(x, m; y, n) \widehat{u}(y, j) e^{ip_j n} \\ &= \frac{1}{2L+1} \sum_{m=-L}^L e^{-ip_k m} \sum_{j=-L}^L e^{ip_j m} \sum_y \mathcal{L}_p(x; y) \widehat{u}(y, j) \\ &= \frac{1}{2L+1} \sum_{m=-L}^L \sum_{j=-L}^L e^{i(p_j - p_k)m} \sum_y \mathcal{L}_p(x; y) \widehat{u}(y, j) \end{aligned} \quad (44)$$

The key observation is that:

$$\frac{1}{2L+1} \sum_{j=-L}^L e^{i(p_j - p_k)m} = \delta_{j,k}$$

hence eq. (44) reduces to:

$$(\mathcal{F}\widetilde{\mathcal{L}}\mathcal{F}^{-1}\widehat{u})(x, k) = \sum_y \mathcal{L}_p(x; y) \widehat{u}(y, j)$$

which can be written, in a more compact notation:

$$(\mathcal{F}\widetilde{\mathcal{L}}\mathcal{F}^{-1})\widehat{u} = \left(\bigoplus_{j=-L}^L \mathcal{L}_{p_j} \right) \widehat{u}$$

If we denote with \mathcal{L}_{p_j} each matrix $\mathcal{L}_{p_j}(x, y)$, the matrix $\mathcal{F}\widetilde{\mathcal{L}}\mathcal{F}^{-1}$, diagonal in blocks, can be described as follows:

$$\mathcal{F}\tilde{\mathcal{L}}\mathcal{F}^{-1} = \begin{pmatrix} \mathcal{L}_{p-L} & 0 & \cdots & \cdots & 0 \\ 0 & \mathcal{L}_{p-L+1} & 0 & \cdots & \vdots \\ \vdots & 0 & \ddots & 0 & 0 \\ \vdots & \vdots & 0 & \ddots & 0 \\ 0 & 0 & 0 & 0 & \mathcal{L}_{p_L} \end{pmatrix}$$

The diagonalization of $\mathcal{F}\tilde{\mathcal{L}}\mathcal{F}^{-1}$ can be performed in blocks, through a series of diagonalization of smaller dimension matrices $\mathcal{L}_{p_j}(x; y)$, which can be efficiently computed by means of numerical routines like *geev* of LAPACK. If we define the eigenvalue problem for the matrix \mathcal{L}_{p_j} :

$$\mathcal{L}_{p_j} = S_{p_j} \Lambda_{p_j} S_{p_j}^{-1}$$

where S_{p_j} are the right eigenfunctions of \mathcal{L}_{p_j} and the Λ_{p_j} the eigenvalues, then joint density \mathcal{Q} can be computed as a direct sum as follows:

$$Q = \mathcal{F}^{-1} \left(\bigoplus_{j=-L}^L S_{p_j} e^{(T-t)\Lambda_{p_j}} S_{p_j}^{-1} \right) \mathcal{F}$$

Or, in matrix form:

$$Q(x, m; y, n) = e^{t\tilde{\mathcal{L}}}(x, m; y, n) = \frac{1}{2L+1} \sum_{j=-L}^L e^{ip_j(m-n)} \sum_{k=1}^N e^{(T-t)\lambda_{p_j k}} u_{p_j k}(x) v_{p_j k}(y)$$

Fig. (5) show the joint density function calculated on a discretized space $[0, 2L]$ for an example range accrual note, where the LIBOR rate is range bounded between zero and five per cent.

7 Pricing TARNs

Target Redemption Notes (TARN) are interest rate swaps where one counterparty pays a structured coupon and typically receives LIBOR plus a spread. Additionally, all remaining cash flows are cancelled (the trade “burns out”) when the cumulative structured coupon exceeds a given barrier level, namely the target redemption level. Financial institutions usually pay the structured coupon which, due to the target redemption feature, can provide yield enhancement to the receiving counterparty. In fact, a TARN trade embeds a digital option on the underlying swap, which is sold by the receiving counterparty to the paying counterparty. This value is reflected in the higher structured coupon compared to a plain vanilla swap.

Conceptually, the TARN feature can be applied to any type of structured coupon. In its simplest form, the structured coupon of the TARN trade is a

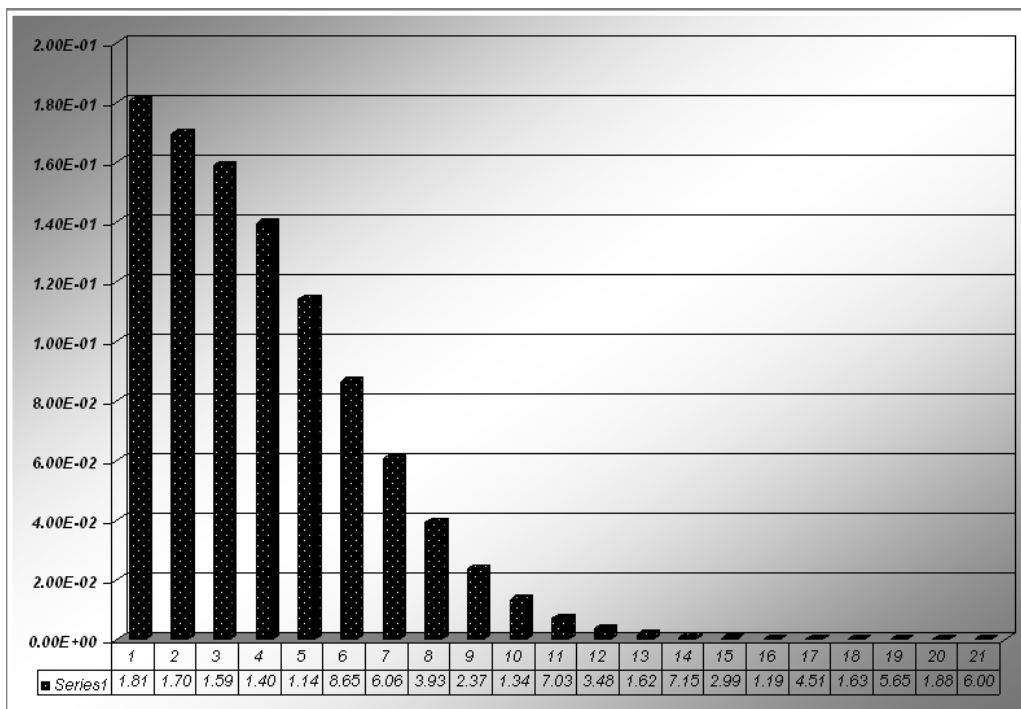


Figure 5: Joint transition probability between the short rate process and the range accrual coupon fraction. The coupon fraction is incremented when the Libor rate resets between 0% and 5%.

LIBOR rate with a cap and a floor or an inverse LIBOR rate, i.e. a fixed coupon minus the LIBOR rate, again with a cap and a floor. Other popular, and more exotic, TARN trades pay structured coupons with periodic features, or linked to the spread between two CMS rates or with range accrual payoffs. In the remainder of this section, we focus on the pricing of inverse capfloater-type TARN trade, as this example is well suited for the presentation of our methodology. The same construction could be applied to other types of structured coupons, with appropriate modifications.

The pricing of TARN trades presents two main challenges: handling the path dependent nature of the payoff and capturing the higher moments on the term structure of the yield curve. In fact, at any given pay date, the payoff depends on the previously accumulated coupon and hence it depends on the past history. Also, typical TARN trades are written with a fairly short intrinsic burn out time, therefore the synthetic barrier option on the trade is sensitive to the level of the long rates, whereas the structured coupon is sensitive to the level of short rates. Hence, it is important to capture tilts in the yield curve as well as parallel shifts.

The stochastic volatility short rate model presented above is rich enough to capture yield curve tilts scenario. We show a construction that allows to lift the Markov generator in order to efficiently take into account the path dependent nature of the payoff.

For the case of an inverse capfloater payoff, the cumulative coupon paid at time t is:

$$I_t = \sum_{t_i < t} C_{t_i} = \sum_{t_i < t} \min(\max(k_i - L_{t_i}(r_{t_i}), f), c)$$

where C_{t_i} is the structured coupon, k_i is a fixed coupon, L_{t_i} is the LIBOR rate reset at time t_i , c is the cap level and f is the floor level. In this section we adopt the same notation we have used for the callable range accrual case: we refer to the state variable x as the multidimensional state variable driving the short rate process and the volatility regimes process. In particular, in presence of stochastic volatility, the x variable $\in [0, \dots, \tilde{N} = (N \times M)]$ is a pair (ξ, a) , where $\xi \in [0, \dots, N]$ discretizes the state space of the short rate r , whereas $a \in [0, \dots, M]$ discretizes the state space of the volatility regimes process.

The objective is to find the joint distribution of the variables x and a , which in turn allows to compute the joint distribution of the short rate r_t and the cumulative coupon I_t . We discretize the random variable I_t of the cumulative coupon as follows:

$$I_t = \alpha t + \beta m_t$$

where m_t is integer valued, and α and β are two constants. We introduce the lifted probability kernel for the time period (t_i, t_{i+1}) , with t_i being a generic payment date:

$$U(x, m, t_i; y, n, t_{i+1}) = U_x(x, t_i; y, t_{i+1})U_m^{(x,y)}(m, t_i; n, t_{i+1})$$

where $U_x(x, t_i; y, t_{i+1})$ is the unconditional probability kernel of x and $U_m^{(x,y)}(m, t_i; n, t_{i+1})$ is the probability kernel of the state variable m conditional on a fixed (x, y) pair.

As $U_m^{(x,y)}$ is a probability kernel, it must satisfy to:

$$\begin{aligned} U_m^{(x,y)}(m, t_i; n, t_{i+1}) &\geq 0 \\ \sum_n U_m^{(x,y)}(m, t_i; n, t_{i+1}) &= 1 \end{aligned}$$

The cumulative coupon process is translation invariant, as it does not depend on the past history of itself but only on the difference in the state space m . Conditional on a state (x, y) , between times (t_i, t_{i+1}) , the incremental coupon $\Delta I(x, y; t_i, t_{i+1})$ is known and can be computed as follows:

$$\begin{aligned} \Delta I(x, y; t_i, t_{i+1}) &= I_{t_{i+1}} - I_{t_i} \\ &= \alpha(t_{i+1} - t_i) + \beta(m_{t_{i+1}} - m_{t_i}) \\ &= \alpha(t_{i+1} - t_i) + \beta(n - m) \end{aligned}$$

We discretize the state space $m \in [0, \dots, 2L]$ and the expected value of n will in general take values in between the discretized node points, $E[n] \in [n_-(x, y, m), n_+(x, y, m)]$:

$$E[n] = \left(m + \frac{1}{\alpha} (\Delta I(x, t_i; y, t_{i+1}) - \beta(t_{i+1} - t_i)) \right) \bmod 2L + 1$$

with

$$\begin{aligned} n_-(x, y, m) &= \left[m + \frac{1}{\alpha} [\Delta I(x, t_i; y, t_{i+1}) - \beta(t_{i+1} - t_i)] \right]_0 \bmod 2L + 1 \\ n_+(x, y, m) &= n_-(x, y, m) + 1 \bmod 2L + 1 \end{aligned}$$

where $n_-(x, y, m)$ is the integer part of the quantity in square brackets, according to the notation:

$$[a]_0 = \sup \{n \in \mathbf{Z} : n \leq a\}$$

The transition probability, conditional on (x, y) , to move from state m at time t_i to state n at time t_{i+1} , can be expressed as a weighted average of the transition probabilities to move from state m to the two states, n_- and n_+ , adjacent to n in the discretized space:

$$U_m^{(x,y)}(m, t_i; n, t_{i+1}) = a_-(x, t_i; y, t_{i+1})\delta_{n, n_-(x, y, m)} + a_+(x, t_i; y, t_{i+1})\delta_{n, n_+(x, y, m)}$$

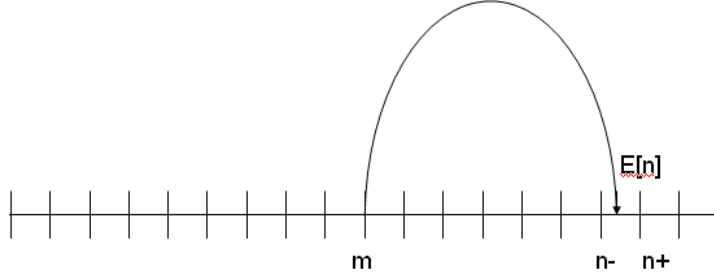


Figure 6: The expected value of n will in general fall in between two discretization points n_+ and n_-

The weights can be computed by imposing the probability conservation condition and matching the first moment of the structured coupon:

$$\begin{aligned} a_- + a_+ &= 1 \\ a_- [(n_-(x, y, m) - m)] + a_+ [(n_+(x, y, m) - m)] &= \frac{1}{\alpha} [\Delta I(x, t_i; y, t_{i+1}) - \beta(t_{i+1} - t_i)] \end{aligned}$$

For a generic time period (t_i, t_j) , with $0 \leq i < j$, the lifted probability kernel associated with payment at time t_i can be computed by taking the $(j - i)$ th power of the lifted probability kernel $U(x, m, t_{j-1}; y, n, t_j)$ computed for the coupon period (t_{j-1}, t_j) :

$$U(x, m, t_i; y, n, t_j) = (U(x, m, t_{j-1}; y, n, t_j))^{j-i}$$

There are two possible strategies one can adopt to perform this operation:

1. Diagonalize $U(x, m, t_{j-1}; y, n, t_j)$, take the power of the eigenvalue matrix and multiply by left and right eigenvectors
2. Multiply the matrix $U(x, m, t_{j-1}; y, n, t_j)$ by itself $(j - i)$ times

The diagonalization of a matrix involves \tilde{N}^3 operations, whereas a matrix multiplication involves $\tilde{N}^{2+\varepsilon}$ operations, with $\varepsilon \ll 1$. Hence, the second approach is preferable from a performance point of view when

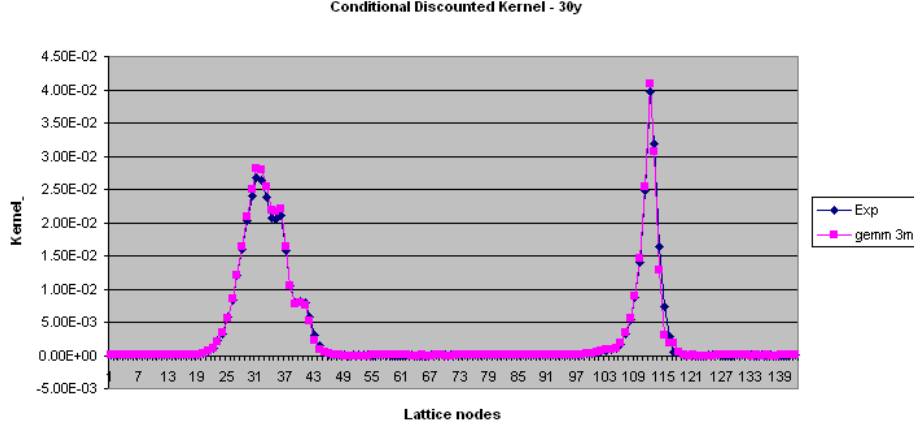


Figure 7: Conditional discounted transition probabilities obtained through straight exponentiation, on one side, and 120 matrix multiplications, on another side

$$j - i \leq \tilde{N}(1 - \varepsilon)$$

which is true in most cases for the lattice dimensions we use. Also, one can verify that repetitive matrix multiplications do not introduce unwanted errors. In Fig. (7) we show the discounted transitional probabilities, conditional on the spot state, for a kernel with two regimes, obtained through straight exponentiation and 120 matrix multiplication. The two methods yield very similar results.

The matrix $U(x, m, t_i; y, n, t_{i+1})$, however, is very large and both the diagonalization and the matrix multiplication algorithm would require too much computational power for practical applications. We note that, as the conditional matrix $U_m^{(x,y)}(m, t_i; n, t_{i+1})$ is translation invariant, we can apply a partial Fourier linear transformation to $U(x, m, t_i; y, n, t_{i+1})$, similarly to what we have done for the range accrual case, in order to reduce the matrix in blocks. With the same notation as in the range accrual case:

$$\hat{U}(j, m, t_i; k, n, t_{i+1}) = (\mathcal{F}U\mathcal{F}^{-1})_{i+1}$$

where $\hat{U}(j, m, t_i; k, n, t_{i+1})$ is a block matrix, which can be efficiently multiplied or diagonalized. The kernel can therefore be efficiently computed as follows:

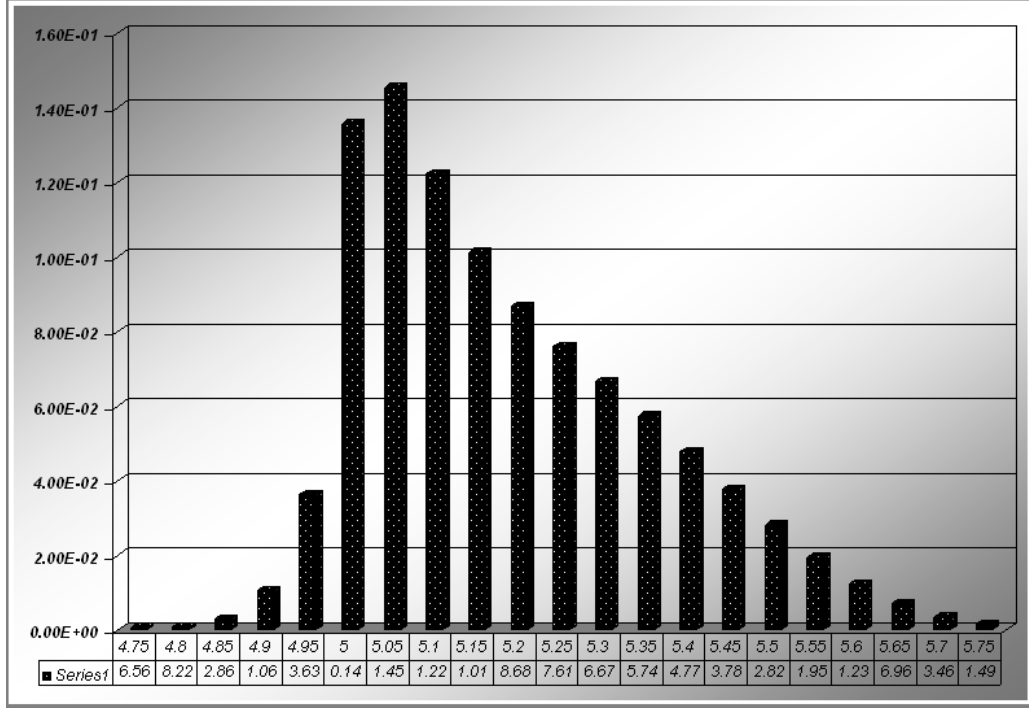


Figure 8: Joint transition probability between the short rate process and the total coupon process

$$\begin{aligned}
 U &= \mathcal{F}^{-1} \left(\prod_i (\mathcal{F}U\mathcal{F}^{-1})_i \right) \mathcal{F} \\
 &= \mathcal{F}^{-1} (\mathcal{F}S\Lambda^{j-i}S^{-1}\mathcal{F}^{-1}) \mathcal{F}
 \end{aligned}$$

Fig. (8) shows an example joint density function for a TARN payoff with cumulative coupon barrier at 4%. Fig. (9) and (10) show the effect of a misspecification of β and α respectively.

8 Conclusions

We present a stochastic volatility term structure model built upon a specification of a short rate process, which combines local volatility, stochastic volatility and jumps. The richness of the model allows to keep the degree of time variability of model parameters to a bare minimum, and obtain a nearly stationary dynamic,

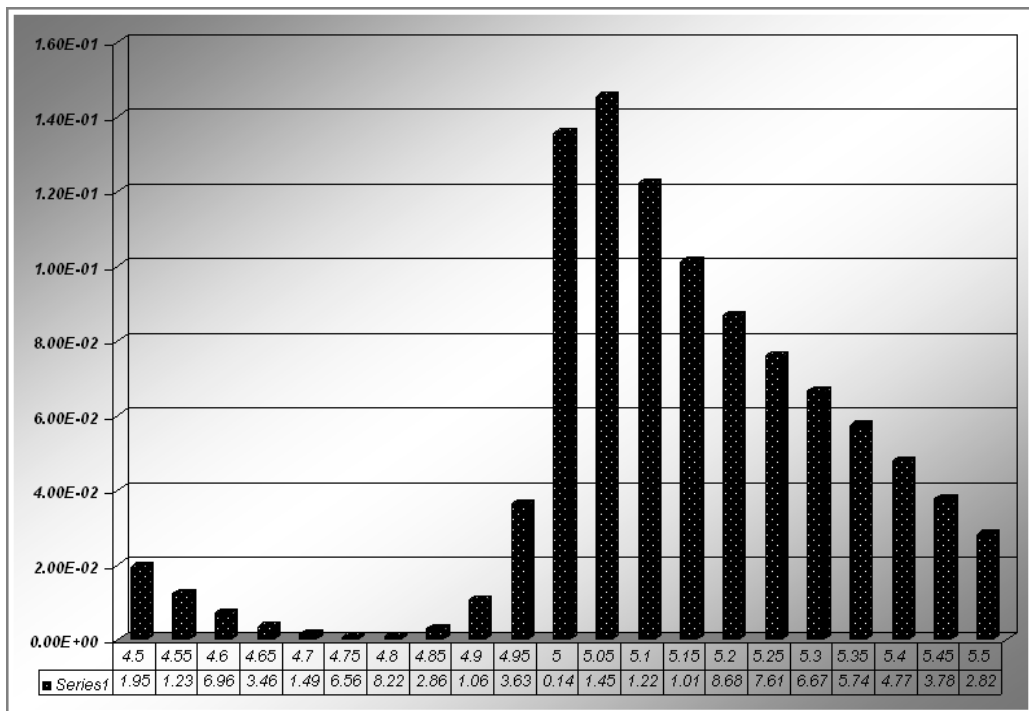


Figure 9: Effect of misspecification of β in the calculation of the joint transition probability between the short rate process and the total coupon process

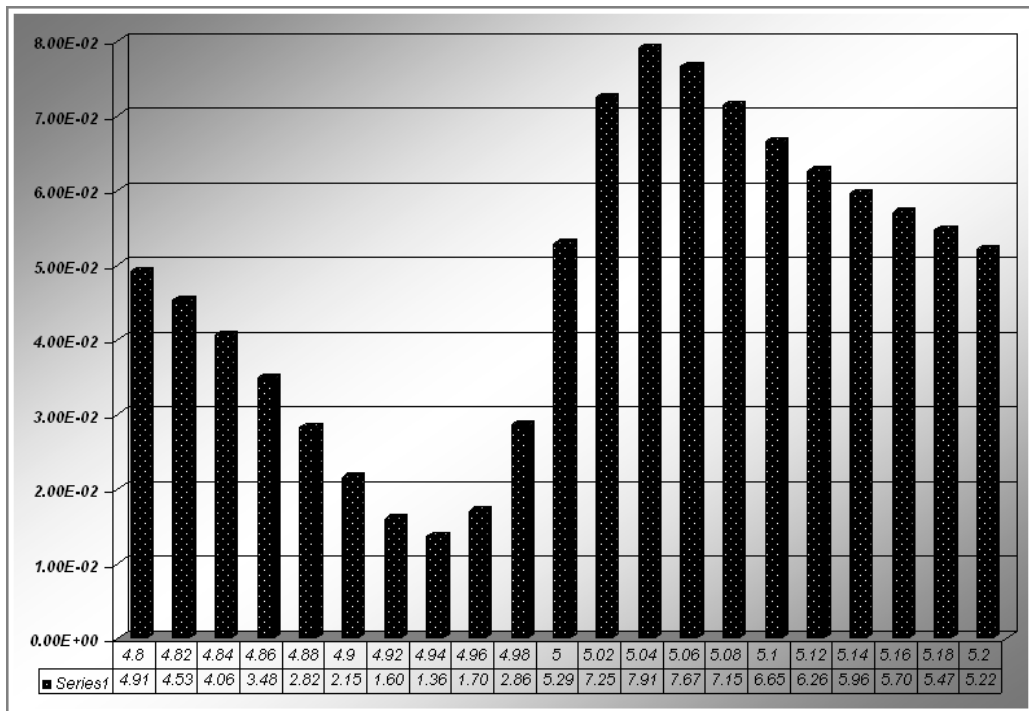


Figure 10: Effect of misspecification of α in the calculation of the joint transition probability between the short rate process and the total coupon process

while achieving also a perfect fit to the term structure of interest rates and a region of the volatility cube corresponding to a selected family of hedging vehicles.

The solution methodology is based on a new type of continuous time lattices, which allow for a numerically stable and quite efficient technique to price fixed income exotics and evaluate hedge ratios. The model provides a consistent framework for pricing European and Bermuda options, as well as callable CMS swaps for which we present an implementation example. We also show the extension of the model to the pricing of path dependent options, with translation invariant state variables. We present the application of the model to the pricing of callable range accruals and Target Redemption Notes. We show how the model can be applied to these payoffs by means of a lifting operation on the Markov generator and describe a partial factorization algorithm that allows for fast block diagonalization of large matrices. The mathematical technique here described is quite flexible and is perhaps the main contribution of this paper.

References

- Albanese, Claudio & Alexey Kuznetsov (2003), Discretization schemes for subordinated processes. *Mathematical Finance*. To appear.
- Albanese, Claudio & Alexey Kuznetsov (2005), ‘Affine lattice models’, *International Journal of Theoretical and Applied Finance* **8(2)**, 223–238.
- Albanese, Claudio & Manlio Trovato (2005), A stochastic volatility model for bermuda swaption and callable cms swaps. Working paper.
- Andersen, Leif B.G. & Jesper Andreasen (2000), ‘Volatility skews and extensions of the libor market model’, *Applied Mathematical Finance* **7**, 1–32.
- Andersen, Leif B.G. & Jesper Andreasen (2002), ‘Volatile volatilities’, *Risk* **15(12)**.
- Andersen, Leif B.G. & Rupert Brotherton-Ratcliffe (2001), Extended libor market models with stochastic volatility. Working paper.
- Black, Fisher & Piotr Karasinski (1991), ‘Bond and option pricing when short rates are lognormal’, *Financial Analysts Journal*. pp. 52–59.
- Brace, A., D. Gatarek & M. Musiela (1996), ‘The market model of interest rate dynamics’, *Mathematical Finance* **7**, 127–154.
- Cox, J.C., J.E. Ingersoll & S.A. Ross (1985), ‘A theory of the term structure of interest rates’, *Econometrica*. **53**, 385–407.
- Geman, Helyette, Nicole El Karoui & J.C. Rochet (1995), ‘Changes of numeraire, changes of probability measure and option pricing’, *Journal of Applied Probability* **32**, 443–458.

- Glasserman, Paul & Nicolas Merener (2001), Cap and swaption approximations in libor market models with jumps. SSRN Working paper.
- Glasserman, Paul & S. Kou (1999), The term structure of simple forward rates with jump risk. Columbia Working paper.
- Hagan, Patrick S., Deep Kumar, Andrew S. Lesniewski & Diana E. Woodward (2002), ‘Managing smile risk’, *Wilmott Magazine* **11**.
- Hu, Zhengyun (2005), The state tree. Working paper, SSRN.
- Hull, J. (2005), *Options, Futures and Other Derivatives*, Prentice Hall.
- Hull, J. & A. White (1993), ‘One-factor interest rate models and the valuation of interest rate derivative securities’, *Journal of Financial and Quantitative Analysis* **28**, 235–54.
- Hunt, Phil, Joanne Kennedy & Anthon Pelsser (2000), ‘Markov-functional interest rate models’, *Finance and Stochastics* **4**(4).
- Jamshidian, Farshid (1993), ‘Option and futures evaluation with deterministic volatilities’, *Mathematical Finance* **3**, 149–159.
- Jamshidian, Farshid (1997), ‘Libor and swap market models and measures’, *Finance and Stochastics* pp. 293–330.
- Joshi, Mark & Riccardo Rebonato (2003), A stochastic-volatility, displaced-diffusion extension of the libor market model. QUARC Working paper.
- Piterbarg, Vladimir (2003), A stochastic volatility forward libor market model with a term structure of volatility smiles. SSRN Working paper.
- Threfethen, Lloyd N. (1999), ‘Computation of pseudospectra’, *Acta Numerica, Cambridge University Press* pp. 247–295.
- Vasicek, O.A. (1977), ‘An equilibrium characterization of the term structure’, *Journal of Financial Economics* **5**, 177–88.

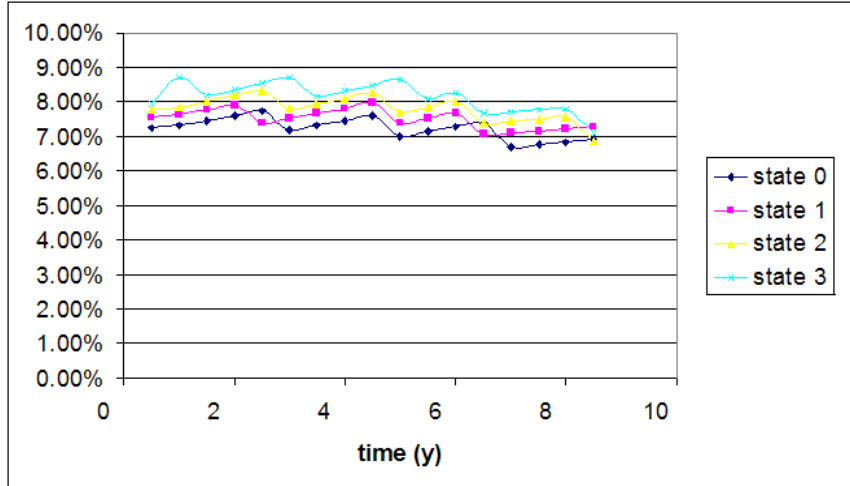


Figure 11: Exercise boundaries for a 10y Bermuda payer swaption with quarterly exercise schedule

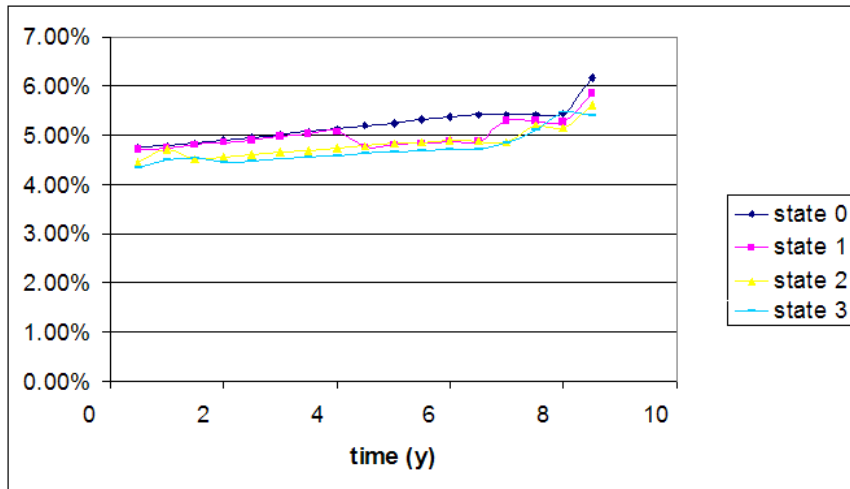


Figure 12: Exercise boundaries for a 10y Bermuda receiver swaption with quarterly exercise schedule

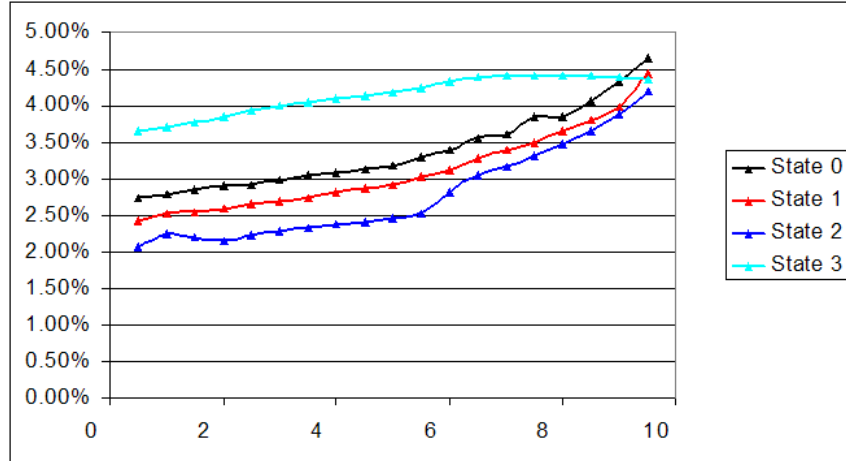


Figure 13: Exercise boundaries for a 5y CMS receiver callable swap with 10Y maturity and quarterly exercise schedule

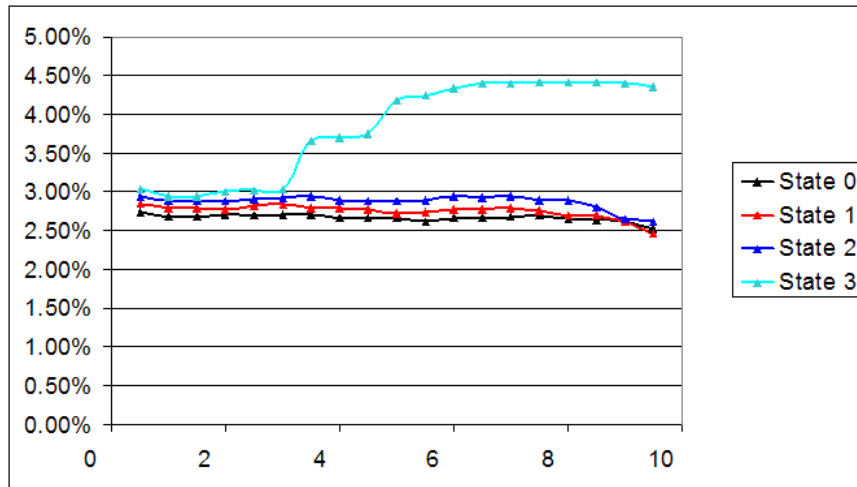


Figure 14: Exercise boundaries for a 5y CMS payer callable swap with 10Y maturity and quarterly exercise schedule

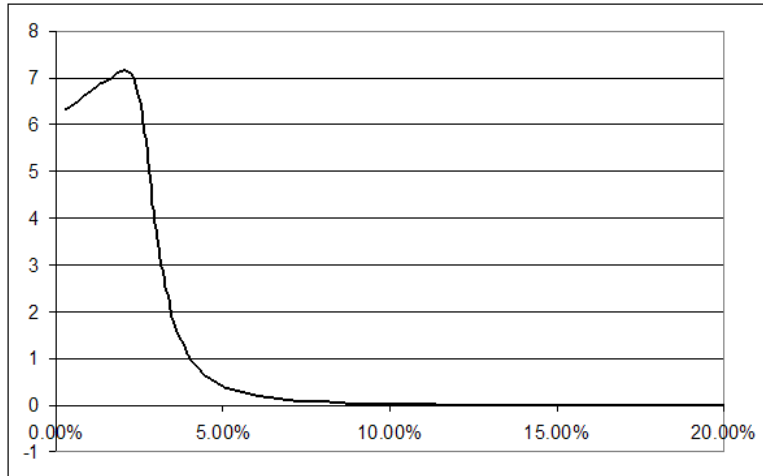


Figure 15: Delta sensitivity of a 10Y Bermudan swaption with quarterly exercise schedule with respect to the 10Y swap rate

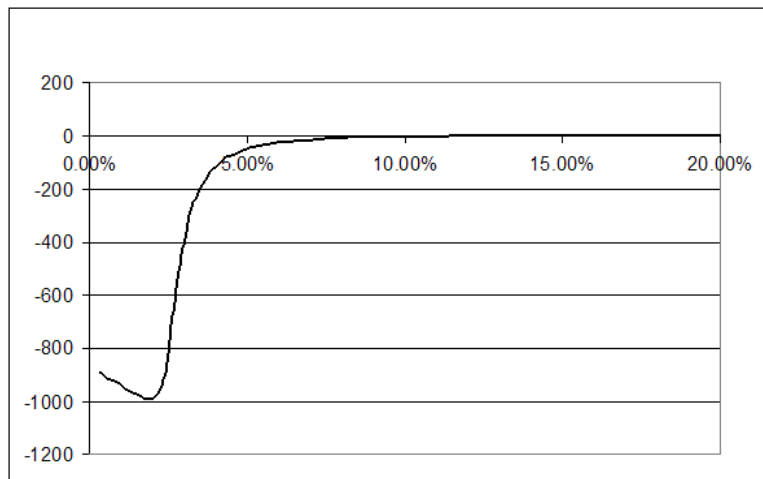


Figure 16: Gamma sensitivity of a 10Y Bermudan swaption with quarterly exercise schedule with respect to the 10Y swap rate

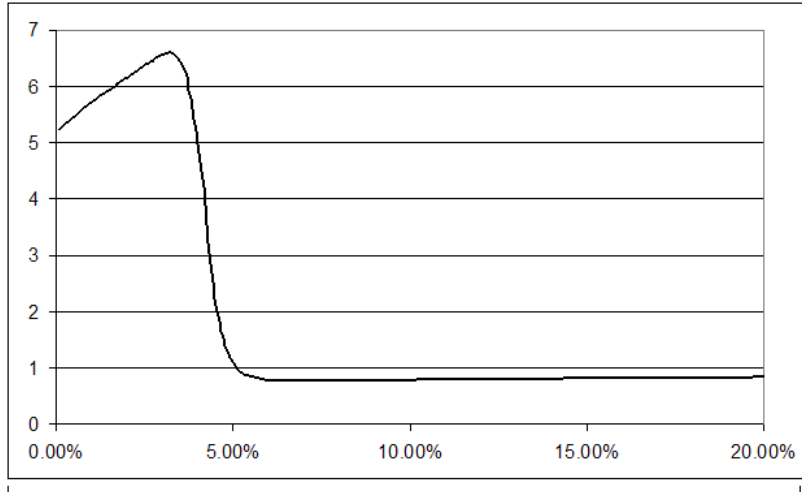


Figure 17: Delta sensitivity of a 5Y CMS callable swap with 10Y maturity and quarterly exercise schedule with respect to the 10Y swap rate

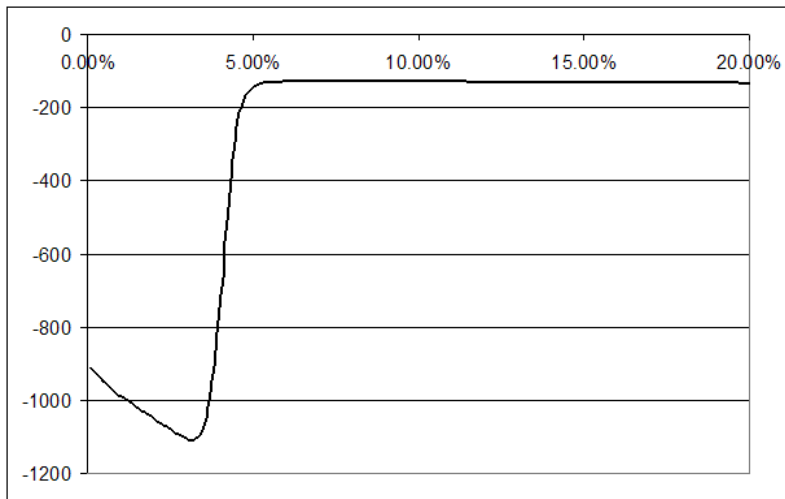


Figure 18: Gamma sensitivity of a 5Y CMS callable swap with 10Y maturity and quarterly exercise schedule with respect to the 10Y swap rate

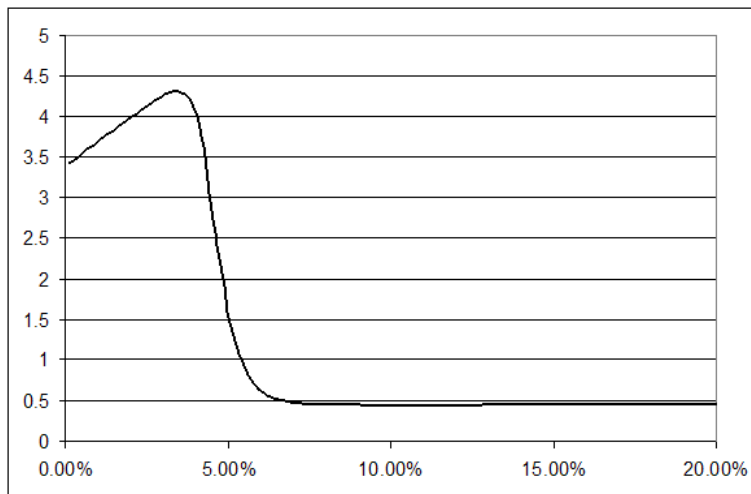


Figure 19: Vega sensitivity of a 5Y CMS callable swap with 10Y maturity and quarterly exercise schedule with respect to the 10Y into 5Y European swaption