Numerical Methods for the Markov Functional Model

Simon Johnson
Financial Engineering
Commerzbank Corporates and Markets
60 Gracechurch Street, London EC3V 0HR

Abstract: Some numerical methods for efficient implementation of the 1- and 2-factor Markov Functional models of interest rate derivatives are proposed. These methods allow a sufficiently rapid implementation of the standard calibration method, that joint calibration to caplets and swaptions becomes possible within reasonable CPU time. Prices for Bermudan swaptions generated within the Markov Functional model are found to be very close to market consensus prices. Bermudans are therefore a good example of a product ideally suited to this model.

1 Introduction

The Libor Market Model of Brace Gatarek and Musiela (BGM) (1997) is the market standard model for pricing and hedging exotic interest rate derivatives. Its advantages include model parameters which are easy to interpret in terms of financial variables, ability to define realistic correlation dynamics, and the ability to price essentially any callable Libor exotics by means of Monte Carlo valuation.

There are two main difficulties in practical implementation of the BGM Model. Firstly the drift is strongly state-dependent and cannot be reduced to a low-dimensional Markovian form. Whilst simulations with long time-steps can still be performed using a suitable differencing scheme (Hunter et al., 2001; Joshi, 2005), this does mean that Monte Carlo simulation is the only practical method for pricing. One must work hard to achieve acceptable convergence, particularly when computing hedge ratios of callable Libor exotics Piterbarg (2004). Secondly, although calibration to market prices of caplets is of course trivial, there is a great deal of debate surrounding the best way to perform global calibration of a BGM model to market prices of all at-the-money swaptions. The choice of calibration method becomes even richer when extensions to BGM, including displaced diffusion, local volatility or stochastic volatility are considered Rebonato (2004). However when the model is used in a production environment, it is by no means simple to ensure that a small change in market quotes give rise to a correspondingly small change in model parameters and hence stable hedges are obtained.

A second type of market model is the swap market model Galluccio et al. (2004). The advantages and disadvantages of this model are closely related to those of the BGM model. For instance, in the case of a co-terminal swap market model, calibration to a set of coterminal swaptions is trivial, but achieving good numerical convergence of greeks and stable global calibration are again very challenging.

A third type of Market Model is the Markov Functional model proposed by Hunt, Kennedy and Pelsser (2000), Hunt and Kennedy (2005). The derivation of this model has the unusual starting point of proposing a state variable which follows a simple driftless Brownian motion with time-dependent volatility, which is typically 1- or 2-dimensional.

$$dx_i = \sigma_i(t)dW_i \tag{1}$$

$$\langle dW_i dW_j \rangle = \rho_{ij} dt \tag{2}$$

The calibration of the model corresponds to the choice of the numeraire as some function $N(\vec{x}, t)$ of this state variable. The standard choice of numeraire, and the one which is used in this paper, is the zero-coupon bond with maturity T_N. The model can therefore be used to price any (non-path dependent) Libor exotics using a backwards finite difference solver. Of course strongly path-dependent products can also be priced using Monte Carlo methods. Whilst global calibration of the 1-factor Markov Functional model is not possible, local calibration to the complete volatility smile of one swap or libor rate per maturity is possible. Because the model allows calibration to the smile, it is particularly suitable for highly smile-sensitive products such as high strike payers Bermudan swaptions.

The purpose of this paper is to present a range of numerical methods which can be used in the calibration of the Markov Functional model. These improve the speed of the standard calibration method described by Hunt, Kennedy and Pelsser, but more importantly they make extensions of the calibration, including joint calibration to caplets and swaptions, more practical.

2 The Markov Functional Model

The standard calibration of the 1-factor Markov Functional model, as described by Hunt, Kennedy and Pelsser, relies on a Jamshidian-type trick. To be more specific, using the market prices of European swaptions we can find the market price of a digital swaption which pays an annuity if the swap rate is greater than some strike:

$$D_{market}(K) = N_0 \mathbb{E}\left[\frac{A_t \Theta(SR_t - K)}{N_t}\Big|\mathcal{F}_t\right] \tag{3}$$

where $\Theta(x)$ represents the Heaviside (step) function. In this calibration method, caplets can be considered to be a single period swaption. Moreover, having calibrated the numeraire at canonical dates $t_N, t_{N-1}, \ldots, t_{i+1}$ we can price in the model a product which pays an annuity if the state variable x is greater than some critical value $x_\star$

$$D_{model}(x_\star) = N_0 \mathbb{E}\left[\frac{A_t(x_t)\Theta(x_t - x_\star)}{N(x_t, t)}\Big|\mathcal{F}_t\right] \tag{4}$$

We make the ansatz that the swap rate is a monotonic function of the state variable and solve for the swap rate as a function of state variable

$$SR(x) = D^{-1}_{market}(D_{model}(x)) \tag{5}$$

So given a continuum of European swaption prices, we can extract the swap rate, and hence the numeraire, as a function of the state variable.

A similar backwards-rolling calibration method can be used in the 2-factor Markov Functional model. As suggested in Hunt and Kennedy (2005) we make the ansatz that the swap rate of interest is a monotonic function of a 1-d projection of the 2-d state variables $z(x, y)$. Hunt and Kennedy choose the projection function using a low-dimensional, Markovian approximation to a BGM model which they call the 'pre-model'. We propose another choice, which is motivated by an approximation to a Hull-White model and which additionally allows the efficient numerical integration methods in the calibration described in section 5.

Consider the two-factor Hull-White short rate model:

$$\begin{aligned} df_t^1 &= -\lambda_1 f_t^1 + \sigma_1 dW_t^1 \\ df_t^2 &= -\lambda_2 f_t^2 + \sigma_2 dW_t^2 \\ r_t &= f_t^1 + f_t^2 + \phi(t) \end{aligned} \tag{6}$$

where $\phi(t)$ is a deterministic term. By making the substitution $x_t = \exp(\lambda_1 t)f_t^1$, $y_t = \exp(\lambda_2 t)f_t^2$ we obtain driving factors in the form of (1).

$$\begin{aligned} dx_t &= \sigma_1 \exp(\lambda_1 t)dW_t^1 \\ dy_t &= \sigma_2 \exp(\lambda_2 t)dW_t^2 \\ r_t &= \exp(-\lambda_1 t)x_t + \exp(-\lambda_2 t)y_t + \phi(t) \end{aligned} \tag{7}$$

In the two-factor Hull-White model, then, the short rate is a monotonic function of the bilinear projection function

$$z(x_t, y_t) = \exp(-\lambda_1 t)x_t + \exp(-\lambda_2 t)y_t \tag{8}$$

or more generally

$$z(x_t, y_t) = x_t/\sqrt{\mathrm{var}(x_t)} + y_t/\sqrt{\mathrm{var}(y_t)} \tag{9}$$

In the case of a 2-factor Markov Functional model, we therefore expect realistic behaviour by assuming that the Libor rate or swap rate is a monotonic function of (9). Moreover, as we shall find in section 5, this simple bilinear form allows very efficient performance of the integrals used in calibration.

3 Extrapolation of Market Swaption Prices

The calibration method described in the previous section is simple and numerically well-behaved. However, market data for European swaptions with extremely high or extremely low strikes might not exist or may be arbitrageable for several reasons:

- Market data for at-the-money swaptions and for swaptions of extreme strikes may not come from the same source or may not have been updated at the same time, and hence may be inconsistent. For example quotes for at-the-money swaptions are very liquid, and will be typically updated on a continuous basis by market data providers. However swaptions with extreme strikes will be much less liquid and may be updated only occasionally.
- In many banks, smile surfaces are parametrised by means of a stochastic volatility model such as the SABR model Hagan et al. (2002). Hagan gives an asymptotic expansion for the price of a European option in this model which is extremely accurate for strikes close to the forward. However in the wings of the distribution, the approximation derived by Hagan for the implied volatility of a European option can give negative probability densities and hence arbitrageable market prices. Of course the problem here is not the use of the SABR model itself, but in pushing an asymptotic expansion beyond its region of applicability. Other interpolation methods have been suggested, such as Gatheral's SVI (stochastic volatility inspired) parameterisation Gatheral (2004), which could also be used to mitigate this problem.

Typically the degree of arbitrage from these causes will be much too small to exploit once transaction costs are taken into account. However it prevents the functional inversion in (5), and as a result, calibration to digital swaptions can only be performed for a finite range of strikes. Outside of this range, some extrapolation method should be used, however it is vitally important that the extrapolation method chosen for digital options must preserve the price of a european option. In other words, if the market price of an at-the-money European swaption (payers or receivers) is E_{market} then the extrapolation must give the correct value for a call option

$$E_{market} = \int_{-\infty}^{F} (df - D_{market}(K))dK \tag{10}$$

and a put option

$$E_{market} = \int_{F}^{\infty} D_{market}(K)dK \tag{11}$$

where df is the discount factor on the payment date.

If we only have trustworthy, non-arbitrageable prices for European swaptions with strikes in the range k_{min} to k_{max}, we must ensure that our extrapolated market prices satisfy the following:

$$\begin{aligned}\int_{-\infty}^{k_{min}} (df - D_{market}(K)) &= E_{market} - \int_{k_{min}}^{F} (df - D_{market}(K))dK \\ \int_{k_{max}}^{\infty} D_{market}(K)dK &= E_{market} - \int_{F}^{k_{max}} D_{market}(K)dK\end{aligned} \tag{12}$$

Otherwise, we may have a Markov Functional model which reprices digital swaptions perfectly within the range of interest, but which fails significantly for at-the-money European swaptions. One way to achieve this is by constructing an interpolating object and performing the integals on the RHS of (12) using a standard method such as adaptive Gauss-Lobatto integration. We can choose any extrapolation method satisfying (12), but if we choose an extrapolation type such as exponential, then the integrals on the LHS can be performed analytically and the task is particularly easy.

4 Interpolation of the Numeraire Function

The main requirements on the numeraire are that

- It must be positive, for all choices of market data, state variable $\vec{x}$ and time t.
- If the numeraire is a zero-coupon bond with maturity T_N, the expectation of $1/N(x,t)$ must equal the discount factor ratio $DF(t)/DF(T_N)$ for all t so that the model will match market prices for zero-coupon bonds stripped from a yield curve

During the model calibration, the numeraire function is only constructed on a set of canonical dates. Typically these canonical dates may be quarterly, semiannual, or annual, with the choice made according to the period of Libor underlying the caplets, or else the fixed period of the swaptions used for calibration. In general we will need to price products with payment dates different from the canonical dates, so some form of interpolation of the numeraire function is necessary.

For a 1-factor Markov Functional model, $N(x,t)$ is stored as a 2-d interpolating object. The first requirement is generally achieved by choosing an interpolation method which preserves positivity. The second requirement is achieved by defining a normalised 'numeraire function'

$$f_i(x/\sqrt{v(t)}) = \frac{DF(T_N)}{DF(t_i)} \frac{1}{N(x,t_i)} \tag{13}$$

where

$$v(t) = \int_0^t \sigma^2(s)ds \tag{14}$$

is the variance of the underlying Wiener process. The functions $f_i(x)$ can use any x-interpolation method, although we have found that log-linear interpolation and flat extrapolation gives particularly good results. For t-interpolation, we can interpolate between the different time-slices $f_i(x)$ linearly in variance $v(t)$. This interpolation method is chosen because it preserves normalisation between the interpolation dates. For example, if the numeraire function is correctly normalised at canonical dates:

$$\begin{aligned}\frac{DF(T_N)}{DF(t_1)}\int_{-\infty}^{\infty}\frac{dx}{\sqrt{2\pi v_1}}\frac{\exp(-x^2/2/v_1)}{N(x,t_1)} &= \int_{-\infty}^{\infty} f_1(\tilde{x})\exp(-\tilde{x}^2/2)\frac{d\tilde{x}}{\sqrt{2\pi}} = 1 \\ \frac{DF(T_N)}{DF(t_2)}\int_{-\infty}^{\infty}\frac{dx}{\sqrt{2\pi v_2}}\frac{\exp(-x^2/2/v_2)}{N(x,t_2)} &= \int_{-\infty}^{\infty} f_2(\tilde{x})\exp(-\tilde{x}^2/2)\frac{d\tilde{x}}{\sqrt{2\pi}} = 1\end{aligned} \tag{15}$$

then the interpolated numeraire function will also be correctly normalised

$$\begin{aligned}\frac{DF(T_N)}{DF(t)}\int_{-\infty}^{\infty}\frac{dx}{\sqrt{2\pi v}}\frac{\exp(-x^2/2/v)}{N(x,t)} &= \int_{-\infty}^{\infty} f(\tilde{x})\exp(-\tilde{x}^2/2)\frac{d\tilde{x}}{\sqrt{2\pi}}\\ &= \int_{-\infty}^{\infty}\left(\frac{v-v_1}{v_2-v_1}f_1(\tilde{x})+\frac{v_2-v}{v_2-v_1}f_2(\tilde{x})\right)\\ &\quad\times\exp(-\tilde{x}^2/2)\frac{d\tilde{x}}{\sqrt{2\pi}}\\ &= \frac{v-v_1}{v_2-v_1}+\frac{v_2-v}{v_2-v_1}=1\end{aligned} \tag{16}$$

and all zero-coupon bonds will be exactly repriced.

5 Numerical Methods for Expectation Integrals

The calibration of the 1-factor model requires calculation of two integrals, the first of which is a convolution:

$$\begin{aligned}I_1(x_t,t) &= \mathbb{E}[f(x_T,T)|\mathcal{F}_t]\\ &= \int_{-\infty}^{\infty}\frac{f(x_T,T)dx}{\sqrt{2\pi(v(T)-v(t))}}\exp\left(-\frac{(x_T-x_t)^2}{2(v(T)-v(t))}\right)\end{aligned} \tag{17}$$

Although the convolution form suggests the use of Fourier methods, our experience is that it can be difficult to prevent edge-effects from diffusing into the solution domain. Instead we perform these integrals using straightforward Gauss-Hermite integration Press et al. (1992). The second integral used in the 1-factor model is the conditional expectation:

$$\begin{aligned}I_2 &= \mathbb{E}[f(x_T,T)\Theta(x_T-x_\star)|\mathcal{F}_0]\\ &= \int_{x_\star}^{\infty}\frac{f(x_T,T)dx}{\sqrt{2\pi v(T)}}\exp\left(-\frac{x_T^2}{2v(T)}\right)\end{aligned} \tag{18}$$

where again the function $f(x_T,T)$ is smooth. In the 1-factor case, the cost of these integrals is not too great, so that the choice of numerical methods used is not too critical. However some improvement in performance is achieved by calculating the conditional expectation integral by a change of variable

$$\begin{aligned}y &= N(x/\sqrt{v(t)})\\ I_2 &= \int_{y_\star}^{1} f(x(y),T)dy\end{aligned} \tag{19}$$

where $N(x)$ is the cumulative normal distribution. The integral can now easily be performed using Gauss-Legendre integration. Note that the method of an inverse cumulative normal transform followed by Gauss-Legendre quadarature only gives good results if the integrand meets some regularity conditions. In particular, for some functions $f(x)$, the polynomial fit implicit in the Gauss-Legendre method is very far from the original function. However in this case, particularly when using flat extrapolation of the numeraire function, the algorithm works well.

In the 2-factor Markov Functional model, the choice of integration method becomes much more important. The convolution integral:

$$\begin{aligned}I_3(\vec{x}_t,t) &= \mathbb{E}[f(\vec{x}_T,T)|\mathcal{F}_t]\\ &= \int_{-\infty}^{\infty}\prod_i\frac{dx_i}{\sqrt{2\pi}}\exp\left(-\frac{1}{2}(\vec{x}_T-\vec{x}_t).\mathbf{v}^{-1}.(\vec{x}_T-\vec{x}_t)\right)\frac{f(x_T,T)}{\sqrt{\det\mathbf{v}}}\end{aligned} \tag{20}$$

is an integral of a smooth function over a Gaussian kernel. One possible approach to integrals of this type is to diagonalise the matrix $\mathbf{v}$, and to use 1-d Gauss-Hermite integration for each eigenvector direction. This is known as 'repeated quadrature'. With Gauss-Hermite quadrature of order n for each direction, this will require n^2 evaluations of the integrand. Another approach, which we have found gives improved speed with no loss of accuracy, is to use a *cubature* formula. This technique is not widely known in Quantitative Finance, and the reader is referred to Cools (1997) for an introduction to the subject. Briefly, instead of the 1-d set of orthogonal polynomials which are used to construct Gauss-Hermite points and weights Press et al. (1992), cubature techniques start with a basis set of 2-d orthogonal polynomials. Again the aim is to find a set of weights and points such that the integral is approximated by

$$\int K(\vec{x})f(\vec{x})d\vec{x}\approx\sum_{i=1}^{N}w_i f(\vec{x}_i) \tag{21}$$

Finding the optimal set of points $\vec{x}_i$ at which the integrand should be evaluated is a great deal more difficult than in the 1-dimensional case and relies on finding points which are simultaneously zeros of as many as possible of the basis functions. This uses advanced group-theoretic techniques and efficient formulae are only known for a few values of N. In the case of a 2-dimensional integration with a Gaussian kernel, for example, a number of efficient formulae of degree 5 are known (i.e. integrating exactly all bivariate polynomials of order 5 and less) [3] for values of N between 7 and 12. A number of formulae of degree 9 are known, with values of N between 18 and 25. A number of higher order schemes are known, but those described give remarkably quick and robust results for the cost of only a small number of function evaluations.

The integral I_4 is defined by

$$\begin{aligned}I_4 &= \mathbb{E}[f(\vec{x}_T,T)\Theta(z(x,y)-z_\star)|\mathcal{F}_0]\\ &= \int_{-\infty}^{\infty}\prod_i\frac{dx_i}{\sqrt{2\pi}}\Theta(z(x,y)-z_\star)\exp\left(-\frac{1}{2}\vec{x}_T^T.\mathbf{v}^{-1}.\vec{x}_T\right)\frac{f(\vec{x}_T,T)}{\sqrt{\det\mathbf{v}}}\end{aligned} \tag{22}$$

and since we chose a bilinear projection function (9)

$$z(x,y)=c_x x+c_y y=r(x\cos\theta+y\sin\theta) \tag{23}$$

we can rotate the coordinates of the integral so that the digital condition affects only one direction.

$$\vec{x}' = \mathbf{U}\vec{x} \tag{24}$$

$$\mathbf{U} = \begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix}$$

so that

$$I_4 = \int_{x_\star}^{\infty} dx' \int_{-\infty}^{\infty} dy' \exp\left(-\frac{1}{2}\vec{x'}_T^T.\mathbf{U}.\mathbf{v}^{-1}.\mathbf{U}^{\mathbf{T}}.\vec{x'}_T\right)\frac{f(\mathbf{U}^T\vec{x'}_T, T)}{2\pi\sqrt{\det \mathbf{v}}} \tag{25}$$

We can then perform the y'—integral using Gauss-Hermite, transform the x'—coordinate using (19) and then perform the x'—integral using Gauss-Legendre. This method gives excellent accuracy because it deals explicitly with the discontinuity in the integrand.

6 Calibration of the Volatility Function and Typical Results

When implemented in C++, the numerical methods described above allow an efficient implementation of the standard calibration of the Markov Functional model. For a simple example with 20 canonical dates (10 yrs, semiannual), calibration of the 1-factor model on a grid of size 50 takes around 0.06 seconds on a 2.8GHz Intel Pentium IV Xeon. For the 2-factor model, calibration of the numeraire on a 30*30 grid takes about 0.28 seconds.

In each case, the calibration is sufficiently fast that we can perform some or all of this calibration sweep iteratively, whilst solving for the volatility $\sigma(t)$ of the Markovian process (1). As described in Hunt et al. (2000), the term-structure of volatility effectively controls the integrated correlation of different forward rates, whilst preserving a link to a continuous hedging argument. To give a simple example, we have used a Levenberg-Marquardt algorithm Press et al. (1992), Nielsen (1999) to calibrate the piecewise constant volatility of the Markovian process, enabling calibration to the whole (smile-consistent) distribution of caplets, whilst additionally calibrating to a set of at-the-money coterminal swaptions. Each call to the error function of the nonlinear solver does the following steps:

- set the piecewise constant volatility of the Markovian process to the chosen value;
- perform the Markov-functional calibration sweep;
- using the newly calibrated model, compute the prices of the set of at-the-money coterminal swaptions;
- the error function returned is the difference between model and market prices for these coterminal swaptions;

Such a calibration type might be suitable for a product such as a callable range accrual on Libor. Digital caplets with any strike are correctly repriced, so we can be confident that the underlying range accrual, which can be decomposed into a sum of digital caplets, is correctly priced. The optionality will be priced correctly, at least in the limit that the barrier levels are widely spaced, because we can reprice all of the underlying European options into which we might exercise.

Typical results from the calibration were gathered using market data for the Euro interest rates market, observed on 9th August 2005. For the 1-factor Markov Functional model, the numeraire was stored on a 80 point grid with 20 canonical dates (10 yrs semiannual). Calibration of the numeraire function and of $\sigma(t)$ took 35 seconds using the algorithm described above, and 3 outer iterations of the Levenberg-Marquardt algorithm were used. As expected, the complete smile for the caplets was reproduced perfectly (see figure 1). The ATM coterminal swaptions were also repriced, although again as expected, out-of-the-money the smile was not matched perfectly (figure 2). Figure 3 shows repricing errors for ATM options across the swaption matrix. The calibrated term-structure of $\sigma(t)$ is shown in figure 4.

The remaining calibration error had two sources. Firstly, numerical convergence error which could be reduced by increasing the number of points on the numeraire grid and the number of Levenberg-Marquardt iterations. Secondly, a small error was introduced by calibrating to *canonical* caplets (with no fixing lag) and then repricing real caplets with a 2 business day fixing lag.

Similar results were obtained using a 2-factor model. For example, with a 30*30 grid and 2 outer iterations of the Levenberg-Marquardt solver, global calibration to caplets and coterminal swaptions took 99 seconds. Because of the somewhat coarser numerical grid, the maximum calibration error for the ATM caplets or coterminal swaptions in the calibration set increased from 0.1% illustrated in figure 3, to 0.21%. Of course the use of a 2-factor

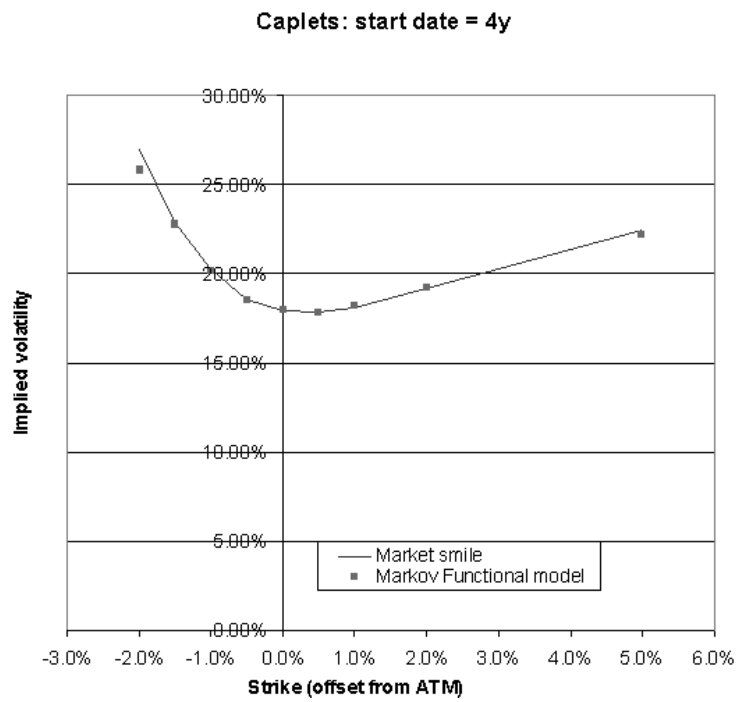

Figure 1: Calibration results for caplets with a start date of 4y. As these are being used for the calibration of the numeraire function, they are all exactly repriced.

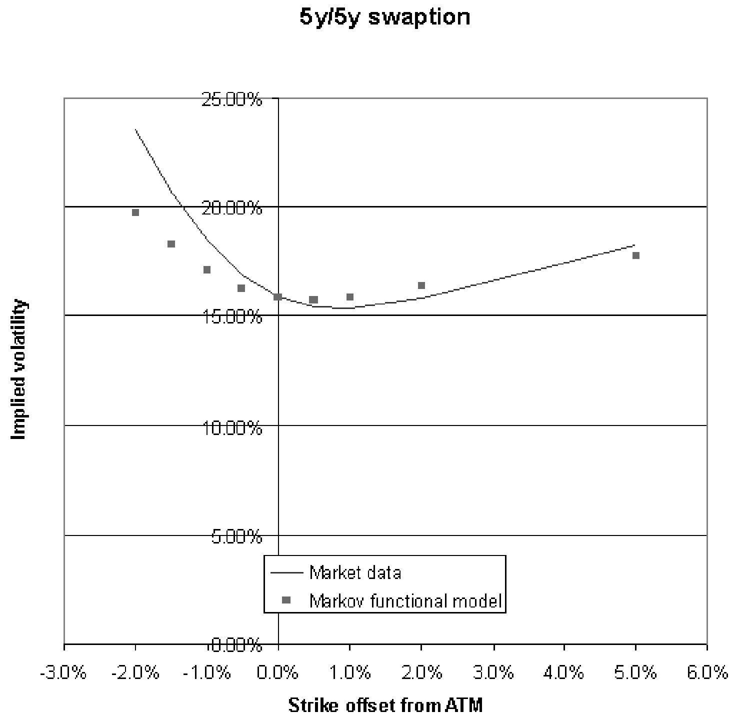

Figure 2: Calibration results for 5y/5y swaptions. As these are being used for the calibration of the volatility $\sigma(t)$, the ATM coterminal swaptions are exactly repriced.

ATM calibration error	6m	1y	2y	3y	4y	5y	6y	7y	8y	9y
1y	**-0.08%**	-0.31%	-0.18%	-0.23%	-0.29%	-0.12%	0.09%	0.16%	0.18%	**-0.01%**
2y	**-0.09%**	-0.03%	-0.47%	-0.25%	0.05%	0.18%	0.28%	0.21%	**0.01%**	
3y	**0.02%**	0.03%	-0.28%	0.09%	0.27%	0.28%	0.23%	**0.02%**		
4y	**0.10%**	0.11%	0.15%	0.34%	0.36%	0.31%	**-0.01%**			
5y	**0.03%**	0.35%	0.50%	0.50%	0.33%	**-0.01%**				
6y	**0.05%**	0.38%	0.49%	0.41%	**-0.01%**					
7y	**0.01%**	0.32%	0.33%	**-0.03%**						
8y	**0.06%**	0.28%	**-0.09%**							
9y	**0.00%**	**-0.08%**								

Figure 3: Calibration results for the ATM swaption matrix. The table shows the difference between market implied volatilities and implied volatilities from the calibrated model. The options which were part of the calibration set are shown in boldface.

model would now enable more realistic modelling of products which depend on CMS spread. As a simple example, figure 5 shows a scatter plot of Monte Carlo paths, illustrating the decorrelation between the 5yr swap rate and the 1yr swap rate, observed in 5 years.

As a final illustration of the use of the Markov Functional model, the 1-factor model was used to price Bermudan swaptions. The prices of co-terminal swaptions were used to calibrate the numeraire function, and the prices of ATM caplets were used to calibrate the volatility function. The results obtained were compared with prices from a Hull-White model calibrated to at-the-money coterminal swaptions, and also with market consensus

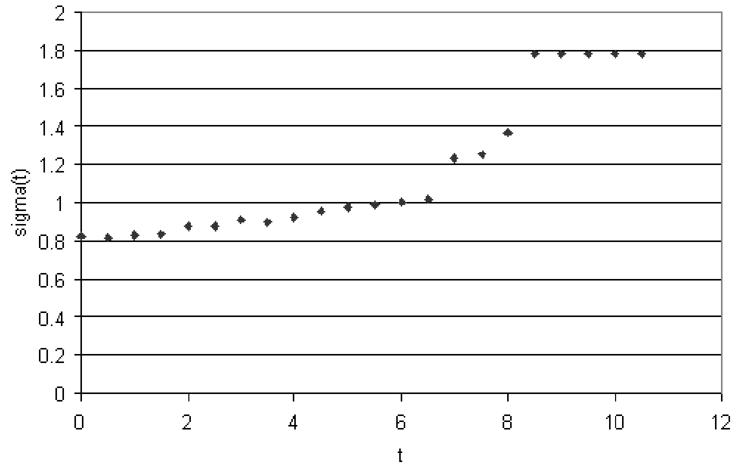

Figure 4: Calibration results for the Markovian process volatility $\sigma(t)$.

Bermudan prices (source: Markit Group). In an illustration of the phenomenon described at the start of section 3, the data supplier of the swaption and caplet prices used as calibration inputs was different to the source of the Bermudan prices used to compare the model outputs. The data supplier provided prices for caps and forward-starting caps which were then subjected to a proprietary caplet stripping method.

Results are presented in figure 6 for 10y no-call 1y EUR Bermudan swaptions. As expected, the prices of low strike payers and high strike receivers options are dominated by their intrinsic value, and hence are not sensitive to details of the model. The Hull-White model gives acceptable accuracy for

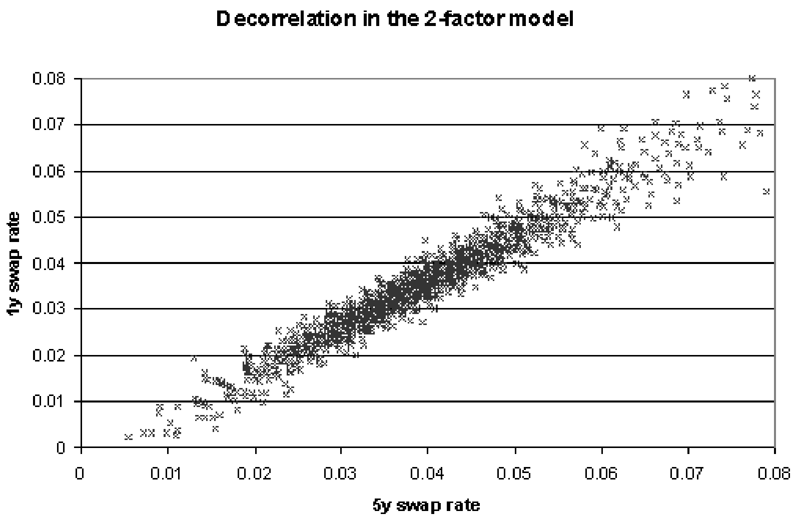

Figure 5: Decorrelation between the 5yr swap rate and the 1yr swap rate observed in 5yrs, using the 2-factor Markov Functional model. 1024 Monte Carlo samples are plotted. Integrated lognormal correlation between these two swap rates is 90.8%, comparable with that observed in the market.

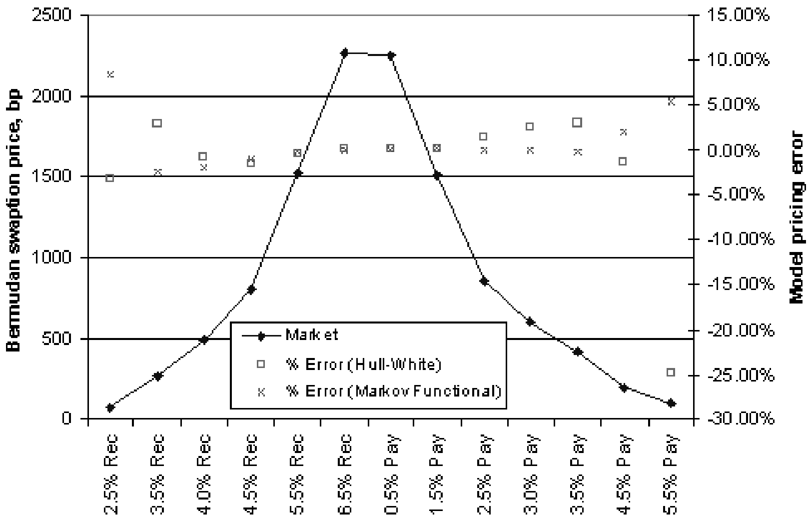

Figure 6: Comparison of market prices for Bermudan payers and receivers swaptions (Source: Markit Group) against results from the Hull-White and Markov Functional models. Prices are shown in basis points; model pricing error is defined as 100%*((model price)/(market price)-1). The Hull-White model was calibrated to ATM coterminal swaptions—see section 6 for discussion of this. The Markov Functional model was calibrated to the smile of all coterminal swaptions and to ATM caplets. Very in-the-money options are omitted for clarity as the values of these are dominated by their intrinsic value. Note that the Markov Functional model gives better results than Hull-White for the case of higher strike payers swaptions, most clearly those with strikes 3.0%, 3.5%, and 5.5%. The Hull-White model performs satisfactorily for payers swaption with strike 4.5% because this is sufficiently close to the typical strikes of the ATM swaptions used in calibration. The performance of the Markov Functional model is actually worse than the Hull-White model for the case of the 2.5% payers swaption. This is because arbitrage in the input data starts to become apparent for long-dated swaptions with strikes around 2.2%, and we have therefore chosen k_{min} = 2.5% as the lower limit in the range of calibration strikes (see section 3).

low strike receivers options because the volatility skew enforced by Hull-White is reasonably close to that observed in the market. However for high-strike payers swaptions, we find that the Markov Functional model's ability to match the smile exactly gives a significant improvement in the accuracy of pricing.

The Hull-White results were obtained using calibration to at-the-money options, to highlight the smile-sensitivity of Bermudan swaptions. In this case there is some debate regarding whether it is preferable to calibrate to at-the-money, at-the-strike, or at-the-exercise-boundary options. The second and third of these choices will give a better match to market prices than at-the-money calibration. However there are strong objections to the use of these methods in production. Firstly, there is a risk of self-arbitrage, inconsistency between the various locally calibrated models used to price options within a single Bermudan book. Second, it is not clear how to extend these methods to other products such as callable range accruals or snowballs, whilst guaranteeing stable hedge parameters.

It is important to ensure that the extra degrees of freedom added to the Markov-Functional model did not result in over-fitting market data. The calibration method described above decouples the roles of the different model parameters: the numeraire function $N(x, t)$ is calibrated to caplets, whilst the volatility function $\sigma(t)$ is calibrated to swaptions. Hunt, Kennedy and Pelsser give a simple financial interpretation of the term-structure of $\sigma(t)$ in terms of a mean-reversion parameter, controlling the terminal decorrelation. And experimentally we find that a small change in market data gives rise to a small change in model parameters, so that we believe that the system has a single global optimum.

7 Conclusion

The following numerical methods are proposed for use with the Markov-Functional model:

- a safe method to extrapolate digital option prices beyond those observed in the market;
- storage of a rescaled numeraire function, to enforce correct normalisation at all event dates;
- in the case of the 2-factor model, use of a simple linear projection function. This allows the integrals in calibration to be performed very efficiently;
- cubature techniques which can give significantly higher speed than repeated Gauss-Hermite quadrature.

When these methods are used, standard calibration as described by Hunt, Kennedy and Pelsser is extremely rapid. In fact, it can be fast enough that it is possible to calibrate the Markovian volatility functions, thereby achieving joint calibration to swaptions and caplets.

REFERENCES

■ A. Brace, D. Gatarek and M. Musiela, The Market Model of Interest Rate Dynamics, Mathematical Finance, 1997, 7, 127–155.

■ R. Cools, Constructing Cubature Formulae: the science behind the art. Acta Numerica (1997) 1–54.

■ Encyclopedia of Cubature Formulas, http://www.cs.kuleuven.ac.be/~nines/research/ecf/ecf.html

■ S. Galluccio, Z. Huang, J-M Ly, O. Scaillet, Theory and Calibration of Swap Market Models, 2004, BNP Paribas, http://www.fame.ch/library/EN/RP107.pdf

■ J. Gatheral, A parsimonious arbitrage-free implied volatility parameterization with application to the valuation of volatility derivatives, Merrill Lynch, (2004) http://www.math.nyu.edu/fellows_fin_math/gatheral/madrid2004.pdf

■ P. Hagan, D. Kumar, A.S. Lesniewski and D.E. Woodward, Managing Smile Risk, Wilmott Magazine, September 2002, 84–108.

■ P.J. Hunt and J.E. Kennedy and A. Pelsser, Markov-Functional Interest Rate Models, 2000, Finance and Stochastics, 4, 391–408.

■ P.J. Hunt and J.E. Kennedy, Longstaff-Schwartz, Effective Model Dimensionality and Reducible Markov-Functional models, 2005, http://ssrn.com/abstract=627921

■ C. Hunter, P. Jäckel and M. Joshi, Getting the Drift, RISK Magazine, July 2001.

■ M. Joshi, Rapid Computation of Drifts in a Reduced Factor Libor Market Model, Wilmott Magazine, June 2005.

■ H.B. Nielsen, Damping Parameter in Marquardt's Method, Technical Report, Technical University of Denmark, (1999).

■ V.V. Piterbarg, A Practitioner's Guide to Pricing and Hedging Callable Libor Exotics in Forward Libor Models, 2004, Bank of America—Quantitative Research, http://ssrn.com/abstract=427084

■ W. H. Press, S. A. Teukolsky, W. T. Vetterling and B. P. Flannery, Numerical Recipes in C, Cambridge University Press, 1992.

■ R. Rebonato, Volatility and Correlation, Wiley (2004).

ACKNOWLEDGEMENTS

Thanks for useful discussions to Denis Desbiez, Sergio Dutra, Martin Forde, Peter Jäckel, Rhodri Wynne, and particularly Ralph Sebastian. Thanks also to an anonymous referee for valuable comments. The views expressed in this article are personal and do not represent the views of Commerzbank.