The University of Manchester
Manchester Business School

Interest Rate Models: An ALM Perspective

Ser-Huang Poon
Manchester Business School

Interest Rate Models: An ALM Perspective (with NAG implementation)

Ser-Huang Poon
Manchester Business School

Full paper: http://www.personal.mbs.ac.uk/ser-huang-poon/

Joint work with Eric Guan, Bing Gan and Aisha Khan
All postgraduate students at MBS
MSc QFFE (Quantitative Finance and Financial Engineering) Industry Linked Project

The University of Manchester
Manchester Business School

QFFE Course Structure

<u>Semester I: 3 months</u>

- Asset Pricing Theory
- Derivative Securities
- Stochastic Calculus
- VBA, C++

<u>Optional units</u>

- Portfolio Investment (Barra)
- Macro and International Investment
- Martingale Process

<u>Semester II: 3 months</u>

- Financial Econometrics
- Interest Rate & Credit Derivatives

<u>Optional units</u>

- Credit Risk Modelling & Management
- Computational Finance
- Mathematical Modelling of Finance
- Real Options

Summer Dissertation: June-July, Preparation starts from Spring

Submission of thesis in first week of September

Project Remit

- Proposed by Group ALM of ABN AMRO in Amsterdam
- How sensitive is model choice to ALM outcome
  - Consider switching from BK to HW
  - Question if they should move to more sophisticated model: LMM or SMM
- No access to bank's confidential data
- Four contesting models:
  - Hull-White (HW)
  - Black-Karasinski (BK)
  - Libor Market Model (LMM) and
  - Swap Market Model (SMM)
- Test 20 annual hedging performance of 10x1 Bermudan swaption of the single factor version of the four contesting models on EURO and USD.

Different Types of Interest Rates

- Short rate

$$dr = (\theta(t) - ar)dt + \sigma dz$$

- Forward rate
- Libor rate

$$dL_i(t) = \mu_i(t)L_i(t)dt + \sigma_i(t)^T L_i(t)dW(t)$$
$$0 \le t \le T_i, \quad i = 1, 2, \ldots N$$

$$dS_{n,m}(t) = S_{n,m}(t)\sigma_{n,m}(t)dW_t^{n,m}$$

- Swap rate
  - Fixed strike in exchange for a series of floating rates
  - Payer vs. Receiver

[No arbitrage condition must hold for all interest rates and all interest rate products]

Cash Flows and Rates

$$
\begin{array}{ccccccccc}
0 & \text{---} & 1 & \text{---} & 2 & \text{---}\cdots\text{---} & t & \text{---}\cdots\text{---} & T \\
r_0 & & r_1 & & r_2 & & r_t & & r_T
\end{array}
$$

$$
\begin{array}{ccccccc}
f_0(0) & & & & & & \\
f_1(0) & f_1(1) & & & & & \\
f_2(0) & f_2(1) & f_2(2) & & & & \\
\vdots & & & & & & \\
f_T(0) & f_T(1) & f_T(2) & \cdots & f_T(t) & \cdots & f_T(T)
\end{array}
$$

$$
\begin{array}{ccccccc}
S_{0,T}(0) & & & & & & \\
S_{1,T}(0) & S_{1,T}(1) & & & & & \\
S_{2,T}(0) & S_{2,T}(1) & S_{2,T}(2) & & & & \\
\vdots & & & & & & \\
S_{T-1,T}(0) & S_{T-1,T}(1) & S_{T-1,T}(2) & \cdots & S_{n,m}(t) & \cdots & S_{T-1,T}(T)
\end{array}
$$

Plain Vanilla Products

- Yield curve, Y(T)
- Bond prices
- Forward rate agreement (FRA)
- Caplet, floorlet, cap, floor
- Swap, forward starting swap
- Swaption: European vs. Bermudan

Price quotation in Black Vol

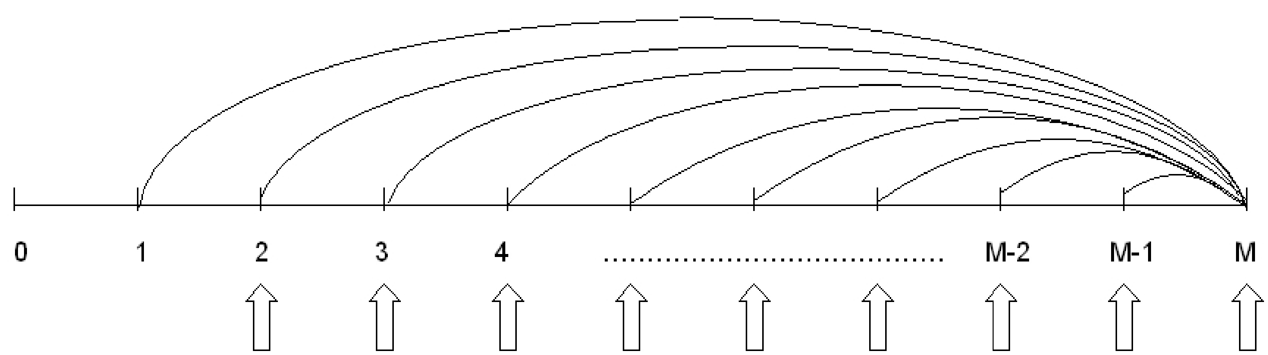
- Only for European (ATM) products and under the terminal measure
- Bloomberg gives implied vol for Gaussian model also

Swap Tenor	Swaption Maturities									
Mean	1	2	3	4	5	6	7	8	9	10
1	19.63%	19.39%	18.86%	18.12%	17.38%		16.03%			14.50%
2	19.44%	18.99%	18.36%	17.62%	16.85%		15.52%			
3	19.09%	18.49%	17.81%	17.07%	16.36%		15.12%			
4	18.74%	17.93%	17.22%	16.54%	15.90%		14.76%			
5	18.26%	17.38%	16.69%	16.05%	15.48%					
6	17.67%	16.88%	16.24%	15.68%	15.19%					
7	17.13%	16.43%	15.90%	15.40%						
8	16.66%	16.06%	15.59%							
9	16.23%	15.72%								
10	15.85%									

$$PSwaption_{t,T} = \delta \sum_{j=n+1}^{N+1} B_{t,T_j} \left[S(t,T_i,T_N) N(d_1) - kN(d_2) \right]$$

Bermudan Swaption

- Constant maturity swap vs. Fixed tail swap
- Fixed tail swap resembles home mortgage with prepayment feature
- Fixed tail swap can be hedged using 10x1 co-terminal European swaptions



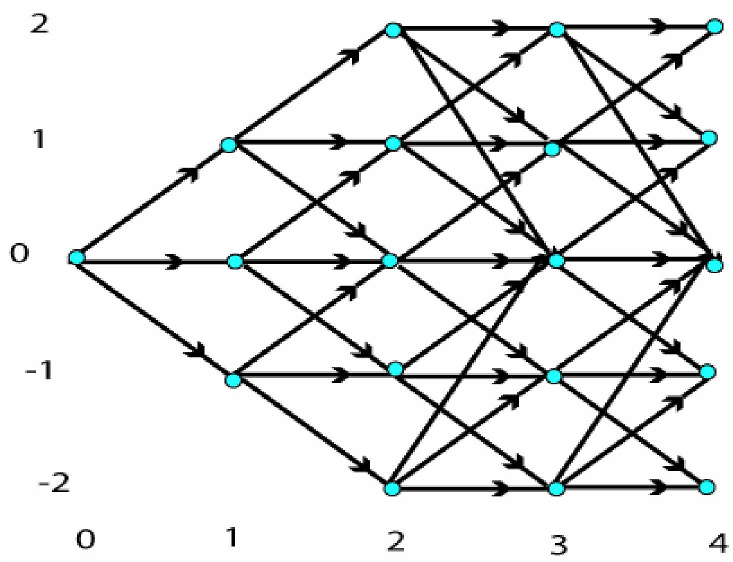
Choice of Calibrated Instruments

- Guided by no-arbitrage or hedging trades
- Rates fixed using bond prices or FRA
- Volatility fixed using caplet or European swaptions
- The choice of calibrated instruments determines the complexity of optimisation procedure and the time needed to reach convergence.

Swap Tenor	Swaption Maturities									
Mean	1	2	3	4	5	6	7	8	9	10
1	19.63%	19.39%	18.86%	18.12%	17.38%		16.03%			14.50%
2	19.44%	18.99%	18.36%	17.62%	16.85%		15.52%			
3	19.09%	18.49%	17.81%	17.07%	16.36%		15.12%			
4	18.74%	17.93%	17.22%	16.54%	15.90%		14.76%			
5	18.26%	17.38%	16.69%	16.05%	15.48%					
6	17.67%	16.88%	16.24%	15.68%	15.19%					
7	17.13%	16.43%	15.90%	15.40%						
8	16.66%	16.06%	15.59%							
9	16.23%	15.72%								
10	15.85%									

Short Rate Trinomial Tree

$$dr = (\theta(t) - ar)dt + \sigma dz$$



- Time step is 0.1 year, 11year=110 time steps
- θ(t) calibrated to bond prices
- Mean reversion rate a(t)->a, estimated from time series data
- σ(t) follows 3-point interpolation

Simulate LMM and SMM under T(n+1) measure

$$\frac{dL_i(t)}{L_i(t)} = -\sum_{j=i+1}^{N} \frac{\delta_j L_j(t)\sigma_{ij}(t)}{1+\delta_j L_j(t)} dt + \sigma_i(t) \cdot dW^{Q^T_{n+1}}(t)$$

$$dS_{n,M}(t) = S_{n,M}(t)\sigma_{n,M}(t)dW_t^M + drift$$

$$\mu_n = \sum_{i=n}^{M-2} \left(\tau_{i+1} \frac{C_{i+1,M}}{C_{M-1,M}} \rho_{n,i+1} S_{n,M} \sigma_{n,M} S_{i+1,M} \sigma_{i+1,M} \prod_{j=n+1}^{i} (1+\tau_j S_{j,M}) \right)$$

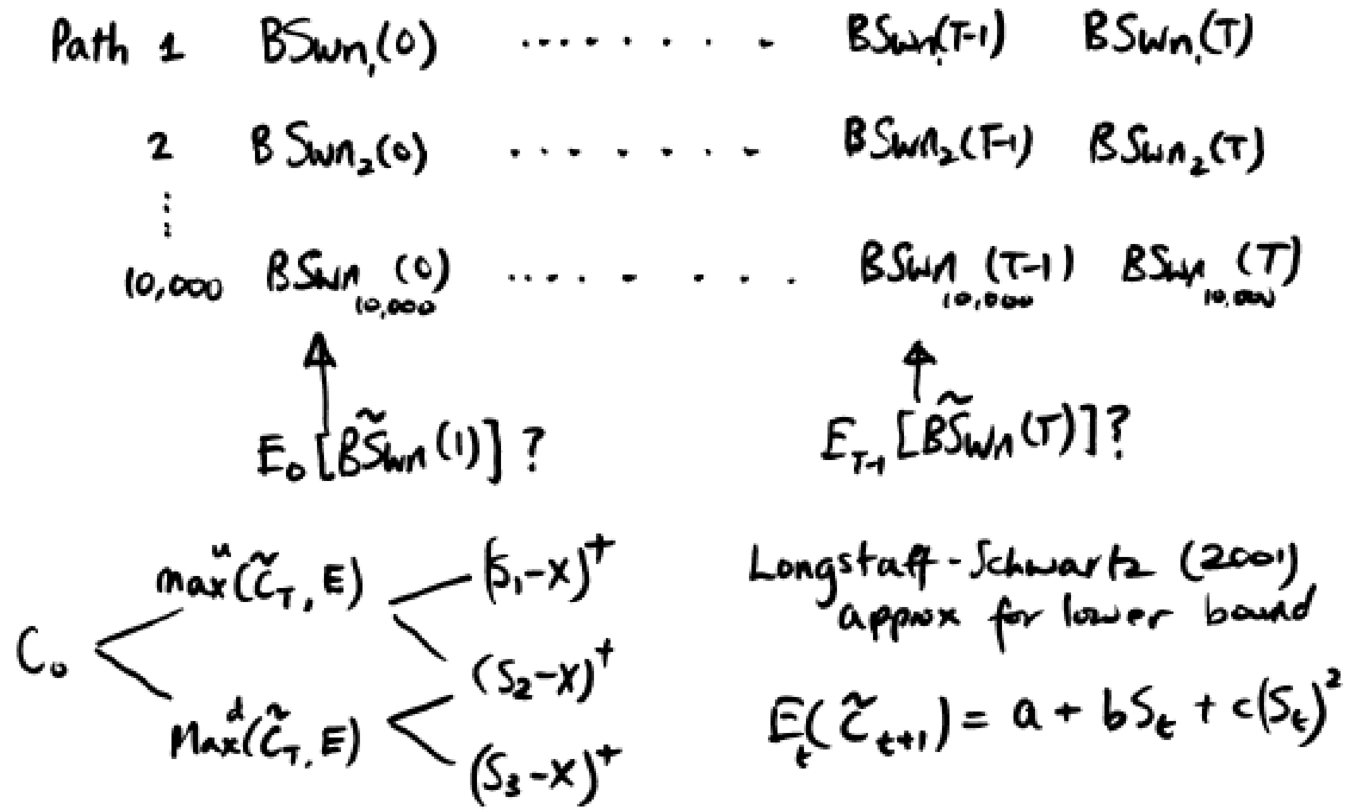
LMM& SMM Simulations

- For each path

$$
\begin{array}{ccccccc}
0 & 1 & 2 & \cdots & t & \cdots & T \\
L_T(0) & L_T(1) & L_T(2) & \cdots & L_T(t) & \cdots & L_T(T) \\
\vdots & & & & & & \\
L_{2,T}(0) & L_{2,T}(1) & L_{2,T}(2) & & & & \\
L_{1,T}(0) & L_{1,T}(1) & & & & & \\
L_{0,T}(0) & & & & & &
\end{array}
$$

- Delta δ=1 yr, 10,000 simulations, single factor model
- Time step δ is not crucial if predictor-corrector method is used to correct the drift
- Use NAG random number generator **g05ddc**

Approximation for early exercise decision



- Call regression routine **g02dac**

Calibration

1. Convert Black Vol into European swaption prices using **g01eac** (prob for std normal distribution).

2. Build short rate HW, estimate σ_0, σ_3 and σ_{11} by minimising RMSPE of 10 co-terminal European swaptions using NAG **e04unc** (solving nonlinear least square using sequential QP).

3. Use the calibrated model to calculate 10x1 Bermudan swaption prices by stepping backward through the tree.

4. Build short rate BK; repeat steps 2-3 for lognormal rate.

Calibration (cont'd)

5. LMM calibrated to European swaption

Use Rebonato's approximation of swaption vol

$$\left(v_{n,M}^{LFM}\right)^2 = \sum_{i,j=\alpha+1}^{\beta} \frac{\omega_i(0)\,\omega_j(0)\,L_i(0)\,L_j(0)\,\rho_{ij}}{T_n\left[S_{n,M}(0)\right]^2} \int_0^{T_n} \sigma_i(t)\,\sigma_j(t)\,dt$$

where $v_{n,M}^{LFM}$ is a proxy for Black volatility $v_{n,M}(T_n)$.

$$\begin{aligned} \sigma_k &= \phi_k \times \varphi_k(a,b,c,d) \\ \varphi_k(a,b,c,d) &= \left[a(T_k - t) + b\right] e^{-c(T_k - t)} + d \end{aligned}$$

Call Nag **e04unc** to minimise vol errors for 10 rates instead of minimising 10 pricing errors using 10,000 simulation for each loop

Calibration (cont'd)

(Cont'd LMM)

- 10,000 simulation (**g05ddc**) to calculate European swaption prices,
- 10 regressions (each with 10,000 observations, **g02dac**) to derive Bermudan prices.

6. SMM; same as step 5 but for SMM. Need additional step to solve simultaneous equations (**f07adc** LU factorization, and **f07aec** solution of real sys of linear equations) in order to derive the discount factor.

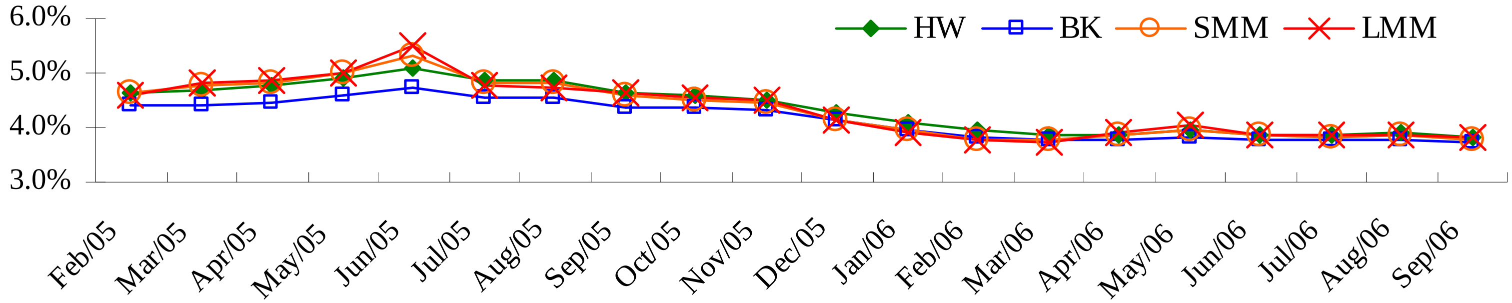
7. Repeat steps 2-6 for pricing 9x1 swaption.

8. Repeat steps 2-7 20 times through the sample period.

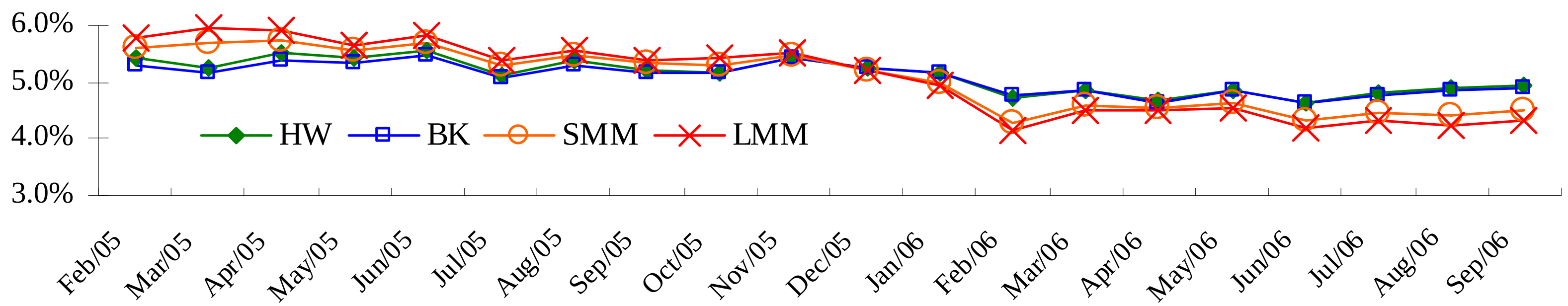
9. Do it once for Euro rate; repeat the whole process for USD interest rate; supposed to calibrate to Brazilian data, but no appropriate market data.

11-Y Bermudan Swaption Prices

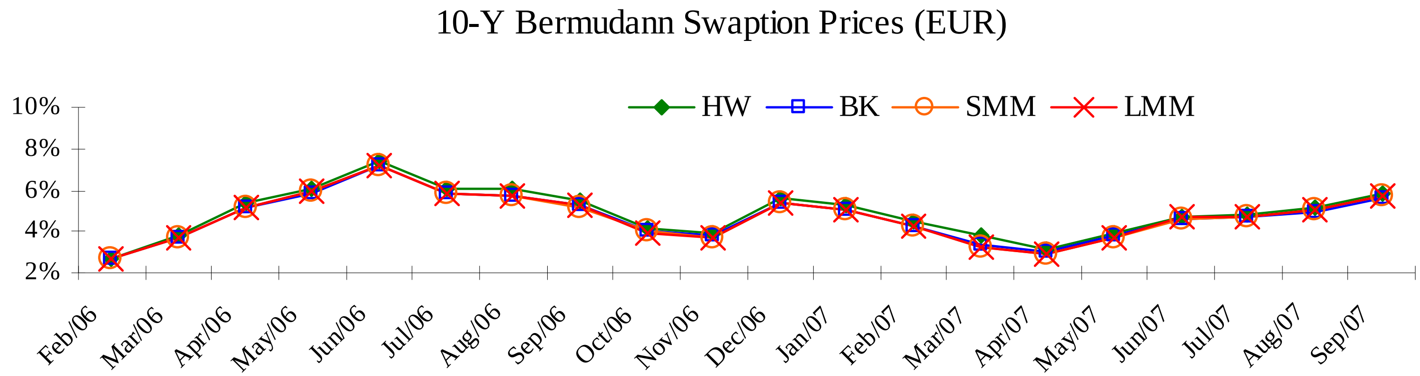
11Y Bermudan Swaption Prices (EUR)

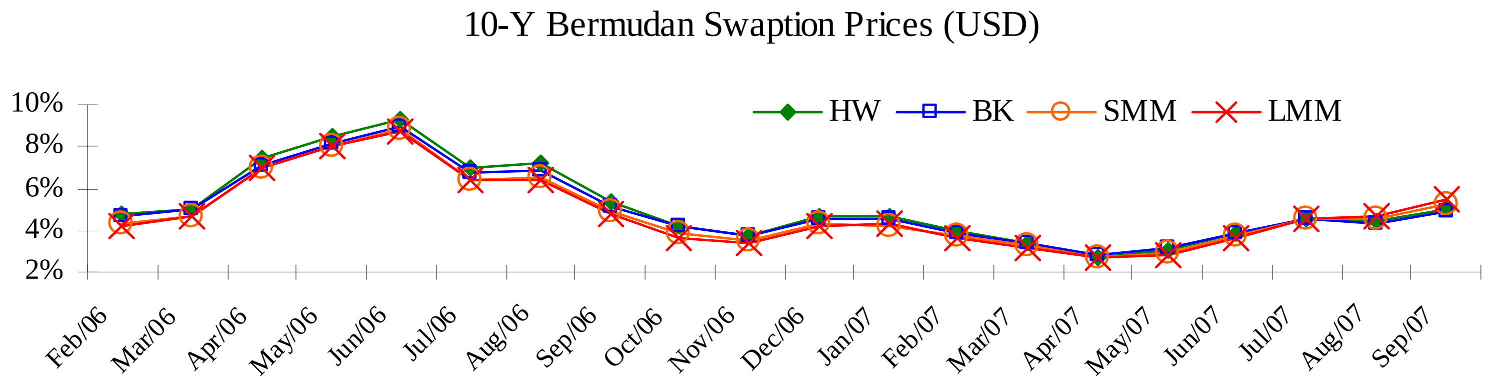


11Y Bermudan Swaption Prices (USD)

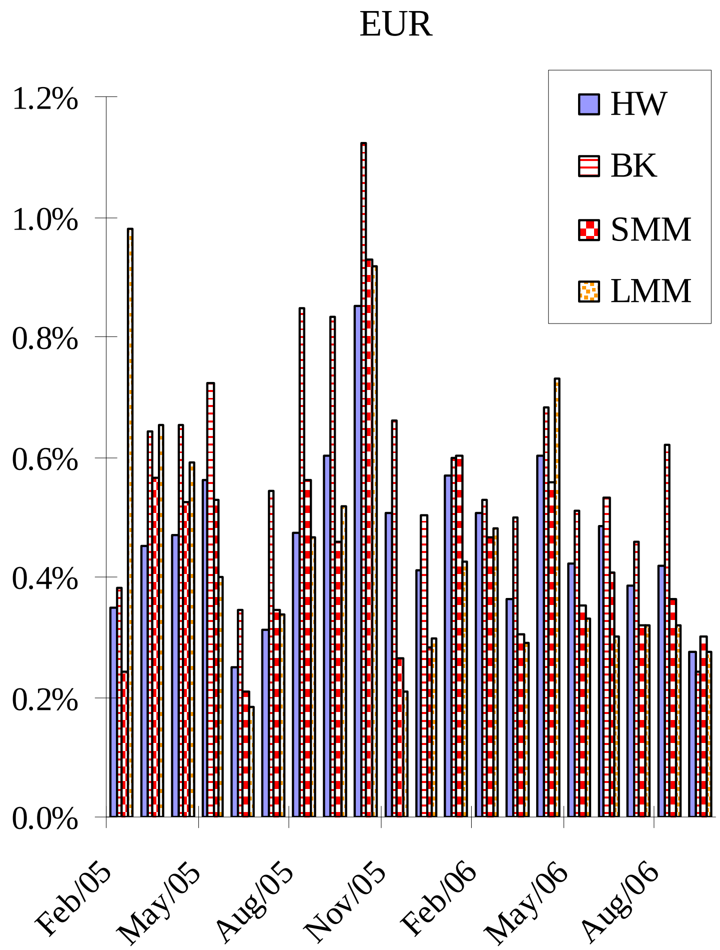


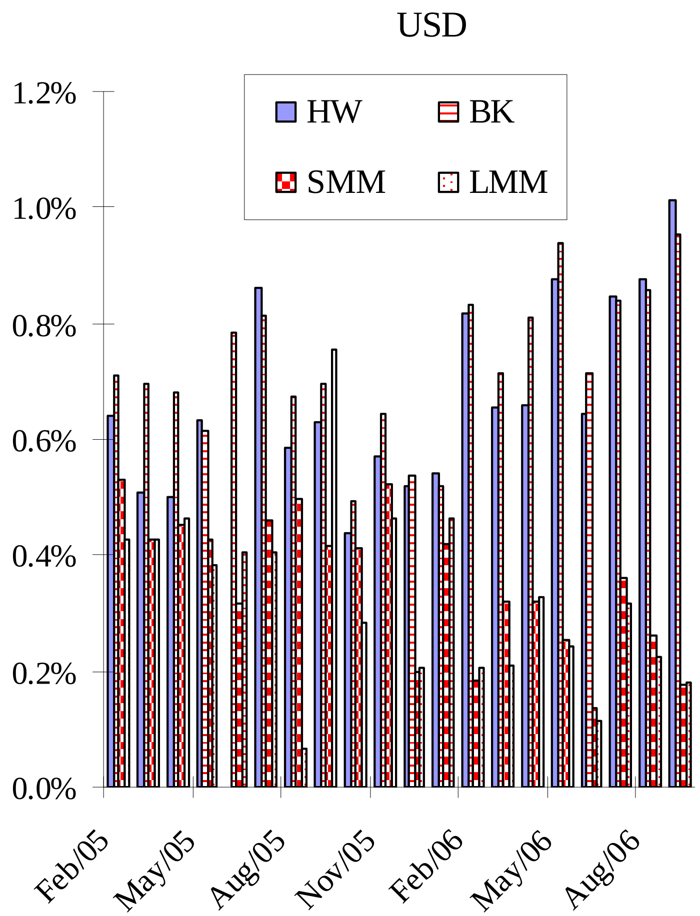
10-Y Bermudan Swaption Prices



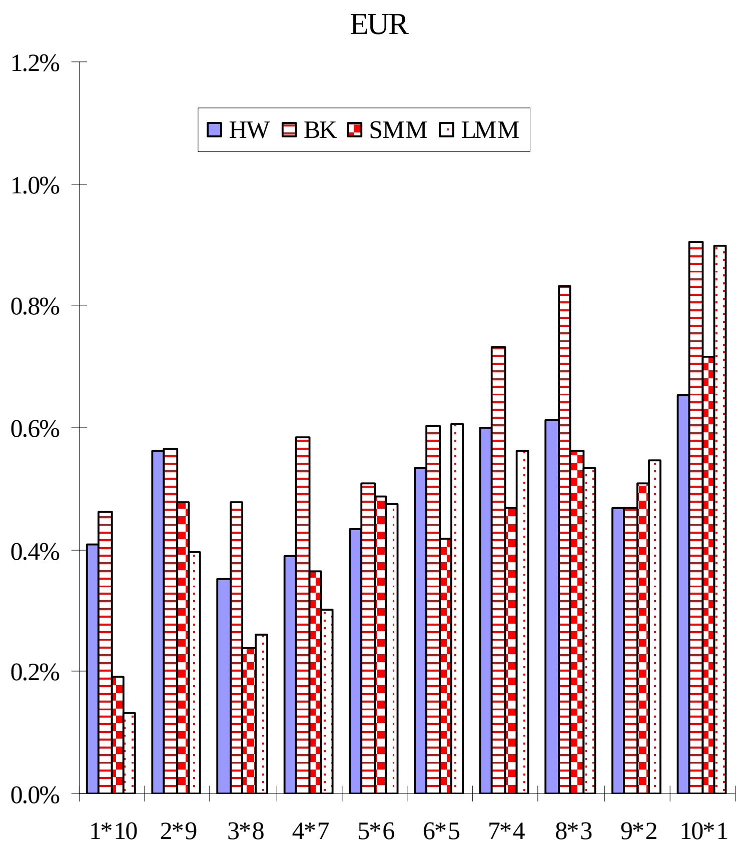


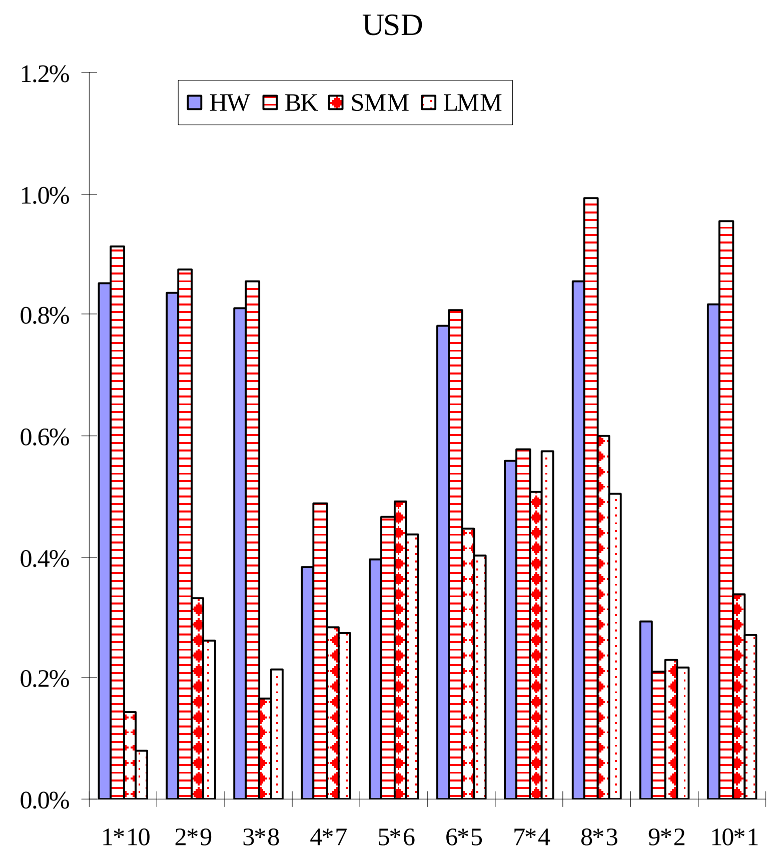
RMSE (Pricing Error) by Date

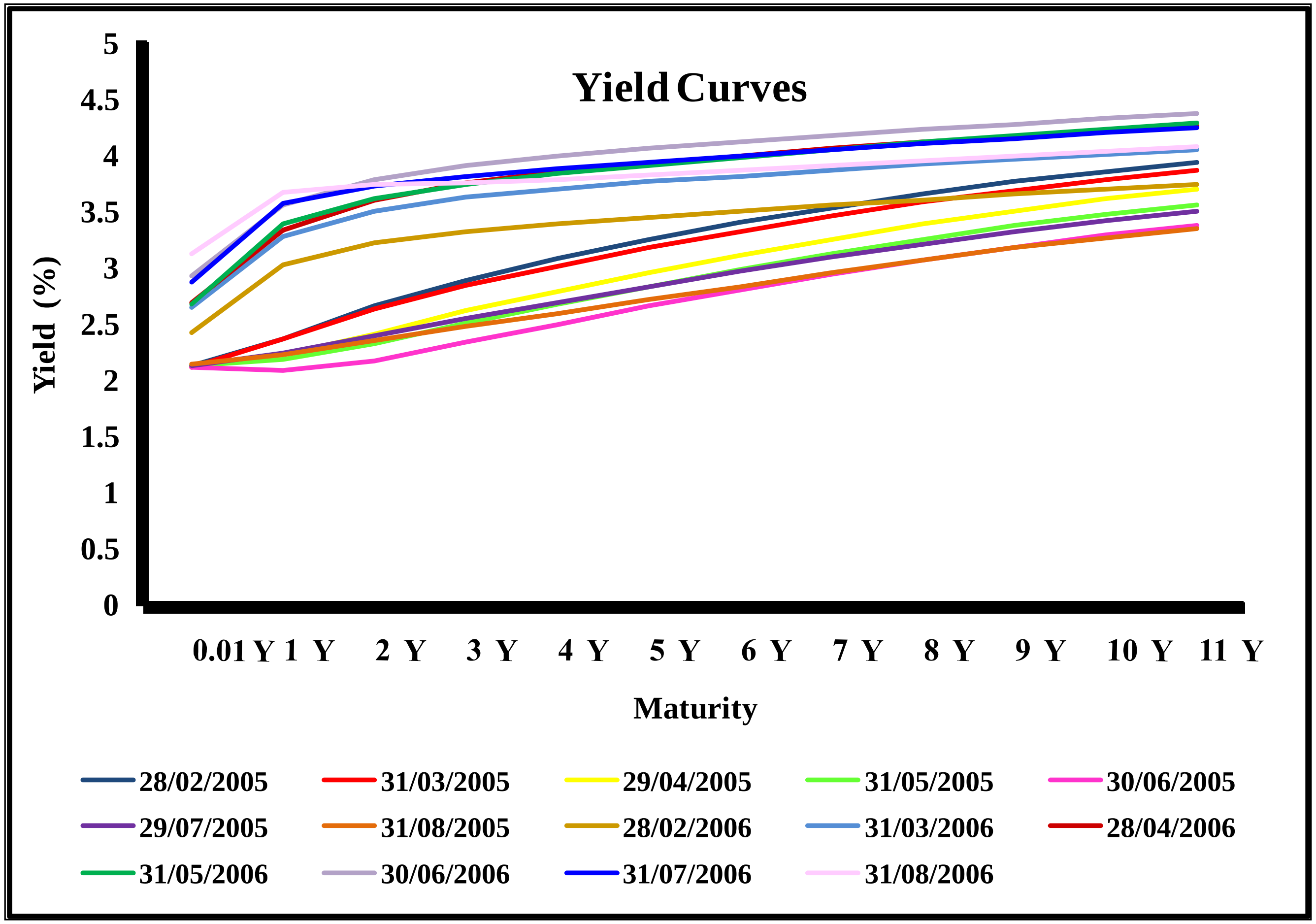


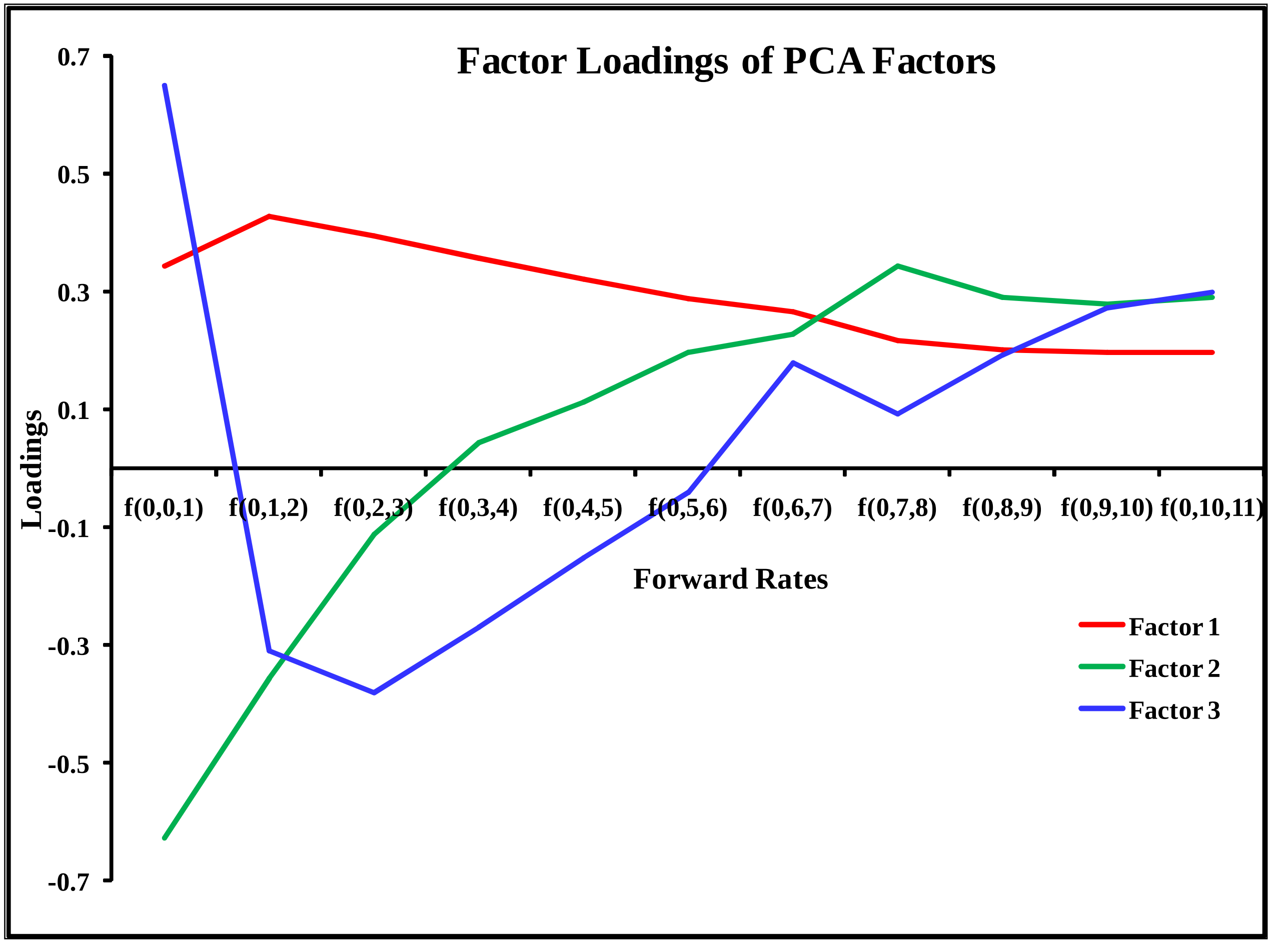


RMSE (Pricing Error) by Contract






Yield Curves
Yield (%)
5
4.5
4
3.5
3
2.5
2
1.5
1
0.5
0
0.01 Y
1 Y
2 Y
3 Y
4 Y
5 Y
6 Y
7 Y
8 Y
9 Y
10 Y
11 Y
Maturity
28/02/2005
31/03/2005
29/04/2005
31/05/2005
30/06/2005
29/07/2005
31/08/2005
28/02/2006
31/03/2006
28/04/2006
31/05/2006
30/06/2006
31/07/2006
31/08/2006



Use NAG **g03aac** to carry out PCA analysis

Delta Hedging

- "Bumping" the yield curve by perturbing the first three PCA factors by $\pm$ mean absolute change.
- Assume no change in interest rate volatility.
- Calculate price sensitivities (delta) of 10x1 Bermudan swaption, and the hedge instruments (swaps, 1-, 5- and 11-year)
- Form a delta hedged portfolio at t i.e. Σ delta=0. Use NAG **e04ccc** (unconstrained min using simplex algorithm).
- Unwind at t+1 and use price information of 9x1 Bermudan swaption and swaps (0-, 4- and 10-year) to calculate portfolio value at t+1.
- Calculate P&L; good model should give P&L=0.

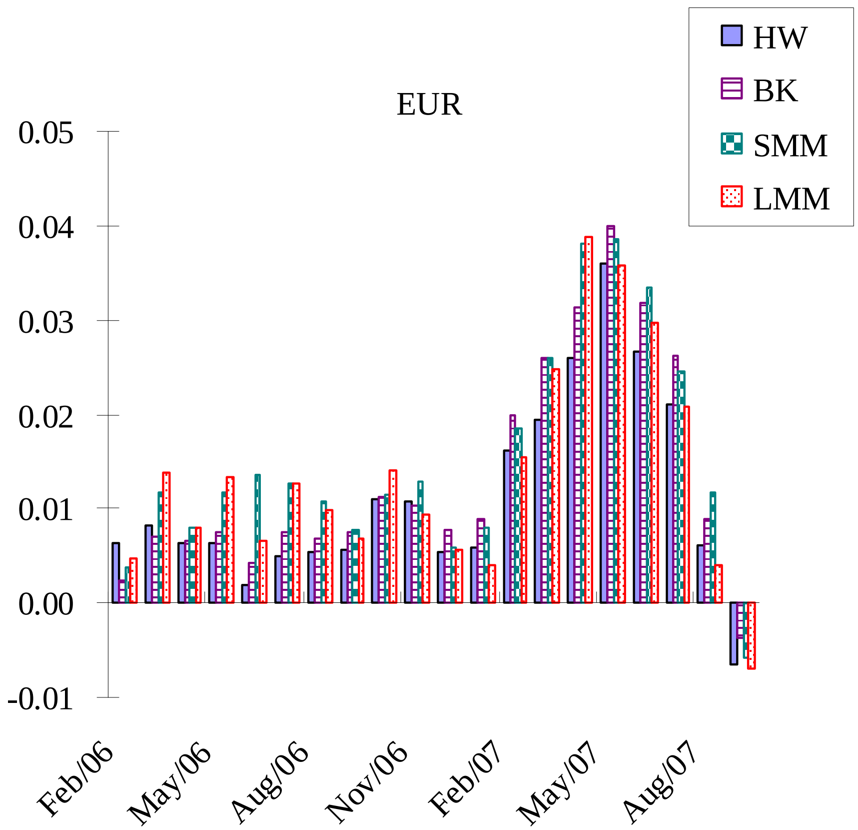
P&L Analysis

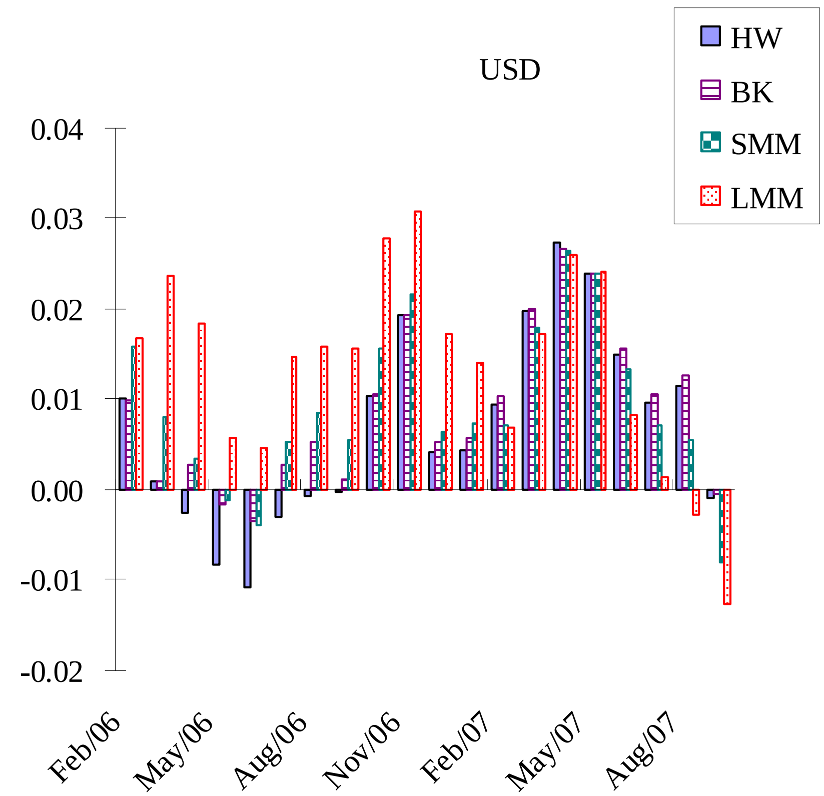
- nag_opt_simplex (e04ccc) is used
- RMSE of Hedging P&L

	HW	BK	SMM	LMM
EUR	1.486%	1.751%	1.886%	1.752%
USD	5.544%	5.484%	5.701%	7.754%

The University of Manchester
Manchester
Business School

P&L of Hedging Results





Summary

- Pricing and hedging results are similar for all four models.
- HK~BK, and LMM~SMM
- For 1-factor pricing and long term risk management, there is little to choose between them. One should focus more on setting scenarios.
- Given 1-factor finding above, for ALM purpose, HW seems best; fast and better convergence property when compared with BK.

The University of Manchester
Manchester
Business School

MANCHESTER 1824

NAG Visual Studio C++

- Visual C++ Express 2005 - CLDLL07XL (version 8 is now available)

<u>One time link for all current projects when you start working in the C++ studio</u>

- Started Visual Studio, click on Tools -> Options; Expand 'Projects and Solutions' and click on VC++ Directories;
- Change the 'Show directories for:' box so that it refers to include files and add the following line C:\Program Files\Numerical Algorithms Group\CLDLL074X\include
- Change the 'Show directories for:' box to show library files and add the following line C:\Program Files\Numerical Algorithms Group\CLDLL074X\include
- Finally change the box to show Executable files and add the line C:\Program Files\Numerical Algorithms Group\CLDLL074X\include
- Then click on OK. Do this once for all projects.

C++ (Cont'd)

To compile:

- Right click on the project name and click on 'properties'.
- Expand 'Configuration Properties' and then 'Linker'.
- Now click on 'Input'.
- The resulting window will contain an empty 'additional dependencies' line. Add nagc.lib to this. You should now be able to compile.

Calling C-Nag library from C++ code

- The best way to start is to download the example programme from NAG website, and modify the programme to suit your application.

The University of Manchester
Manchester
Business School

MANCHESTER 1824

QFFE Industry Linked Project

- Recognition of the needs of financial practitioners. The course is structured to
  - 1) better prepare the students for life in market roles, and
  - 2) investigate real world issues through the dissertation.

- Invitation for small, self-contained industry relevant research topics
- Students are available for internship from September to December.
- Contributors include:
  - Alliance and Leicester,
  - Standard Bank
  - Morgan Stanley
  - ABN AMRO
  - Shell
  - Numerical Algorithms Group
  - Riskmetrics