

The Practicalities of Libor Market Models

Peter Jäckel

pj@otc-analytics.com

There are many publications on the theory of the Libor market model and its extensions. There are very few sources on the issues a practitioner faces during implementation and operation of the model. This presentation is on the subject of how to make a Libor Market Model work in practice.



Contents

I. Standard and skewed Libor market model dynamics	3
II. Derivation of the indirectly stochastic drift	16
III. Leaving the canon	22
IV. Futures convexity corrections in the Libor market model	27
V. Speed is everything — the predictor-corrector scheme	39
VI. Parametrisation of correlation and volatility backbone	44
VII. Factor reduction — pros and cons	49

VIII. Speed is everything — the drift term	53
IX. Analytical calibration to coterminal swaptions	58
X. Non-parametric volatility specification	75
XI. Global calibration to the full swaption matrix	80
XII. Bermudan Monte Carlo	97
XIII. Cross-currency Libor market modelling	144
XIV. Calibration of FX volatilities in a cross-currency Libor market model	151



I. Standard and skewed Libor market model dynamics

The concept of a *market model* is to describe directly the dynamics of observable market quotes of financially tradeable contracts, rather than to fall back on a hidden process driving the entirety of the fixed income market.

A *Libor market model* is based on the discretisation of the yield curve into *discrete spanning forward rates*.

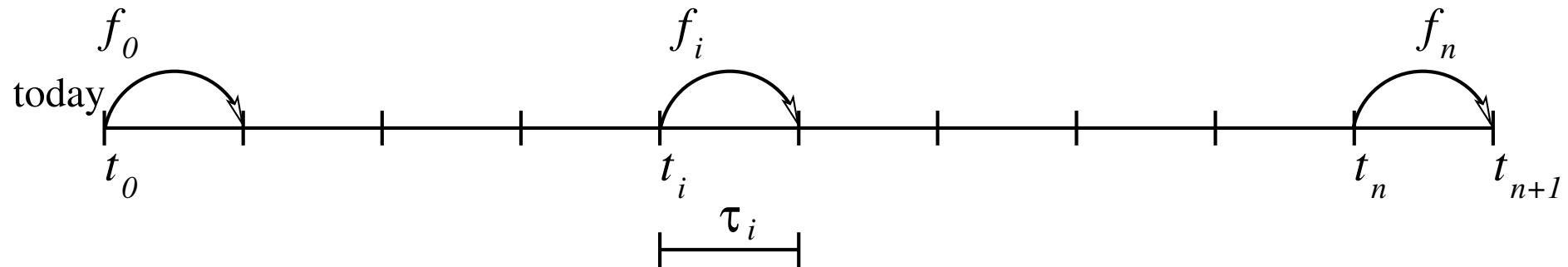


Figure I.1.

Each forward rate immediately represents the (modelled) market quote for an associated Forward Rate Agreement (FRA).

A forward rate agreement quote f_i for period $t_i \rightarrow t_{i+1}$ with accrual factor $\tau_i \simeq t_i - t_{i+1}$ means:

- Upon deposit of a notional at time t_i , at the later time t_{i+1} the notional plus interest amounting to $f_i \cdot \tau_i$ times the notional is returned to the depositor.
- For a borrower of money, the effective funding discount factor over the (forward) interval $t_i \rightarrow t_{i+1}$ is given by $1 / (1 + f_i \tau_i)$.

The fair value of a Forward Rate Agreement on rate f_i struck at K is

$$P(t, t_{i+1}) \cdot (f_i - K) \tau_i$$

where $P(t, t_{i+1})$ is the value at time t of a zero coupon bond paying one domestic currency unit at time t_{i+1} .

At time t_i , the value becomes $(f_i - K) \tau_i / (1 + f_i \tau_i)$.



In the standard Libor market model for discretely compounded interest rates, we assume that each of n spanning forward rates f_i evolves according to the stochastic differential equation

$$\frac{df_i}{f_i} = \mu_i^{\text{LN}}(f, t)dt + \sigma_i(t)d\widetilde{W}_i. \quad (1)$$

This ensures that all interest rates remain positive at all times.

Note:

the drift terms are yet to be determined by the aid of no-arbitrage arguments!

Correlation is incorporated by the fact that the n standard Wiener processes in equation (1) satisfy

$$\mathbb{E}\left[d\widetilde{W}_i d\widetilde{W}_j\right] = \rho_{ij}dt. \quad (2)$$

The elements of the instantaneous covariance matrix $C(t)$ of the n forward rates are thus

$$c_{ij}(t) = \sigma_i(t)\sigma_j(t)\rho_{ij}. \quad (3)$$



Using a decomposition of $C(t)$ into a pseudo-square root A such that

$$C = AA^\top, \quad (4)$$

we can transform equation (1) to

$$\frac{df_i}{f_i} = \mu_i^{\text{LN}} dt + \sum_j a_{ij} dW_j \quad (5)$$

with dW_j being n independent standard Wiener processes where dependence on time has been omitted for clarity.

The matrix A is sometimes also referred to as the *driver*, or as the *dispersion*¹ matrix, or even as the *factor loading* matrix.

Note : it is not advisable to assume that $t_{i+1} - t_i \equiv \tau$ for some constant τ for all i since the error incurred by this approximation is too large.

¹Karatzas and Shreve [KS91], page 284.



A commonly used extension is to introduce an implied volatility skew by permitting the Libor rates to experience a *displaced diffusion* [Rub83]:

$$\frac{d(f_i + s_i)}{f_i + s_i} = \mu_i^{\text{DD}}(f, s, t) dt + \sigma_i^{\text{DD}} dW_i. \quad (6)$$

This model can be calibrated comparatively easily to at-the-forward implied volatility $\hat{\sigma} := \hat{\sigma}(f)$ quotes and the skew at the forward,

$$\hat{\sigma} = \zeta + \frac{1-\beta^2}{24} \zeta^3 T + \frac{7-10\beta^2+3\beta^4}{1920} \zeta^5 T^2 + \mathcal{O}(\zeta^7) \quad (7)$$

$$\zeta = \hat{\sigma} + \frac{\beta^2-1}{24} \hat{\sigma}^3 T + \frac{3-10\beta^2+7\beta^4}{1920} \hat{\sigma}^5 T^2 + \mathcal{O}(\hat{\sigma}^7) \quad (8)$$

$$\left. \frac{d\hat{\sigma}(K)}{dK} \right|_{K=f} = -\frac{(1-\beta)}{2} \frac{\hat{\sigma}}{f} \cdot \left[1 + \frac{1}{12} \hat{\sigma}^2 T + \frac{1}{240} \hat{\sigma}^4 T^2 + \frac{1}{6720} \hat{\sigma}^6 T^3 + \mathcal{O}(\hat{\sigma}^8) \right], \quad (9)$$

where we have used the parametrisation

$$\sigma_i^{\text{DD}} = \beta_i \zeta_i \quad \text{and} \quad s_i = (1 - \beta_i) \frac{f_i(0)}{\beta_i}. \quad (10)$$



Another common extension is to introduce *Constant Elasticity of Variance* [Bec80, CR76, Sch89, AA00]:

$$df_i = \mu_i^{\text{CEV}}(f, \beta, t) dt + \sigma_i^{\text{CEV}} f_i^{\beta_i^{\text{CEV}}} dW_i. \quad (11)$$

For moderate maturities, the following expansions can be used for calibration:

$$\hat{\sigma} = \sigma^{\text{CEV}} \cdot f^{(\beta^{\text{CEV}} - 1)}(0) + \mathcal{O}(\sigma^{\text{CEV}}{}^3) \quad (12)$$

$$\sigma^{\text{CEV}} = \hat{\sigma} \cdot f^{(1 - \beta^{\text{CEV}})}(0) + \mathcal{O}(\hat{\sigma}{}^3) \quad (13)$$

$$\left. \frac{d\hat{\sigma}(K)}{dK} \right|_{K=f} = -\frac{(1-\beta)}{2} \frac{\hat{\sigma}}{f} + \mathcal{O}(\hat{\sigma}{}^3), \quad (14)$$

For longer maturities, one ought to use the general CEV option pricing formula

$$\mathbb{E}[(f - K)_+] = \begin{cases} f \cdot Q(a, 2 + b, c) - K \cdot [1 - Q(c, b, a)] & \text{for } \beta < 1 \\ f \cdot Q(c, -b, a) - K \cdot [1 - Q(a, 2 - b, c)] & \text{for } \beta > 1 \end{cases} \quad (15)$$

with

$$a = \frac{K^{2(1-\beta)}}{(1-\beta)^2 \sigma^{\text{CEV}}^2 T} \quad (16)$$

$$b = \frac{1}{1-\beta} \quad (17)$$

$$c = \frac{f(0)^{2(1-\beta)}}{(1-\beta)^2 \sigma^{\text{CEV}}^2 T} \quad (18)$$

and $Q(a, b, c)$ being the complementary non-central χ^2 distribution function for b degrees of freedom and non-centrality parameter c .

Note: the Constant Elasticity of Variance distribution has a continuous and a discrete part due to the fact that for $\beta < 1$ and $t > 0$ there is a positive probability that $f = 0$. To avoid confusion with your implementation, we have defined all terms by the complementary distribution function.

The implied volatility surfaces of the displaced diffusion model and the constant elasticity of variance model are very similar for a wide range of strikes and skew coefficients β . However, the constant elasticity of variance model does not permit long time step Monte Carlo integration methods as efficiently as the displaced diffusion model.

This is why the displaced diffusion model is very frequently used to match a market observable implied volatility skew.

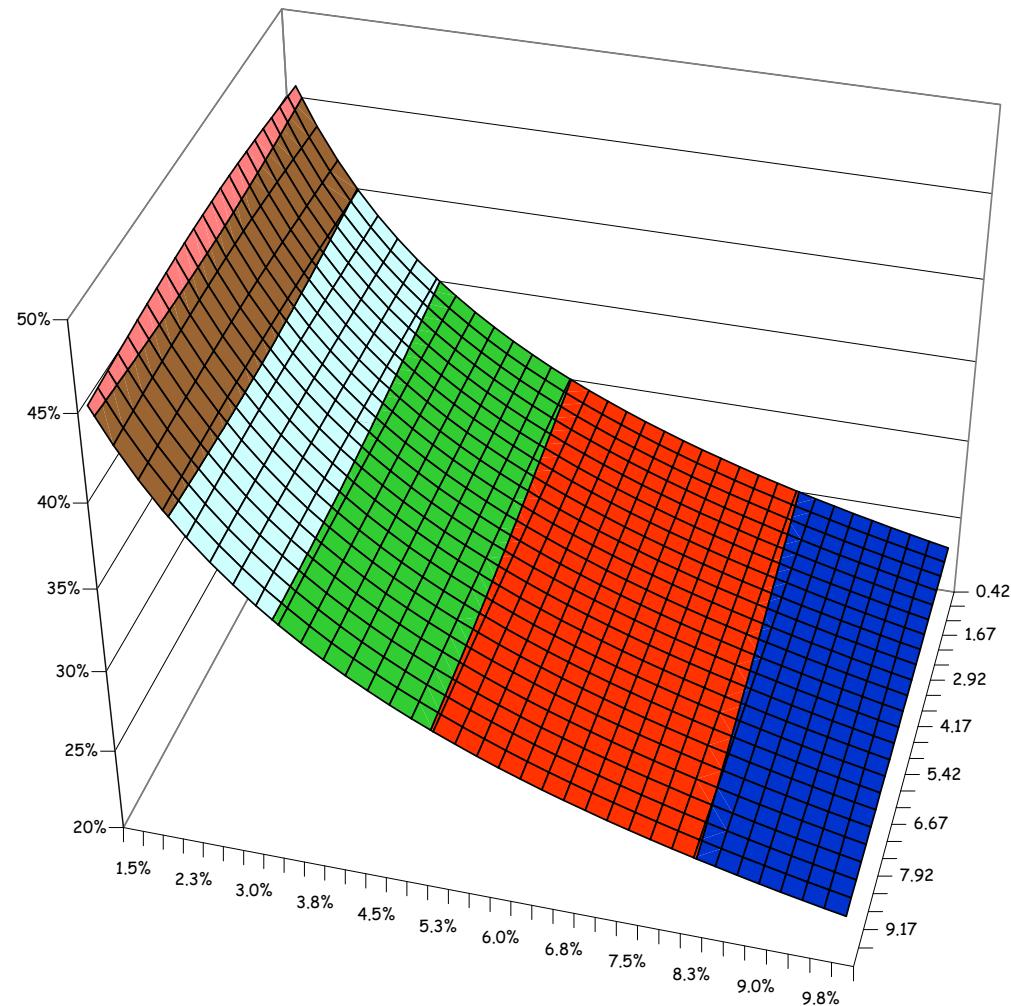


Figure I.2. Implied volatilities for CEV model with $f = 5\%$, $\beta = 1/4$, $\zeta = 30\%$.

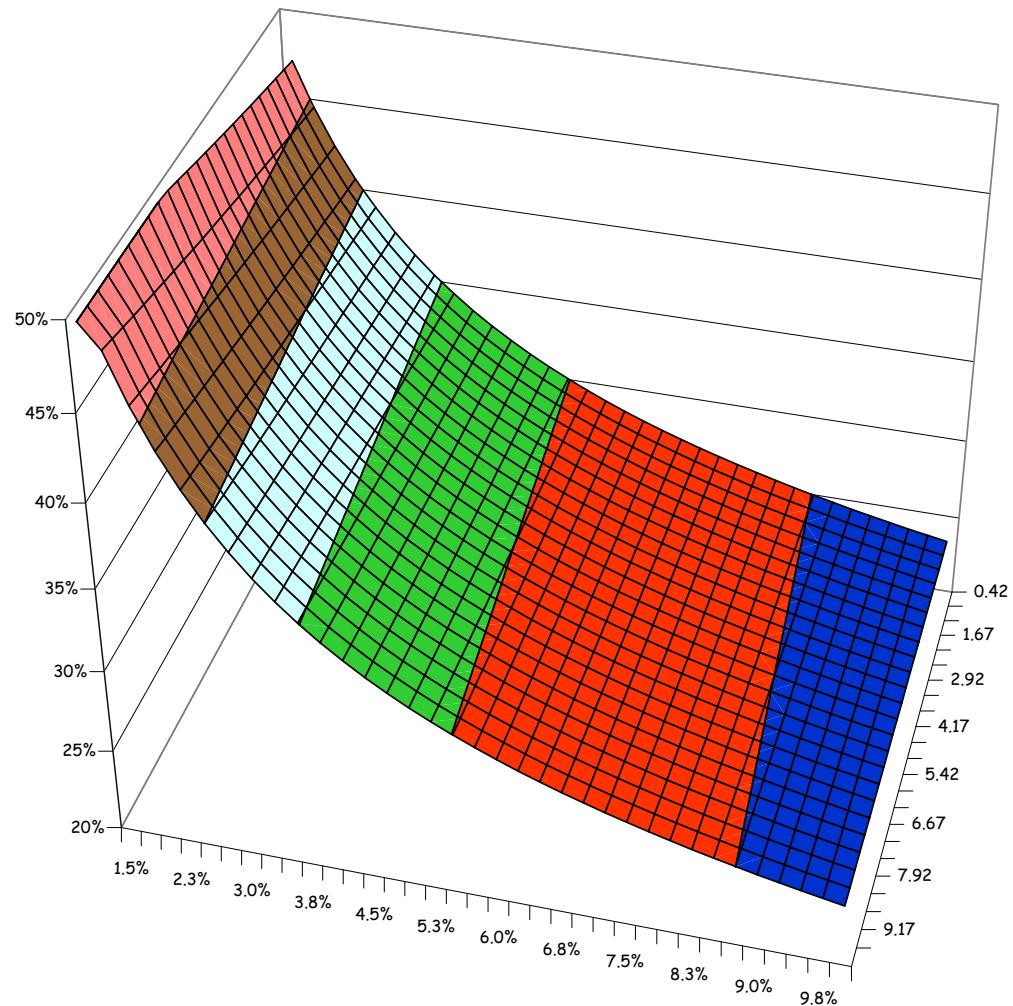


Figure I.3. Implied volatilities for displaced diffusion model with $f = 5\%$, $\beta = 1/4$, $\zeta = 30\%$.

Neither the displaced diffusion nor the constant elasticity of variance model gives control over the smile, i.e. the curvature of the implied volatility profile.

An old option market maker trick to account for uncertainty in the implied volatility away from the money was to take the average of two Black prices for slightly different implied volatilities.

Mathematically, this is consistent with the decomposition of the risk neutral density into the weighted sum of two individually given lognormal probability density functions.

Implying the single associated Black volatility that reproduces those option prices results in an implied volatility smile. It can be shown, however, that this approach cannot reproduce a pronounced skew.

Applying the price mixing trick to the displaced diffusion model gives control over both smile and skew.

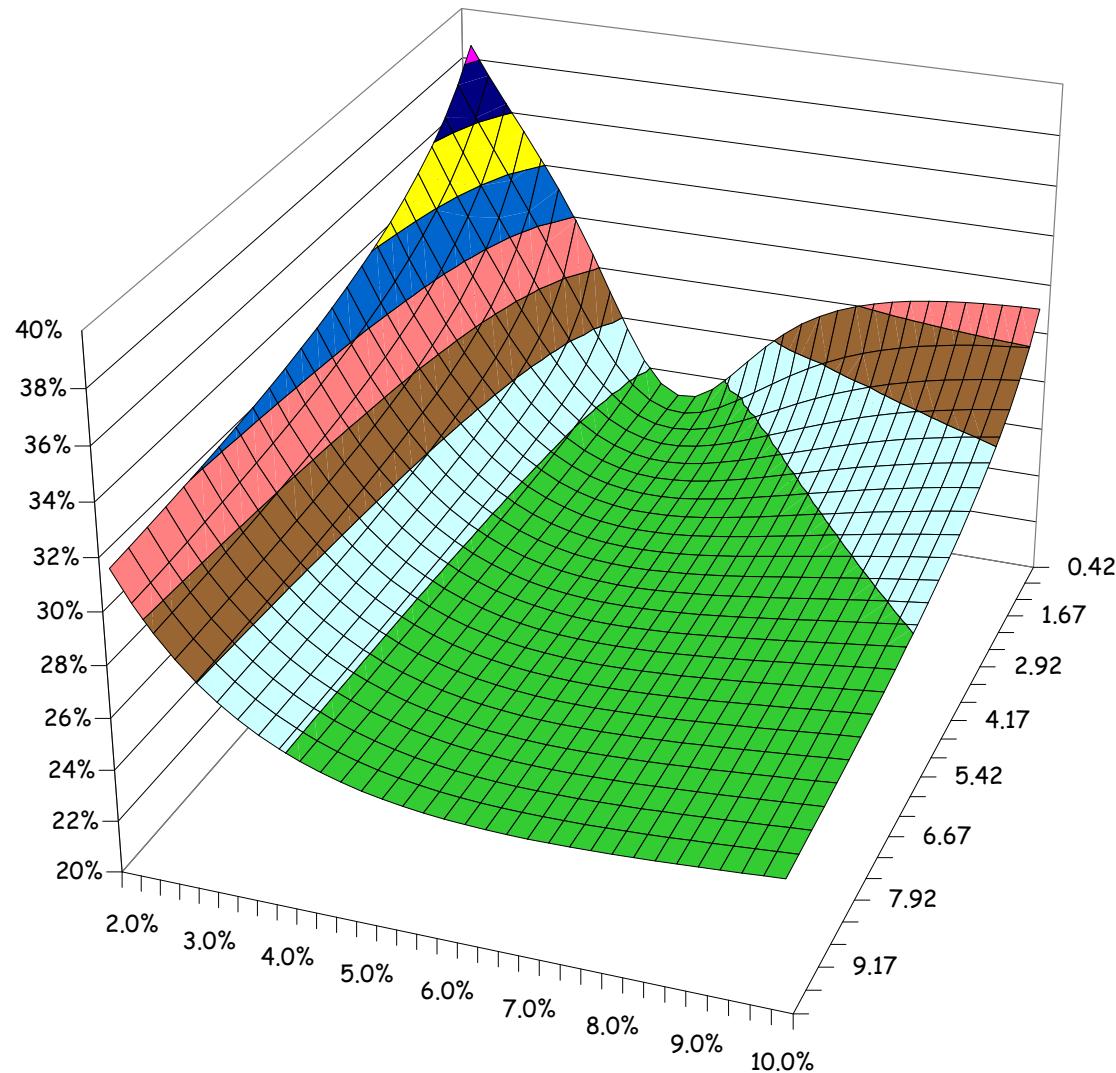


Figure I.4. Implied volatilities for mixtures of displaced diffusion distributions with $f = 5\%$, $\beta_1 = \beta_2 = 3/4$, $\zeta_1 = 35\%$, $\zeta_2 = 15\%$.

CAVEAT: Not all authors and practitioners agree on the meaningfulness of this approach when used literally by mixing the prices from calculations with different model parameters [Pit03c].

However, since the distribution mixing is independent from the underlying spot process, it is possible to reproduce the precise smile from the mixture of two displaced diffusion distributions, both with the same displacement but different diffusion parameters, in the framework of a *local volatility* model.

For a Monte Carlo simulation, this means that the instantaneous (displaced) volatility of anyone forward rate at any time is given by

$$\sigma_{\text{instantaneous}}^{\text{DD}}(f, t) = \sqrt{\frac{w_1 \sigma^{\text{DD (1)}}^2 \psi_1(f, t) + w_2 \sigma^{\text{DD (2)}}^2 \psi_2(f, t)}{w_1 \psi_1(f, t) + w_2 \psi_2(f, t)}}. \quad (19)$$

where $\psi_1(f, t)$ and $\psi_2(f, t)$ are the original distributions that are mixed with weights w_1 and w_2 , respectively.



Alas, this approach, like the genuine CEV model, makes it impossible to accommodate long time steps efficiently.

Another way to model forward Libor rates in order to capture the skew and smile is to make the instantaneous volatility of a displaced diffusion model independently stochastic [Pit03a].

In this presentation, we restrict ourselves to the modelling of the level and the skew of implied volatilities by virtue of the displaced diffusion model.

We will, though, take into account a

term structure

of both

volatility and **correlation**.

II. Derivation of the indirectly stochastic drift

If the forward yield curve is given by n spanning forward rates f_i , whereby the payoff of forward rate agreement i is $f_i \tau_i$ and paid at time t_{i+1} , and a zero coupon bond that pays one currency unit at t_N is used as numéraire,

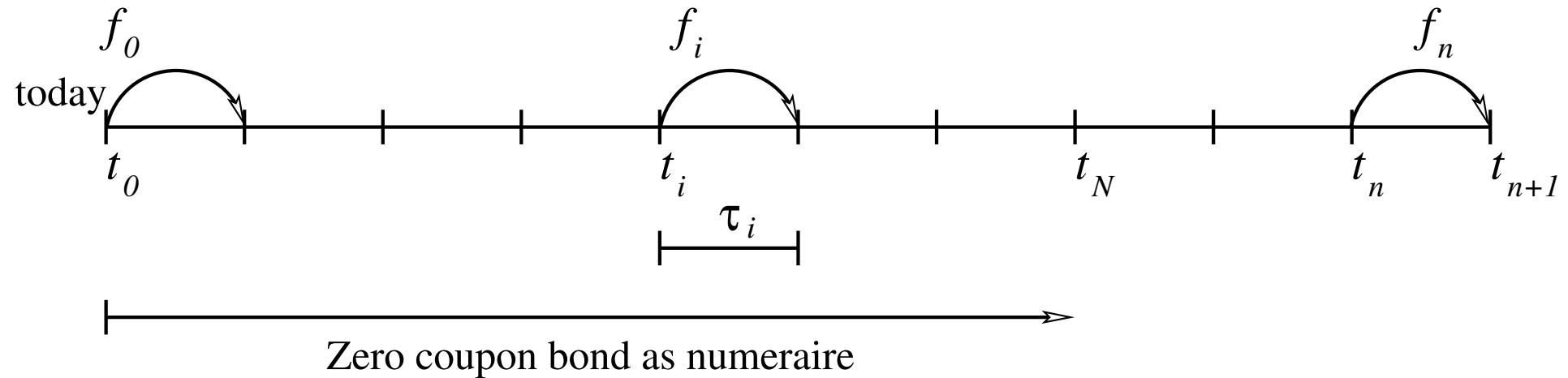


Figure II.1.

then the drifts μ_i in equations (1) and (5) can be calculated from

$$\mathbb{E}_t^{\mathcal{M}_{P_N}} \left[d \left(\frac{f_i P_{i+1}}{P_N} \right) \right] = 0, \quad (20)$$

where $P_i, \forall i = 0, \dots, n$ are the t_i discount bonds and

$$\mathbb{E}_t^{\mathcal{M}_{P_N}}[\cdot]$$

the expectation operator under the equivalent martingale measure induced by the choice of the discount bond P_N as numéraire.

By the fundamental theorem of asset pricing, for the market to be free of arbitrage, all ratios of tradeable assets divided by the numéraire value have to form martingales [HP81], i.e. we also require

$$\mathbb{E}_t^{\mathcal{M}_{P_N}} \left[d \left(\frac{P_i}{P_N} \right) \right] = 0, \quad \forall i = 0, \dots, n, \quad (21)$$

since the discount bonds are assumed to be traded assets. Now, introducing the bond ratio

$$X_i := P_{i+1}/P_N \quad (22)$$

and invoking Itô's formula on equation (20) yields

$$\mathbb{E}_t^{\mathcal{M}_{P_N}} [X_i df_i + f_i dX_i + df_i dX_i] = 0. \quad (23)$$



Since dX is drift-free (it is an asset divided by the numéraire), this reduces to

$$\mathsf{E}_t^{\mathcal{M}_{P_N}}[X_i df_i + dX_i df_i] = \mathsf{E}_t^{\mathcal{M}_{P_N}}[X_i \mu_i f_i dt] + \mathsf{E}_t^{\mathcal{M}_{P_N}}[dX_i df_i] = 0. \quad (24)$$

In the following, we use the instantaneous relative covariance brackets $[a, b]$ defined by the instantaneous drift of the product of the infinitesimal *relative* increments of any two stochastic processes a and b , i.e.

$$[a, b] := \mathsf{E} \left[\frac{da}{a} \frac{db}{b} \right] \Big/ dt. \quad (25)$$

The definition (25) immediately gives us

$$[a, bc] = [a, b] + [a, c] \quad (26)$$

and

$$[a, b/c] = [a, b] - [a, c]. \quad (27)$$

Using this notation, we return to the derivation of the drift of the discrete forward rates. From equation (24), we obtain

$$\mu_i = - \left[f_i, \frac{P_{i+1}}{P_N} \right] \quad (28)$$

which can be evaluated to:

$$\begin{aligned} \left[f_i, \frac{P_{i+1}}{P_N} \right] &= [f_i, P_{i+1}] - [f_i, P_N] \\ &= [f_i, \prod_{k=0}^i (1 + f_k \tau_k)^{-1}] - [f_i, \prod_{k=0}^{N-1} (1 + f_k \tau_k)^{-1}] \\ &= - \sum_{k=0}^i [f_i, 1 + f_k \tau_k] + \sum_{k=0}^{N-1} [f_i, 1 + f_k \tau_k] . \end{aligned} \quad (29)$$



By the definition of the bracket (25) and the dynamics of the individual forward rates (1), each of the terms in the sums on the right hand side of equation (29) can be computed:

$$[f_i, 1 + f_k \tau_k] = \frac{f_k \tau_k}{1 + f_k \tau_k} \sigma_i \sigma_k \rho_{ik} \quad (30)$$

Finally, cancellation of summation terms leads to the drift formulæ

$$\mu_i^{(t_N)}(f(t), t) = \begin{cases} -\sigma_i \sum_{k=i+1}^{N-1} \frac{f_k(t) \tau_k}{1 + f_k(t) \tau_k} \sigma_k \rho_{ik} & \text{for } i < N-1 \\ 0 & \text{for } i = N-1 \\ \sigma_i \sum_{k=N}^i \frac{f_k(t) \tau_k}{1 + f_k(t) \tau_k} \sigma_k \rho_{ik} & \text{for } i \geq N \end{cases} \quad (31)$$

Note that this means that the drift of all of the forward rates but one are indirectly stochastic, i.e. it is stochastic due to its explicit dependence on the stochastic forward rates themselves. When $i = N-1$, i.e. for a **drift-free** forward rate f_i , we call the numéraire associated with the pricing measure the **natural numéraire** of the forward rate f_i .

Computing the drift terms in different measures is readily accomplished by evaluation of the effect of the change from

measure \mathcal{M}_A induced by numéraire N_A

to

measure \mathcal{M}_B induced by numéraire N_B

on the driving Wiener processes:

$$dW^{\mathcal{M}_B} = E \left[d \ln \left(\frac{d\mathcal{M}_B}{d\mathcal{M}_A} \right) \cdot dW^{\mathcal{M}_A} \right] + dW^{\mathcal{M}_A} \quad (32)$$

with the Radon-Nikodym derivative

$$\frac{d\mathcal{M}_B}{d\mathcal{M}_A} = \frac{N_B}{N_A} \cdot \frac{N_A(0)}{N_B(0)} . \quad (33)$$

III. Leaving the canon

Not all products can be described by the aid of canonical forward rates alone.

In practice, a Libor market model implementation has to cope with many intermediate cashflows, with settlement delays, fixing conventions, and many other idiosyncrasies of the interest rates markets.

A flexible Libor market model may have to handle stub discount factors that cover only part of the associated discrete forward rate's accrual period.

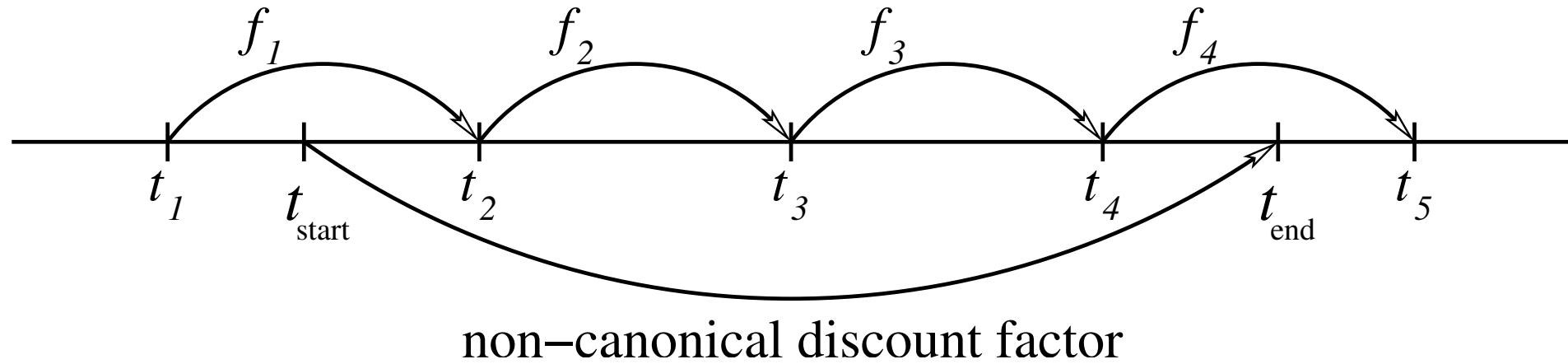


Figure III.1.

It is difficult to construct non-canonical discount factors from a given set of discrete forward rates in a *completely* arbitrage-free manner.

However, in practice, it is usually sufficient to choose an approximate interpolation rule such that the residual error is well below the levels where arbitrage could be enforced.

It is also important to remember that the numerical evaluation of any complex deal with a Libor market model is ultimately still subject to inevitable errors resulting from the calculation scheme: Monte Carlo simulations, non-recombining trees, or recombining trees with their own drift approximation problems.

In this context it may not be surprising that the following discount factor interpolation approach is highly accurate for practical purposes.

Define the interval indicator

$$i[s] := \max (i \mid s \geq t_i) \quad (34)$$

and the modified accrual factor

$$\theta[s] := t_{i[s]+1} - s . \quad (35)$$

Also define $f[s]$ as the discrete interest rate for the accrual period from time s to the next canonical time $t_{i[s]+1}$.

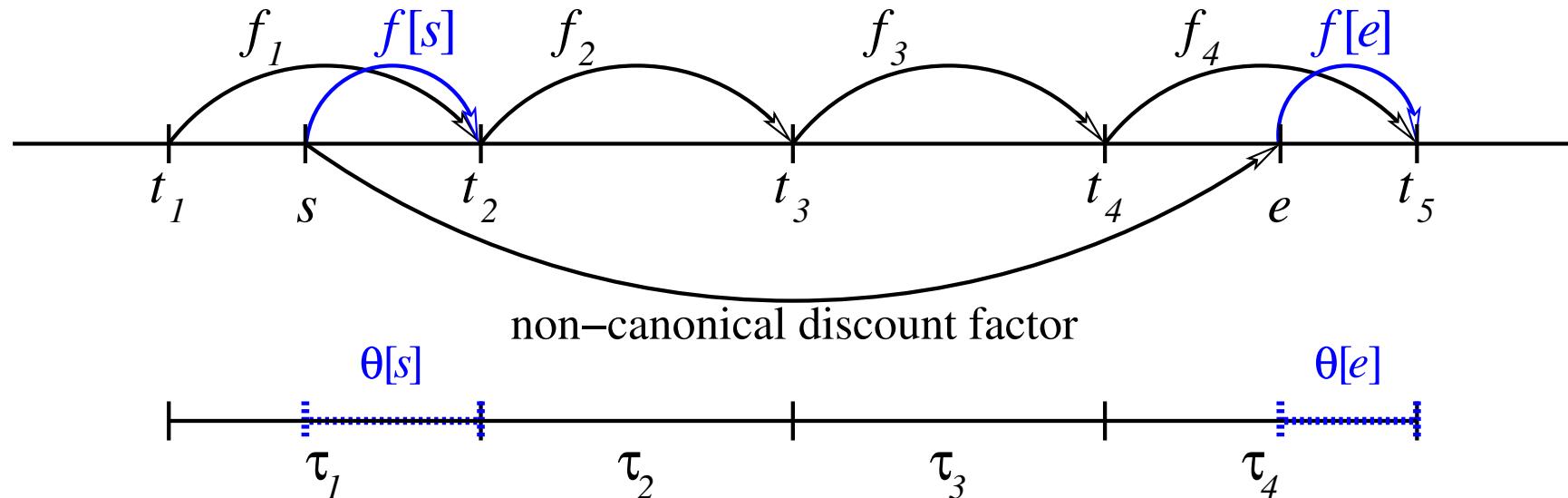


Figure III.2.

A forward discount factor thus decomposes into

$$P[s, e](t) = \frac{P(t, e)}{P(t, s)} = \frac{1 + f[e](t) \cdot \theta[e]}{1 + f[s](t) \cdot \theta[s]} \prod_{j=i[s]+1}^{i[e]+1} \frac{1}{1 + f_j(t) \tau_j}. \quad (36)$$

An approximation that for practical purposes suffices is to set

$$f[s](t) = \gamma[s] \cdot f_{i[s]}(t) \quad (37)$$

wherein the constant $\gamma[s]$ takes care of the fine structure of the yield curve in between canonical dates as seen at inception:

$$\gamma[s] = \frac{f[s](0)}{f_{i[s]}(0)} = \frac{\left(\frac{P(0, s)}{P(0, t_{i[s]+1})} - 1 \right) / \theta[s]}{f_{i[s]}(0)}. \quad (38)$$

This approach essentially approximates the yield curve dynamics in between canonical times by a one factor model.

Note: Since we decompose all stub rates into pieces depending directly only on a canonical rate of the same natural measure, it is straightforward to extend this model to allow for

stochastic stub rates

in between $t_{i[s]}$ and s by simply continuing the nominal state variable $f_{i[s]}$ as a stochastic process all the way until $t_{i[s]+1}$ instead of freezing it at $t_{i[s]}$.

The generic interpolation rule (37) then takes care of the evaluation of $f[s]$ both for

$$t < t_{i[s]}$$

and for

$$t_{i[s]} \leq t \leq t_{i[s]+1} .$$



IV. Futures convexity corrections in the Libor market model

The value of a future contract on a forward rate fixing at time t_i is given by the expectation of f_i in the spot measure (also known as the measure associated with the chosen numéraire being the continuously rolled up money market account)²:

$$\hat{f}_j = \mathbb{E}^*[f_j(t_j)] \quad (39)$$

In the discretely rolled up spot measure, we have

$$\frac{d(f_j + s_j)}{f_j + s_j} = \mu_j^{*\text{DD}}(f, s, t) dt + \sigma_j^{\text{DD}} dW_j^* \quad (40)$$

with

$$\mu_j^{*\text{DD}}(f, s, t) = \sigma_j^{\text{DD}}(t) \cdot \sum_{k=i[t]+1}^j \frac{(f_k(t) + s_k)\tau_k}{1 + f_k(t)\tau_k} \sigma_k^{\text{DD}}(t) \rho_{jk}(t) . \quad (41)$$

²For a proof of this result, which was first published in [CIR81], see, for instance, theorem 3.7 in [KS98].



The lowest order futures convexity correction for the j -th forward rate is

$$E^*[f_j(t_j)] \approx (f_j(0) + s_j) \cdot e^{\int_0^{t_j} \mu_j^*(f(0), s, t) dt} - s_j. \quad (42)$$

For the Libor market model, the above approximation usually fails quite dramatically due to the fact that the drift expression $\mu_i^*(f, t)$ is itself stochastic.

For the lognormal dynamics (1), Matsumoto [Mat01] suggests the approximation

$$E^*[f_j(t_j)] \approx f_j(0) \left(1 + \varepsilon^{(\text{Matsumoto})} \right) \quad (43)$$

with

$$\varepsilon^{(\text{Matsumoto})} := P_{j+1}(0) \cdot \sum_{k=0}^j f_k(0) \tau_k \cdot e^{\int_0^{t_j} \mu_k^{(t_j+1)}(f(0), t) dt} \left(e^{\int_0^{t_j} \sigma_j(t) \sigma_k(t) \rho_{jk}(t) dt} - 1 \right). \quad (44)$$



and $\mu_k^{(t_{j+1})}(f(0), t)$ defined as in equation (31).

A better approximation can be derived by the aid of the technique of iterated substitutions (also known as Itô-Taylor expansion) [JK05]:

$$\mathbb{E}^*[f_j(t_j)] \approx \frac{f_j(0)}{\beta_j} \left[\frac{1}{\beta_j} \left(e^{\beta_j \varepsilon_j^{(n)}} + \beta_j - 1 \right) + \beta_j - 1 \right] \quad (45)$$

$$\beta_j := \frac{f_j(0)}{f_j(0) + s_j} \quad (46)$$

$$\varepsilon_j^{(n)} := \sum_{k=1}^n \sum_{l=0}^j \frac{1}{k!} \frac{(f_l(0) + s_l) \tau_l}{(1 + f_l(0) \tau_l)^k} \left(\int_0^{t_j} \sigma_j(t) \sigma_l(t) \rho_{jl}(t) dt \right)^k \quad (47)$$

$$+ \frac{3}{2} \sum_{k=2}^n \frac{1}{k!} \left(\sum_{l=0}^j \frac{(f_l(0) + s_l) \tau_l}{1 + f_l(0) \tau_l} \int_0^{t_j} \sigma_i(t) \sigma_l(t) \rho_{jl}(t) dt \right)^k. \quad (48)$$

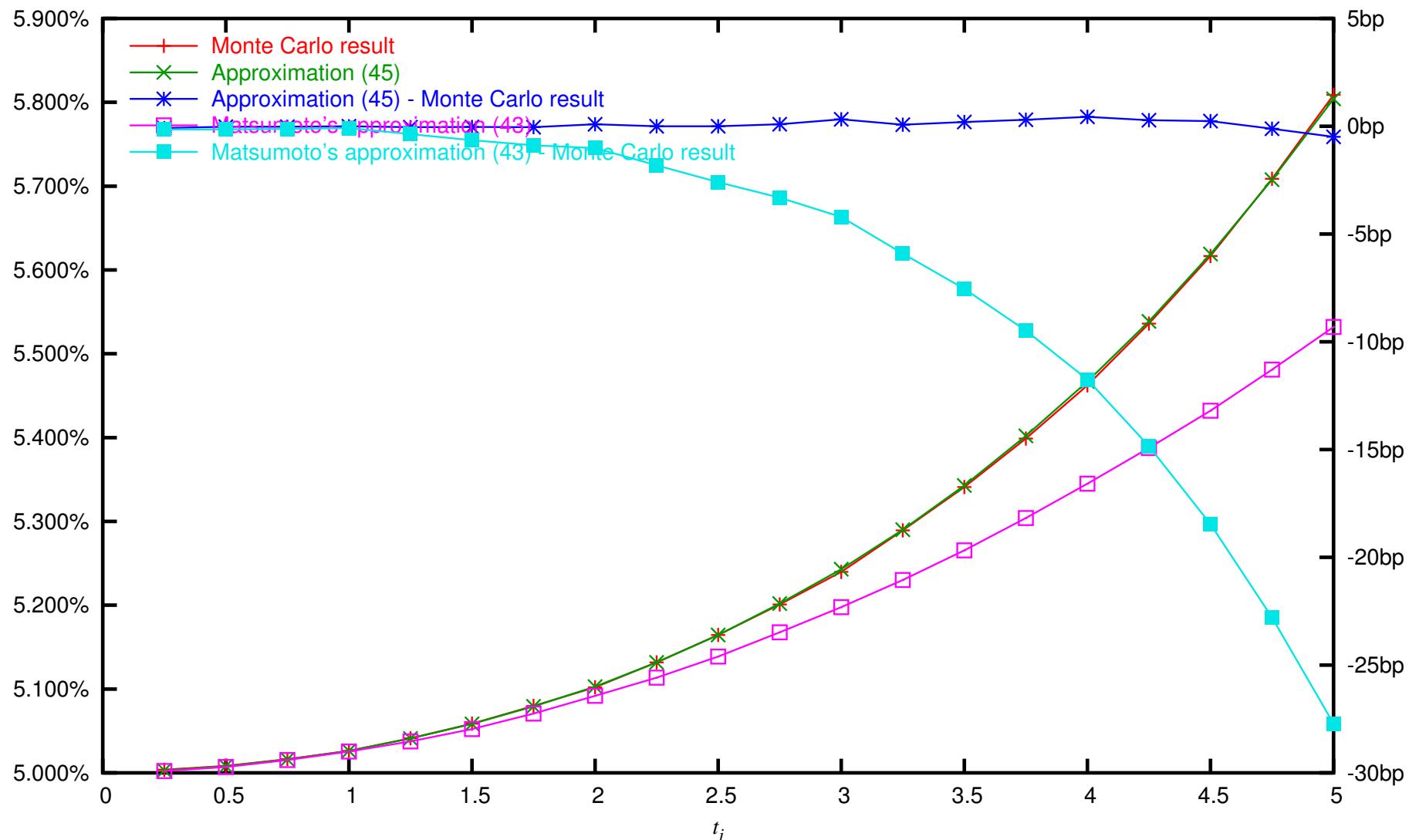


Figure IV.1. Numerical and analytical results for $\rho_{jl} = 1$, $f_j = 5\%$, $\hat{\sigma}_j = 40\%$, and $\beta_j = 1$.

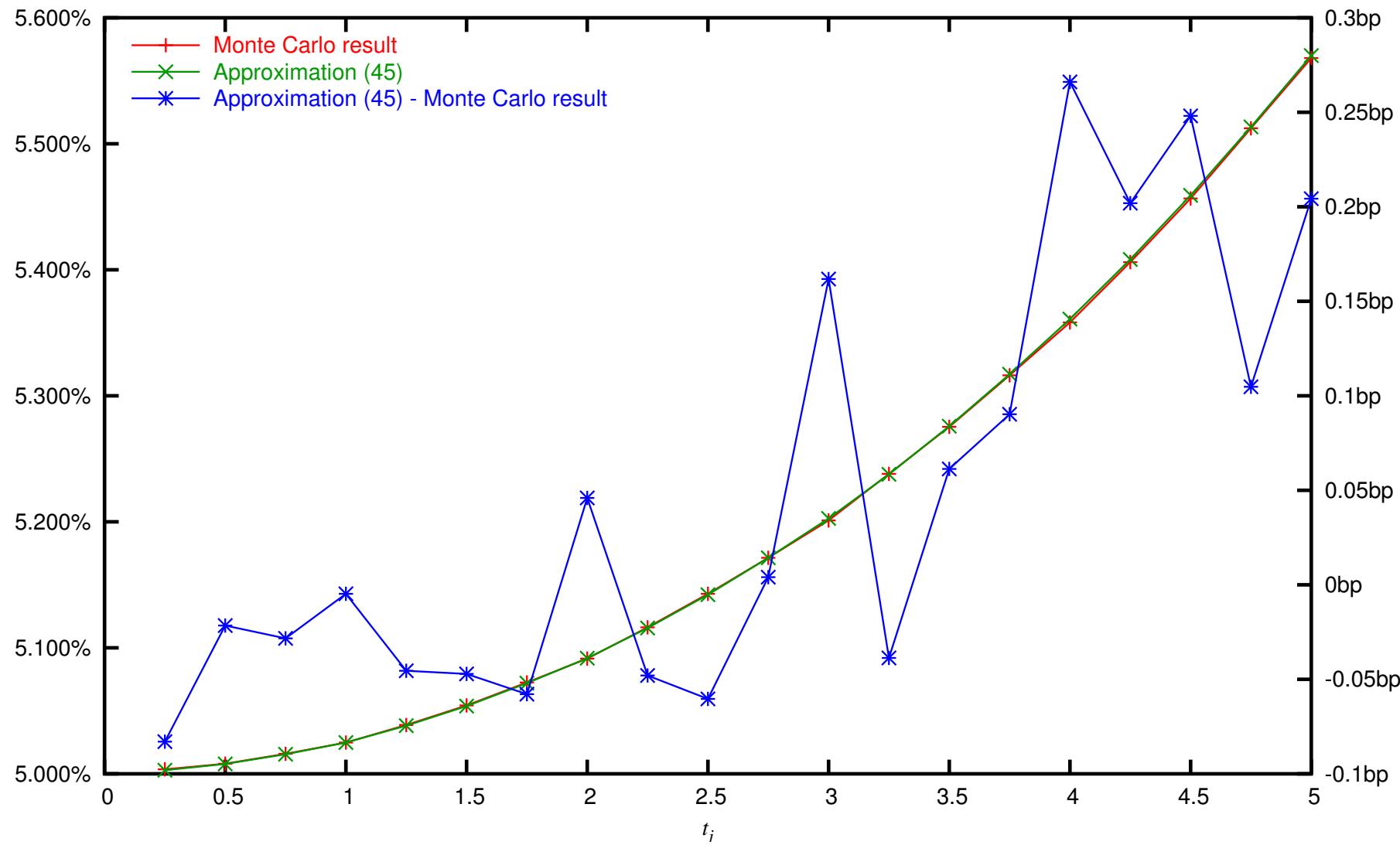


Figure IV.2. Numerical and analytical results for $\rho_{jl} = 1$, $f_j = 5\%$, $\hat{\sigma}_j = 40\%$, and $\beta_j = 1/2$.

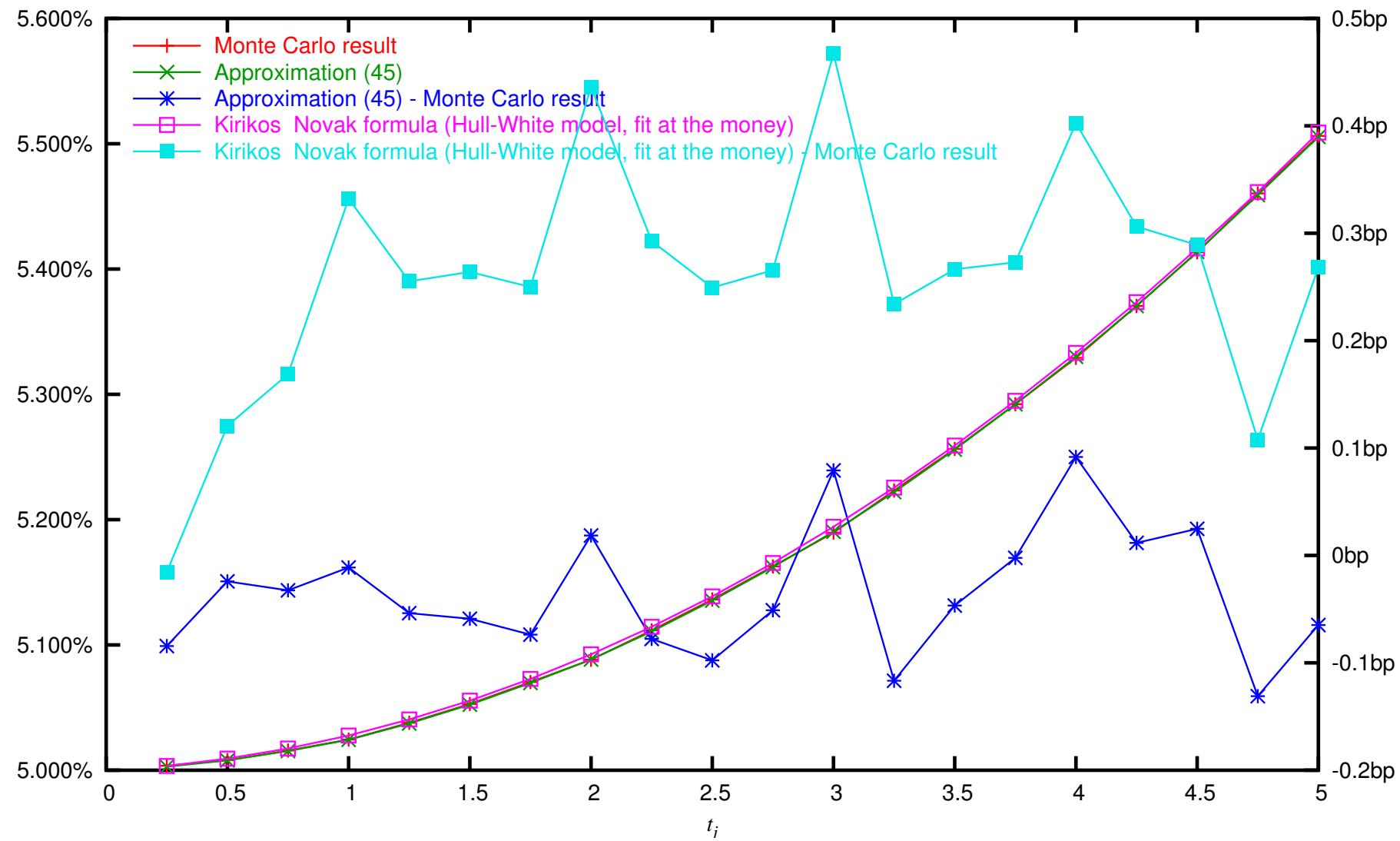


Figure IV.3. Numerical and analytical results for $\rho_{jl} = 1$, $f_j = 5\%$, $\hat{\sigma}_j = 40\%$, and $\beta_j = 0.036$. The Hull-White coefficient used for the Kirikos-Novak formula [KN97] was $\sigma_{HW} = 1.91\%$.

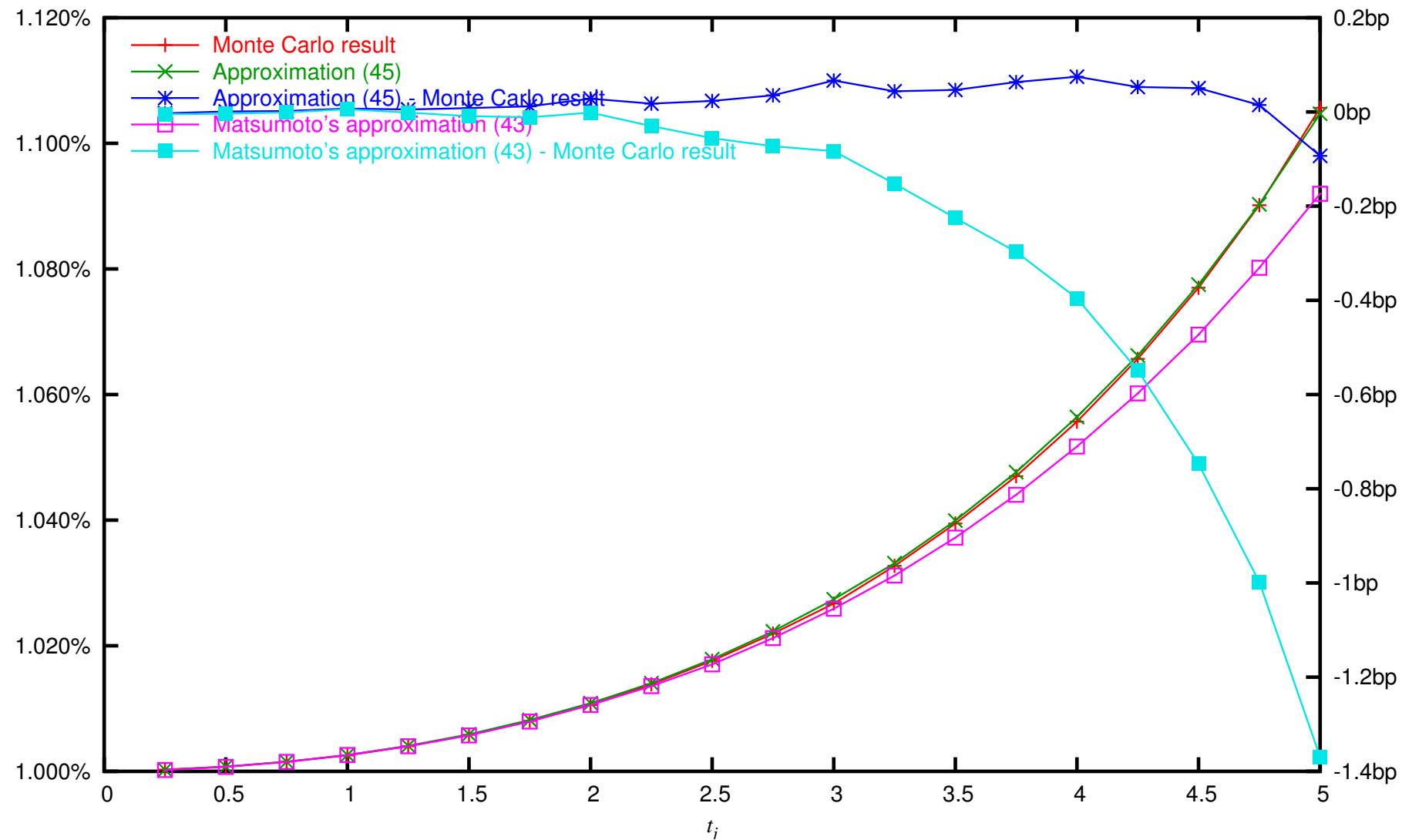


Figure IV.4. Numerical and analytical results for $\rho_{jl} = 1$, $f_j = 1\%$, $\hat{\sigma}_j = 60\%$, and $\beta_j = 1$.

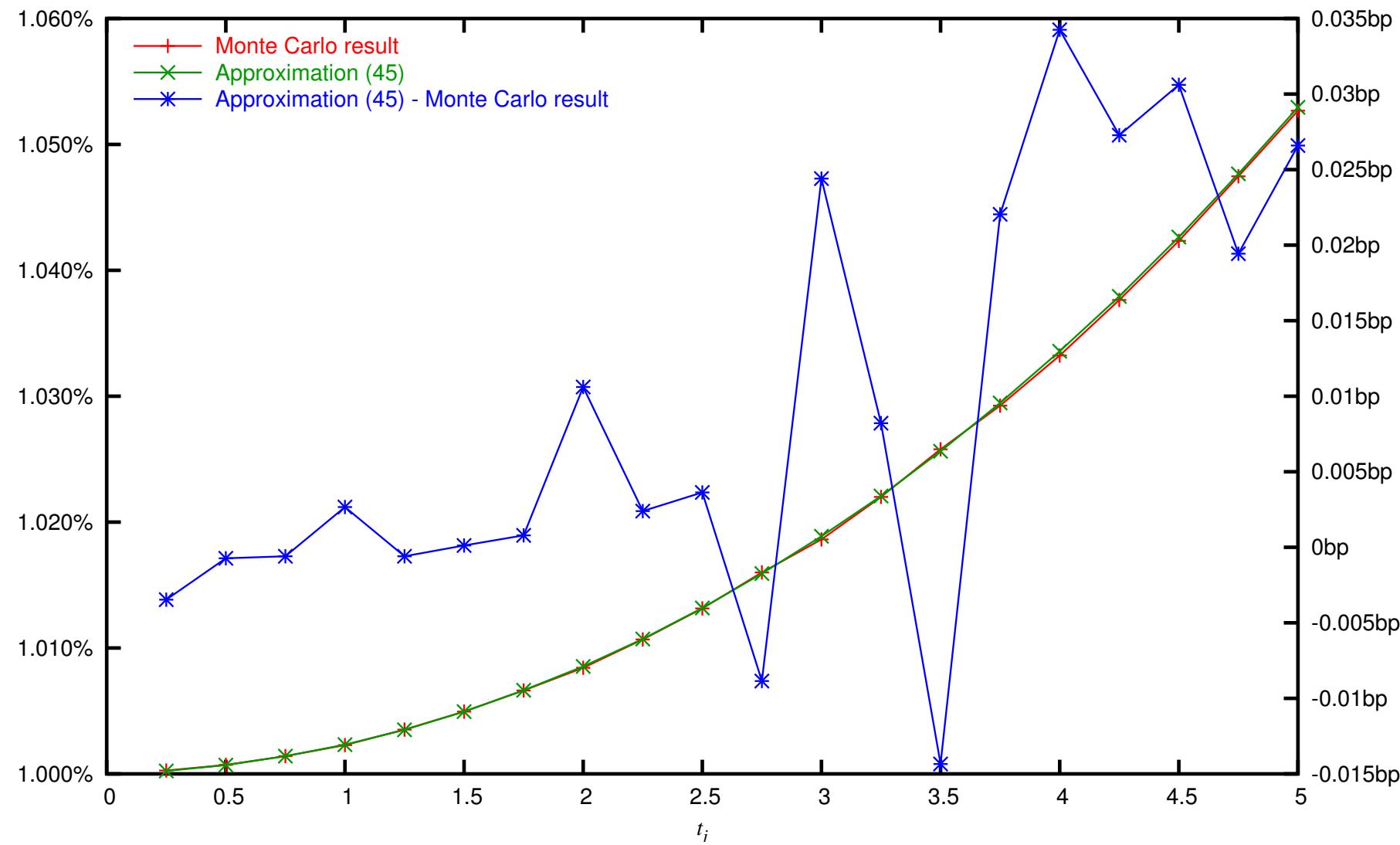


Figure IV.5. Numerical and analytical results for $\rho_{jl} = 1$, $f_j = 1\%$, $\hat{\sigma}_j = 60\%$, and $\beta_j = 1/2$.

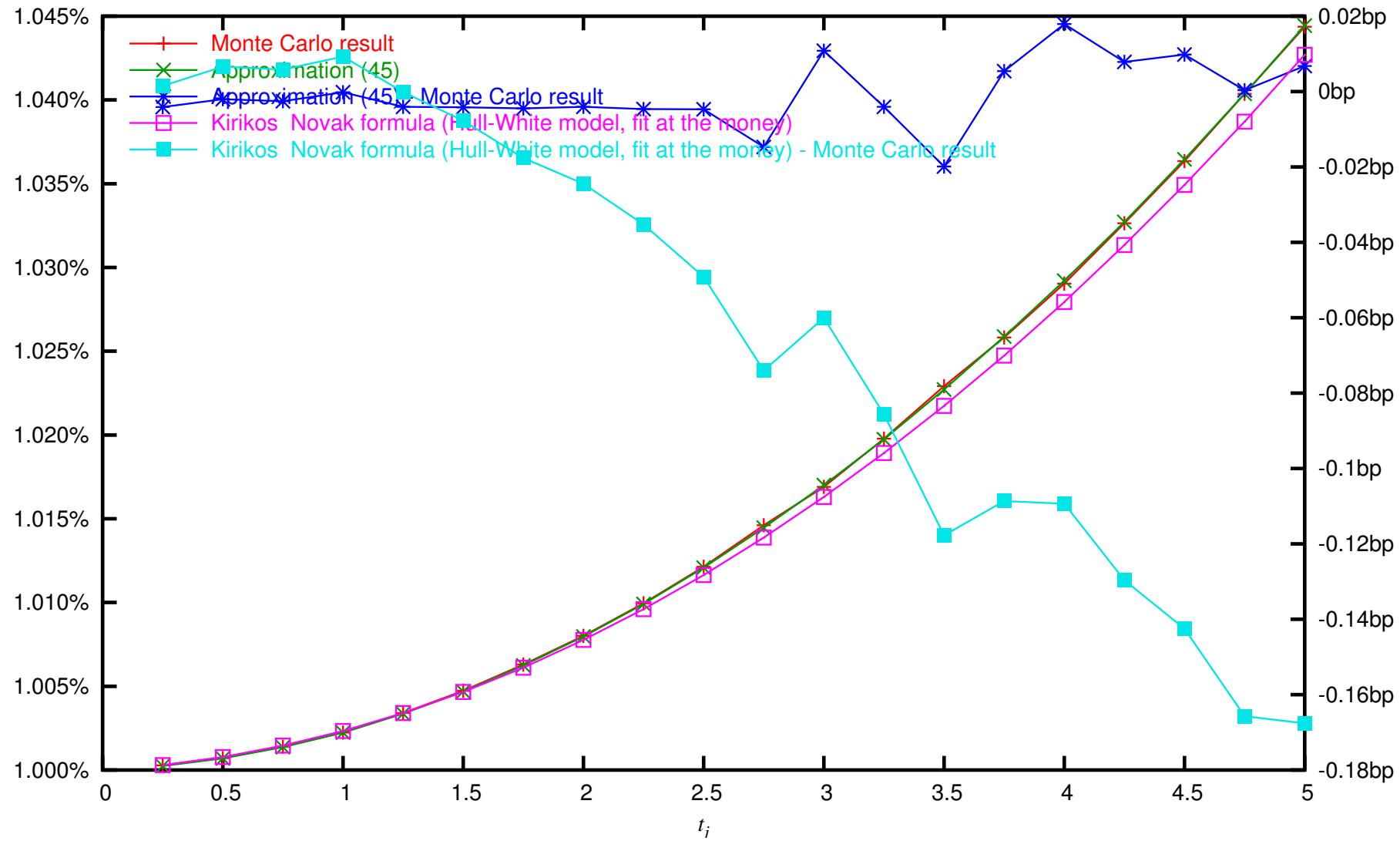


Figure IV.6. Numerical and analytical results for $\rho_{jl} = 1$, $f_j = 1\%$, $\hat{\sigma}_j = 60\%$, and $\beta_j = 0.036$. The Hull-White coefficient used for the Kirikos-Novak formula [KN97] was $\sigma_{HW} = 0.56\%$.

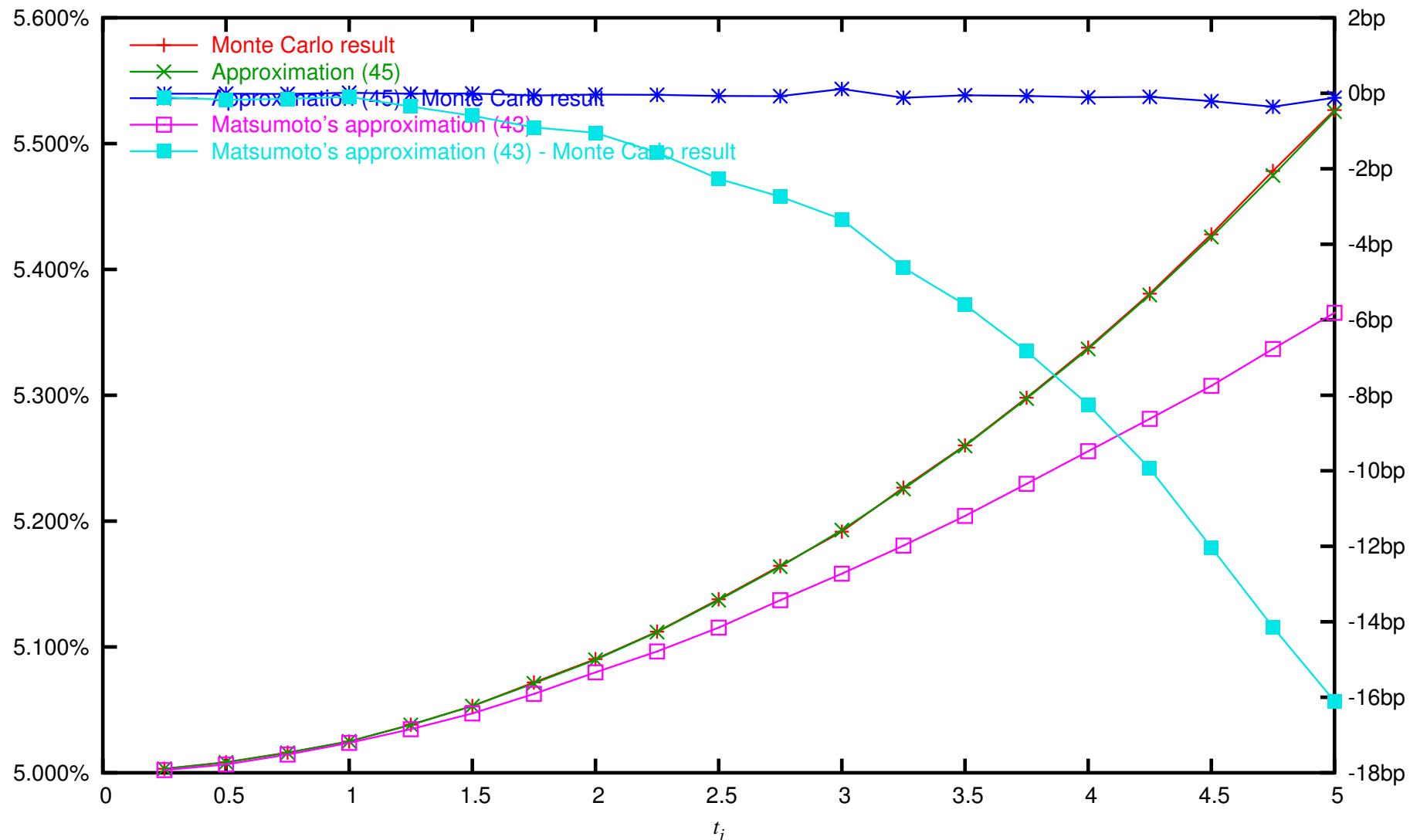


Figure IV.7. Numerical and analytical results for $\rho_{jl} = e^{-|t_i - t_j|/4}$, $f_j = 5\%$, $\hat{\sigma}_j = 40\%$, and $\beta_j = 1$.

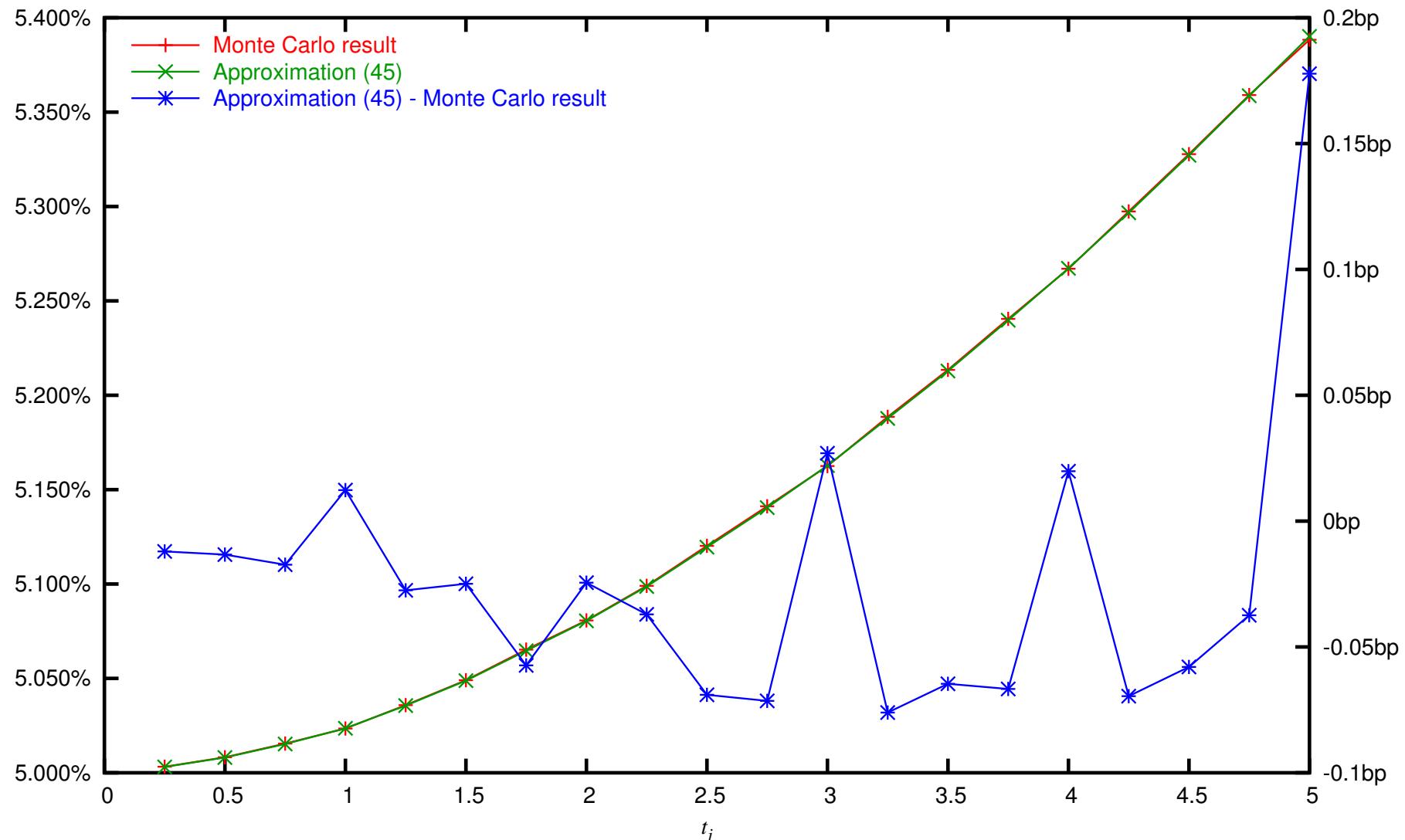


Figure IV.8. Numerical and analytical results for $\rho_{jl} = e^{-|t_i - t_j|/4}$, $f_j = 5\%$, $\hat{\sigma}_j = 40\%$, and $\beta_j = 1/2$.

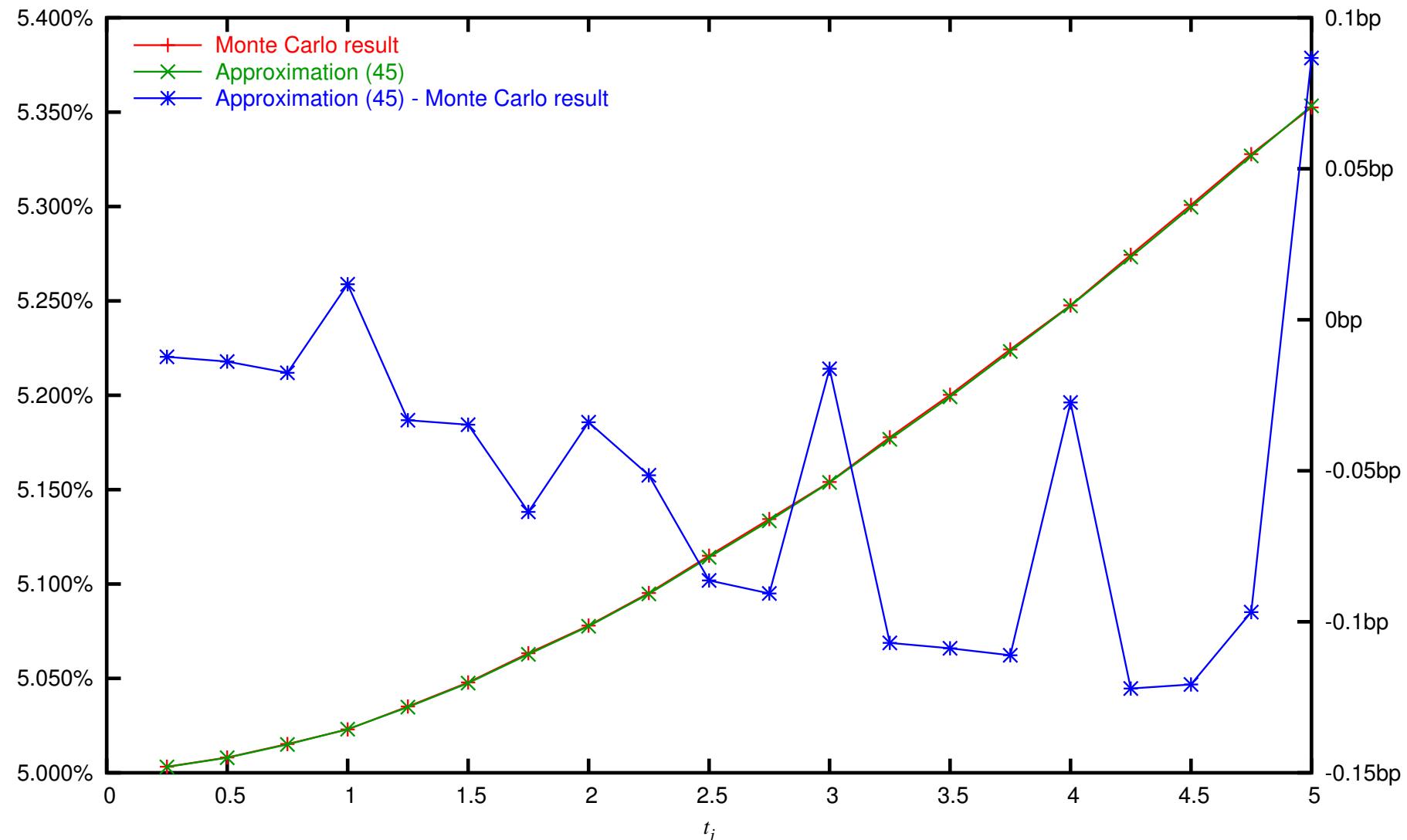


Figure IV.9. Numerical and analytical results for $\rho_{jl} = e^{-|t_i - t_j|/4}$, $f_j = 5\%$, $\hat{\sigma}_j = 40\%$, and $\beta_j = 0.036$.

V. Speed is everything — the predictor-corrector scheme

In order to price an exotic interest rate derivative, we need to evolve the set of forward rates f from its present values into the future.

The drift terms given by equation (41) are clearly state-dependent and thus indirectly stochastic which forces us to use a numerical scheme to solve equation (40) along any one path.

A simple explicit Euler scheme for the state variables $x_i := f_i + s_i$ is

$$x_i^{\text{Euler}}(t + \Delta t) = x_i(t) + x_i(t) \cdot \mu_i(t) \Delta t + x_i(t) \cdot \sum_{j=1}^m \hat{a}_{ij}(t) z_j \sqrt{\Delta t} \quad (49)$$

with z_j being m independent normal variates.

This would imply that we approximate the drift and volatilities as constant over the time step $t \rightarrow t + \Delta t$, not even taking into account any term structure of volatility.

Moreover, this scheme effectively means that we are using a normal distribution for the evolution of the forward rates over this time step.

Whilst we may agree to the approximation of a piecewise constant (in time) drift coefficient μ_i , the normal distribution may be undesirable, especially if we envisage to use large time steps Δt for reasons of computational efficiency.

However, when we assume piecewise constant drift, we might as well carry out the integration over the time step Δt analytically and use the scheme

$$x_i^{\text{Constant drift}}(\mathbf{f}(t), t + \Delta t) = x_i(t) \cdot e^{\mu_i(\mathbf{f}(t), t) \Delta t - \frac{1}{2} \hat{c}_{ii} + \sum_{j=1}^m \hat{a}_{ij} z_j} \quad (50)$$

with

$$\hat{c}_{ij} = \int_t^{t+\Delta t} \sigma_i(u) \sigma_j(u) \rho_{ij}(u) du \quad (51)$$

and \hat{A} being the spectral square root³ of \hat{C} :

$$\hat{C} = \hat{A} \cdot \hat{A}^\top. \quad (52)$$

This is essentially an Euler scheme in logarithmic coordinates.

The above procedure works very well as long as the time steps Δt are not too long and is widely used and also referred to in publications [And00, GZ99].

Since the drift term appearing in the exponential function in equation (50) is in some sense a stochastic quantity itself, we will begin to notice that we are ignoring Jensen's inequality when the term $\mu_i \Delta t$ becomes large enough.

This happens when we choose a big step Δt , or the forward rates themselves or their volatility are large. Therefore, we should use a *hybrid predictor-corrector* method which models *only the drift* as indirectly stochastic.

A method that works very well in practice is as follows.

³also known as *Schur decomposition*

- Given a current evolution of the yield curve denoted by $\mathbf{x}(t)$, we calculate the predicted solution $\mathbf{x}^{\text{Constant drift}}(\mathbf{x}(t), t + \Delta t)$ using one m -dimensional normal variate draw z following equation (50).
- We recalculate the drift using this evolved yield curve. The predictor-corrector approximation $\tilde{\mu}_i$ for the drift is then given by the average of these two calculated drifts, i.e.

$$\begin{aligned}\tilde{\mu}_i(\mathbf{x}(t), t \rightarrow t + \Delta t) = & \frac{1}{2} \left\{ \mu_i(\mathbf{x}(t), t) \right. \\ & \left. + \mu_i(\mathbf{x}^{\text{Constant drift}}(\mathbf{x}(t), t + \Delta t), t) \right\} .\end{aligned}\tag{53}$$

- The predictor-corrector evolution is given by

$$x_i^{\text{Predictor-corrector}}(\mathbf{x}(t), t + \Delta t) = x_i(t) \cdot e^{\tilde{\mu}_i(\mathbf{x}(t), t \rightarrow t + \Delta t) \Delta t - \frac{1}{2} \hat{c}_{ii} + \sum_{j=1}^m \hat{a}_{ij} z_j}\tag{54}$$

wherein we re-use the same normal variate draw z , i.e. we only correct the drift of the predicted solution.

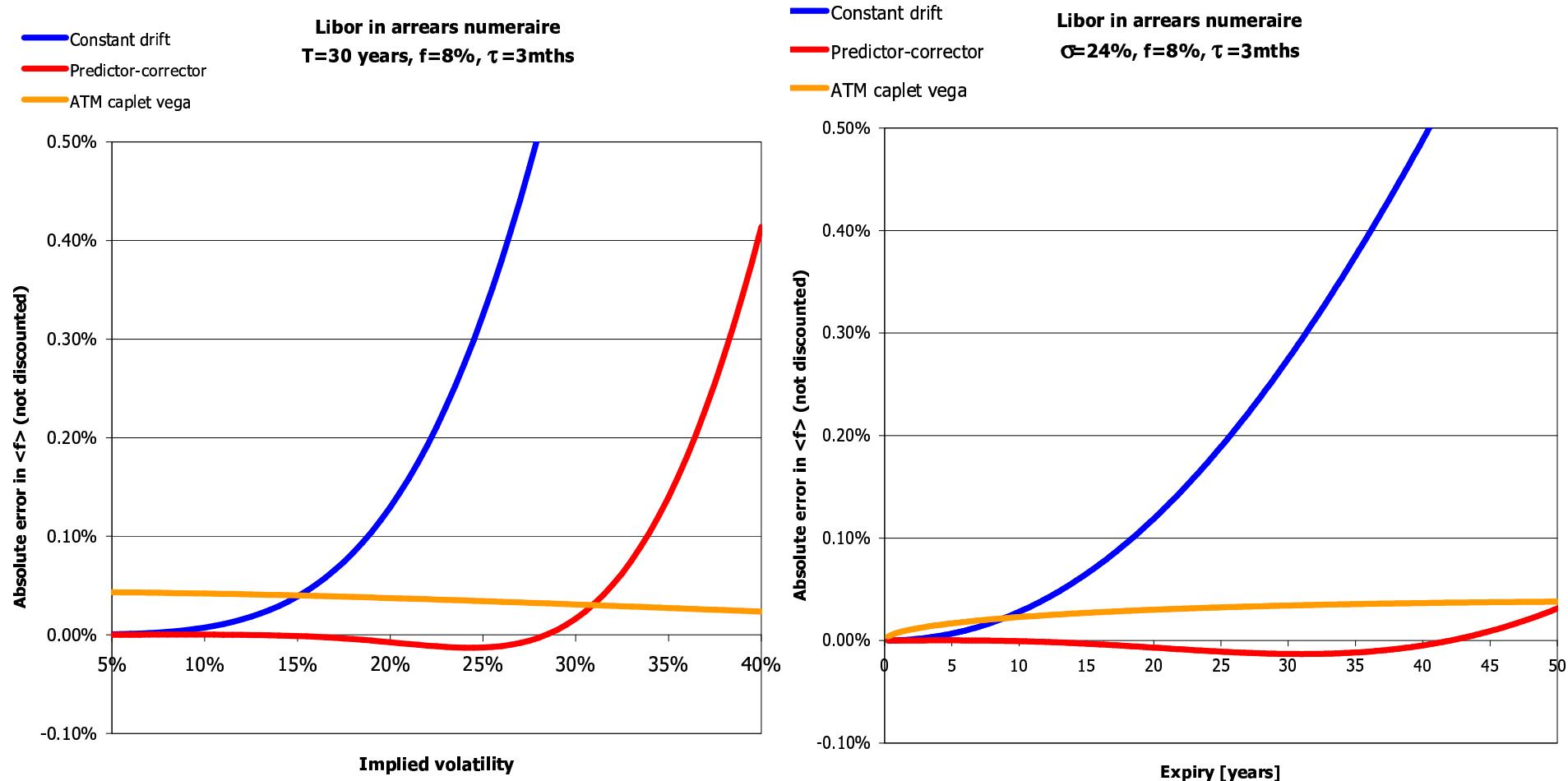


Figure V.1. The stability of the predictor-corrector drift method as a function of volatility level (left) and time to expiry (right) for the Libor-in-arrears convexity.

→ **The predictor-corrector drift approximation is highly accurate!**

VI. Parametrisation of correlation and volatility backbone

Stable calibration of any market model relies on the specification of a robust yet flexible *reference volatility structure*. We call a specification of instantaneous volatility *time-homogeneous* or *stationary* if the volatility of any forward rate f_T that will fix at time T depends on calendar time t only in terms of $T - t$, i.e.

$$\sigma_T(t) = \sigma(T - t) . \quad (55)$$

One cannot fit many market prices with this strict assumption.

In fact, there are frequently good economic reasons why time-homogeneity may not be given for the term structure of instantaneous volatility of forward rates.

In practice, we may want to use an initial parametrised fit in order to find the values a , b , c , and d , such that (only) the caplet implied volatilities resulting from the instantaneous FRA volatility

$$\sigma_i(t) = k_i \left[(a + b \cdot (t_i - t)) \cdot e^{-c \cdot (t_i - t)} + d \right] \cdot \mathbf{1}_{\{t < t_i\}} \quad (56)$$

are perfectly matched to the caplet volatility entries in the swaption matrix with all of the adjustment coefficients k_i being as near to 1 as possible. Note that this is only to obtain a reasonable skeleton for the term structure of FRA volatility. The so obtained parameters a , b , c , and d then determine the *reference* or *skeleton* term structure of instantaneous volatility

$$\sigma_T^{\text{reference}}(t) = \left[(a + b \cdot (T - t)) \cdot e^{-c \cdot (T - t)} + d \right] \cdot \mathbf{1}_{\{t < T\}}. \quad (57)$$

As for the instantaneous correlation between forward rates, a parametrisation that is economically, econometrically, and analytically appealing is the functional form

$$\rho_{ij} = e^{-\beta \cdot (t_i - t_j)} \quad (58)$$

with t_i and t_j , as before, being the expiry times of caplets $\#i$ and $\#j$. A reasonable value for the overall correlation coefficient is $\beta \approx 0.1$.



Since the instantaneous correlation function doesn't actually depend on calendar time t , integrated FRA/FRA covariances can be computed analytically:

$$\begin{aligned} \int \rho_{ij} \sigma_i(t) \sigma_j(t) dt &= e^{-\beta |\delta_i - \delta_j|} \cdot k_i k_j \cdot \frac{1}{4c^3} \cdot \\ &\quad \cdot \left(4ac^2 d [e^{c\delta_j} + e^{c\delta_i}] + 4c^3 d^2 t \right. \\ &\quad - 4bcde^{c\delta_i} [c\delta_i - 1] - 4bcde^{c\delta_j} [c\delta_j - 1] \\ &\quad + e^{c(\delta_i + \delta_j)} \left(2a^2 c^2 + 2abc [1 - c(\delta_i + \delta_j)] \right. \\ &\quad \left. \left. + b^2 [1 + 2c^2 \delta_i \delta_j - c(\delta_i + \delta_j)] \right) \right) \end{aligned} \quad (59)$$

with $\delta_i := t - t_i$ and $\delta_j := t - t_j$.

The quality of a reference fit of the implied volatilities consistent with equation (56) for a typical yield curve and caplet market both in EUR and USD is usually quite good:

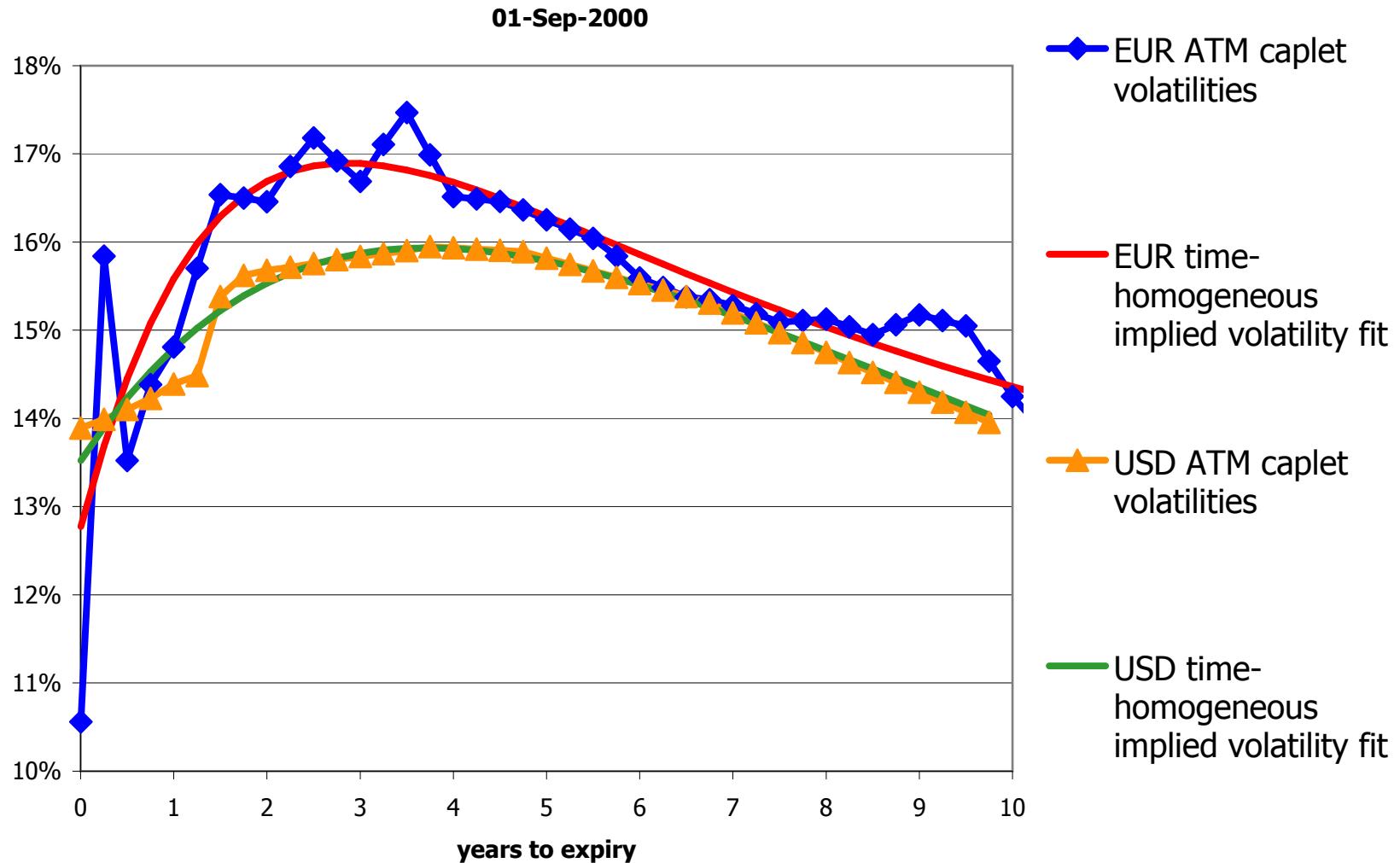


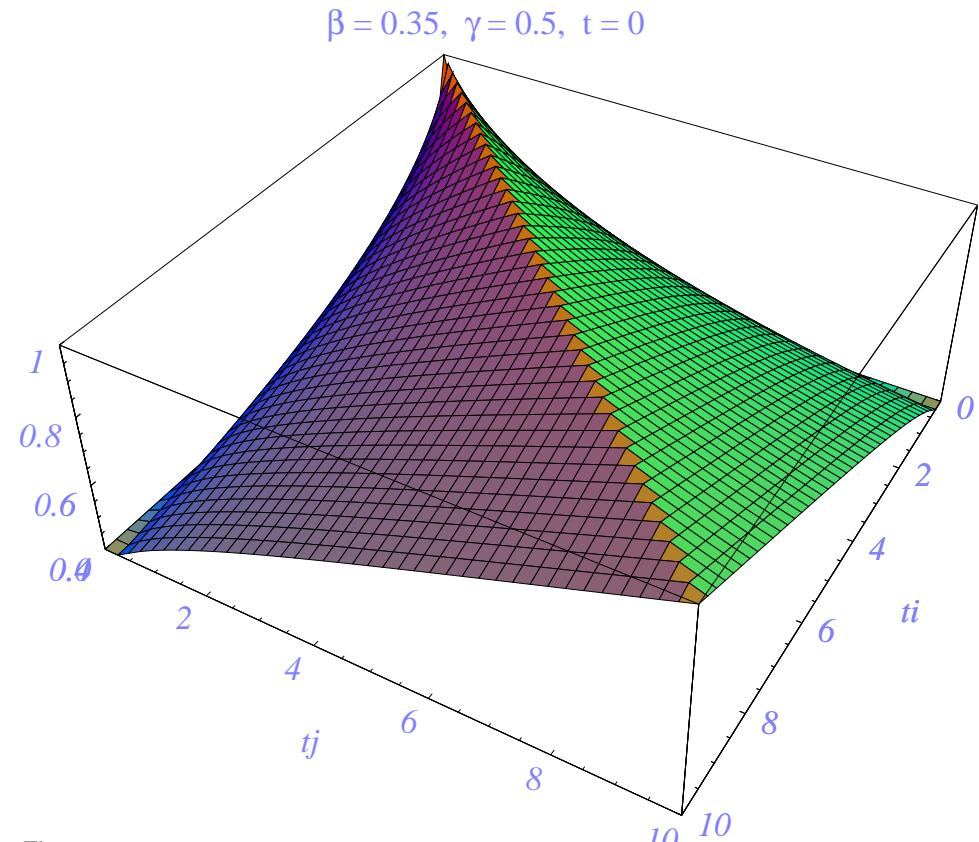
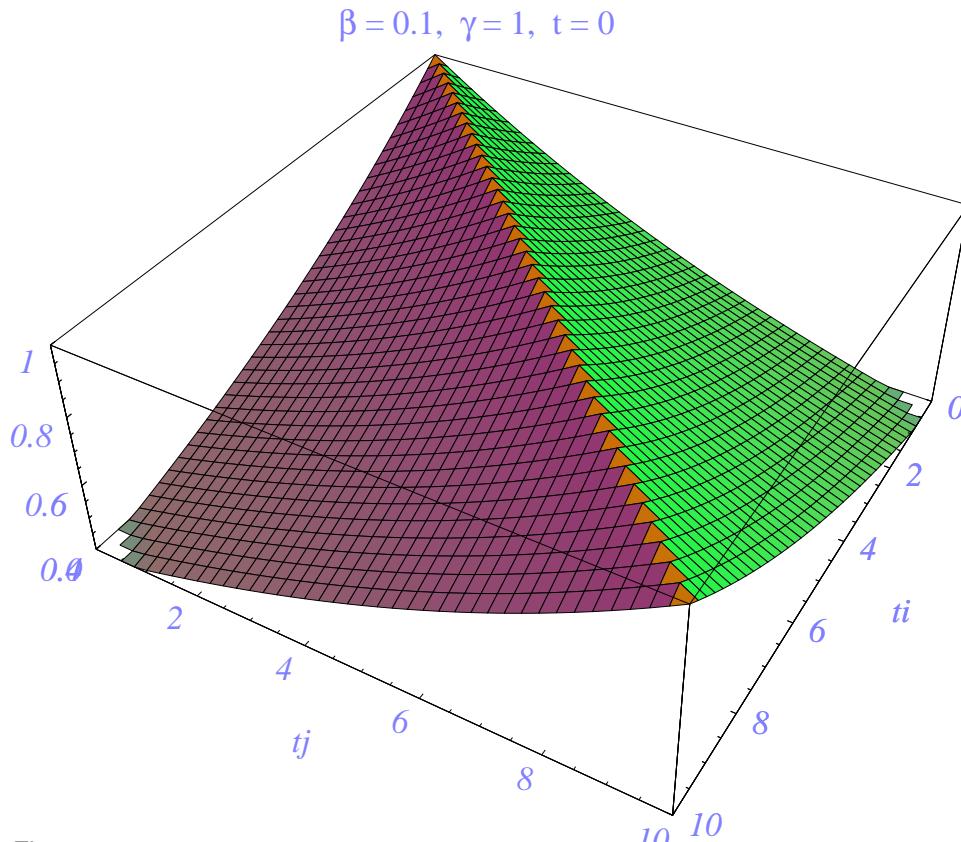
Figure VI.1.



An alternative to the time independent correlation function (58) is

$$\rho_{ij}(t) = (1 - \eta) \cdot e^{-\beta |(t_i - t)^\gamma - (t_j - t)^\gamma|} + \eta \quad (60)$$

with $\eta \in [-1, 1]$. Clearly, for $\gamma = 1$ and $\eta = 0$ this functional form is identical to (58). For the functional form (60), suitable parameters are $\gamma \approx 0.5$, $\beta \approx 0.35$, and $\eta \approx 0$.



VII. Factor reduction — pros and cons

It is possible to drive the evolution of the n forward rates with fewer underlying independent standard Wiener processes than there are forward rates, say only m of them.

In this case, the coefficient matrix $A \in \mathbb{R}^{n \times n}$ is to be replaced by $A \in \mathbb{R}^{n \times m}$ which must satisfy

$$\sum_{k=1}^m a_{jk}^2 = c_{jj} \quad (61)$$

in order to retain the calibration of the options on the FRAs, i.e. the caplets.

In practice, this can be done very easily by calculating the decomposition as in equation (4) as before and rescaling according to

$$a_{jk} \rightarrow a_{jk} \sqrt{\frac{c_{jj}}{\sum_{l=1}^m a_{jl}^2}}. \quad (62)$$

The effect of this procedure is that the individual variances of each of the rates are still correct, even if we have reduced the number of driving factors to one, but the effective covariances will differ.

However, if we allow for a term structure of volatility, or loading coefficients a_{jk} , factor reduction disables us from taking time steps longer than our time discretisation.

For example, assuming piecewise constant instantaneous loading coefficients for two factors of four forward rates over a first semi-annual time step

$$A_1 = \begin{pmatrix} 0.259 & 0.104 \\ 0.245 & 0.017 \\ 0.214 & -0.066 \\ 0.177 & -0.096 \end{pmatrix} \rightarrow C_1 = A_1 \cdot A_1^\top = \begin{pmatrix} 0.078 & 0.065 & 0.049 & 0.036 \\ 0.065 & 0.060 & 0.051 & 0.042 \\ 0.049 & 0.051 & 0.050 & 0.044 \\ 0.036 & 0.042 & 0.044 & 0.040 \end{pmatrix} \quad (63)$$

and over a second semi-annual time step

$$A_2 = \begin{pmatrix} 0.280 & 0.122 \\ 0.282 & 0.037 \\ 0.264 & -0.066 \\ 0.232 & -0.118 \end{pmatrix} \rightarrow C_2 = A_2 \cdot A_2^\top = \begin{pmatrix} 0.094 & 0.084 & 0.066 & 0.051 \\ 0.084 & 0.081 & 0.072 & 0.061 \\ 0.066 & 0.072 & 0.074 & 0.069 \\ 0.051 & 0.061 & 0.069 & 0.067 \end{pmatrix} \quad (64)$$

results in the sum of covariances over both time steps

$$C_{1+2} = C_1 + C_2 = \begin{pmatrix} 0.171 & 0.149 & 0.115 & 0.086 \\ 0.149 & 0.141 & 0.123 & 0.103 \\ 0.115 & 0.123 & 0.124 & 0.113 \\ 0.086 & 0.103 & 0.113 & 0.108 \end{pmatrix} = A_{1+2} \cdot A_{1+2}^\top \quad (65)$$

with

$$A_{1+2} = \begin{pmatrix} 0.381 & 0.161 & 0.005 & 0.000 \\ 0.374 & 0.039 & -0.006 & -0.001 \\ 0.340 & -0.094 & -0.004 & 0.001 \\ 0.291 & -0.152 & 0.006 & 0.000 \end{pmatrix}. \quad (66)$$

It is clear that already the combination of just two atomic time steps in one move requires the retainment of at least one additional factor to do the calibrated covariance structure justice.

**Factor reduction either prohibits term structure of instantaneous volatility
(which is very important in interest rate markets)
or the ability to take long time steps.**

Using fewer factors than discrete forward rates means

- a destruction of
 - ◊ either the term structure of instantaneous volatility of FRAs
 - ◊ or the correlation structure of the FRAs
 - ◊ or both
- that simultaneous calibration to market instruments of different nature such as caplets and swaptions becomes practically impossible
- the model loses its *market* feature and becomes a *factor* model
- no noticeable speed gain unless you have significantly fewer than $n/4$ factors in which case calibration flexibility is almost completely lost



VIII. Speed is everything — the drift term

If we wish to run a Libor market model with deterministic volatilities in the spot Libor measure with m factors, expediency commands that we precompute all of the terms

$$\hat{c}_{ijk} := \int_{t_i}^{t_{i+1}} \sigma_j^{\text{DD}}(t) \rho_{jk}(t) \sigma_k^{\text{DD}}(t) dt \quad (67)$$

and the spectral split matrices \hat{A}_i such that

$$\hat{c}_{ijk} = \sum_{l=1}^m \hat{a}_{ijk} \hat{a}_{ijl} \quad (68)$$

i.e.

$$\hat{C}_i = \hat{A}_i \cdot \hat{A}_i^\top. \quad (69)$$

Let us recall that the predictor corrector scheme expressed in the displaced forward rates

$$\mathbf{x} := \mathbf{f} + \mathbf{s} \quad (70)$$

is

$$x_j^{\text{Predictor}}(t_{i+1}) = x_j(t_i) \cdot e^{\hat{\mu}_{ij}(\mathbf{x}(t_i)) - \frac{1}{2}\hat{c}_{ijj} + \sum_{l=1}^m \hat{a}_{ijl} z_l} \quad (71)$$

$$x_j^{\text{Corrector}}(t_{i+1}) = x_j(t_i) \cdot e^{\hat{\mu}_{ij}(\mathbf{x}^{\text{Predictor}}(t_{i+1})) - \frac{1}{2}\hat{c}_{ijj} + \sum_{l=1}^m \hat{a}_{ijl} z_l} \quad (72)$$

$$x_j^{\text{Predictor-Corrector}}(t_{i+1}) = \sqrt{x_j^{\text{Predictor}}(t_{i+1}) \cdot x_j^{\text{Corrector}}(t_{i+1})} \quad (73)$$



The drift formula, as already given in equation (41), is

$$\hat{\mu}_{ij}(x) := \sum_{k=i}^j q_k \hat{c}_{ijk} \quad (74)$$

$$\begin{aligned} &= \sum_{k=i}^j q_k \sum_{l=1}^m \hat{a}_{ijl} \hat{a}_{ikl} \\ &= \sum_{l=1}^m \hat{a}_{ijl} \sum_{k=i}^j q_k \hat{a}_{ikl} \end{aligned} \quad (75)$$

wherein

$$q_k := \frac{x_k \tau_k}{1 + (x_k - s_k) \tau_k} . \quad (76)$$

In practice, the main bottleneck is the calculation of the drift term $\hat{\mu}_{ij}(x)$.



At time t_i , only $n = N - i$ forward rates are alive.

Thus, for $t_i \rightarrow t_{i+1}$, to compute all of the drift terms $\hat{\mu}_{ij}(\mathbf{x})$ for $j = i \dots N$, using the formula (74)

$$\hat{\mu}_{ij}(\mathbf{x}) = \sum_{k=i}^j q_k \hat{c}_{ijk}$$

involves

$$\frac{n(n+1)}{2} \tag{77}$$

multiplications and additions⁴.

⁴assuming we have precomputed the coefficients q_k from this vector \mathbf{x} which we definitely must do (factor 2 speedup!)



For small m , the calculation of $\hat{\mu}_{ij}(x)$ for $j = i \dots N$, using the formula (75) as in

$$\hat{\mu}_{ij}(x) = \sum_{l=1}^m \hat{a}_{ijl} r_{ijl} \quad (78)$$

with the update rule

$$r_{ijl} = r_{i(j-1)l} + q_j \hat{a}_{ijl} , \quad (79)$$

which involves

$$2nm \quad (80)$$

multiplications and additions, can be advantageous⁵ if $2nm < \frac{n(n+1)}{2}$, i.e. if

$$m \leq n/4 . \quad (81)$$

In practice, the alternative algorithm (78)-(79) is only helpful in conjunction with extreme factor reduction.

⁵This recursive decomposition can be generalised for swap market models and other yield curve representations [PvR05]

IX. Analytical calibration to coterminal swaptions

A forward swap rate S_i

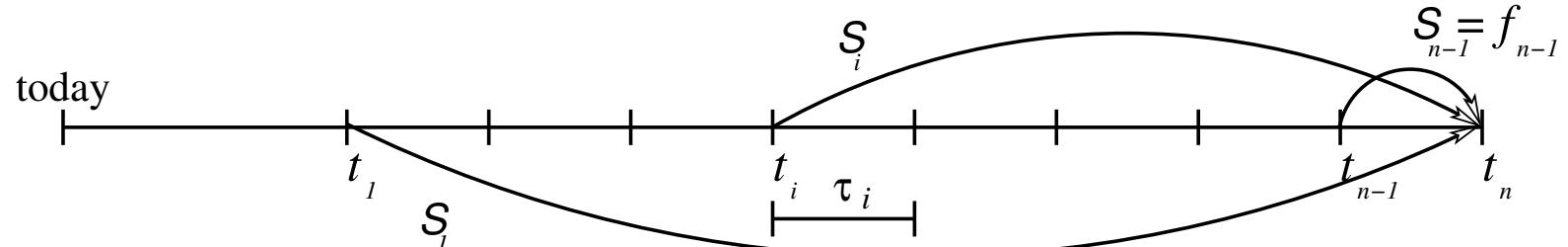


Figure IX.1.

can be written as the ratio

$$S_i = \frac{A_i}{B_i} \quad (82)$$

of the floating leg value A_i and the annuity B_i :

$$A_i := \sum_{j=i}^{n-1} P_{j+1} f_j \tau_j N_j \quad \text{for } i = 0 \dots n-1 \quad (83)$$

$$B_i := \sum_{j=i}^{n-1} P_{j+1} \tau_j N_j \quad \text{for } i = 0 \dots n-1. \quad (84)$$



N_j is the notional associated with accrual period τ_j .

Since the market convention of price quotation for European swaptions uses the concept of implied Black volatilities for the forward swap rate, it seems appropriate to think of the swap rates' covariance matrix in relative terms just as much as for the forward rates themselves.

For a set of coterminal swaps all ending with a final payment at t_n , the elements of the swap rate covariance matrix C^S can therefore be written as

$$\begin{aligned} C_{ij}^S &= \left\langle \frac{dS_i}{S_i} \cdot \frac{dS_j}{S_j} \right\rangle \Big/ dt \\ &= \sum_{k=0, l=0}^{n-1, n-1} \frac{\frac{\partial S_i}{\partial f_k} \cdot \frac{\partial S_j}{\partial f_l}}{S_i \cdot S_j} \cdot f_k f_l \cdot \left\langle \frac{df_k}{f_k} \frac{df_l}{f_l} \right\rangle \Big/ dt \\ &= \sum_{k=0, l=0}^{n-1, n-1} \frac{\partial S_i}{\partial f_k} \frac{f_k}{S_i} \cdot C_{kl}^f \cdot \frac{f_l}{S_j} \frac{\partial S_j}{\partial f_l} . \end{aligned} \tag{85}$$

Defining the elements of the matrix $Z^{f \rightarrow S}$ by

$$Z_{ik}^{f \rightarrow S} = \frac{\partial S_i}{\partial f_k} \frac{f_k}{S_i}, \quad (86)$$

the mapping from the FRA covariance matrix C^{FRA} to the swap rate covariance matrix C^S can be seen as a matrix multiplication:

$$C^S = Z^{f \rightarrow S} \cdot C^f \cdot Z^{f \rightarrow S}^\top. \quad (87)$$

Equations (86) and (87) are the basis of fast constructive calibration algorithms.



When the floating and fixed payments of a swap occur simultaneously with the same frequency, it is possible to find a simple formula for the swap rate coefficients $Z_{ik}^{f \rightarrow S}$. Using

$$\frac{\partial P_{i+1}}{\partial f_k} = -P_{i+1} \frac{\tau_k}{1 + f_k \tau_k} \cdot \mathbf{1}_{\{k \leq i\}} \quad \text{for } i, k < n, \quad (88)$$

where $\mathbf{1}_{\{k \geq i\}}$ is one if $k \geq i$ and zero otherwise, and equations (83), (84), and (82), we have

$$\frac{\partial S_i}{\partial f_k} = \left\{ \frac{P_{k+1} \tau_k N_k}{B_i} - \frac{\tau_k}{1 + f_k \tau_k} \cdot \frac{A_k}{B_i} + \frac{\tau_k}{1 + f_k \tau_k} \cdot \frac{A_i B_k}{B_i^2} \right\} \cdot \mathbf{1}_{\{k \geq i\}}. \quad (89)$$



This enables us to calculate the elements of the forward rate to swap rate covariance transformation matrix $Z^{f \rightarrow S}$ to obtain the expression

$$Z_{ik}^{f \rightarrow S} = \left[\underbrace{\frac{P_{k+1} N_k f_k \tau_k}{A_i}}_{\text{constant weights approximation}} + \underbrace{\frac{(A_i B_k - A_k B_i) f_k \tau_k}{A_i B_i (1 + f_k \tau_k)}}_{\text{shape correction}} \right] \cdot \mathbf{1}_{\{k \geq i\}} \quad (90)$$

The second term inside the square brackets of equation (90) is a *shape correction*. Rewriting it as

$$\frac{(A_i B_k - A_k B_i) f_k \tau_k}{A_i B_i (1 + f_k \tau_k)} = \frac{f_k \tau_k}{A_i B_i (1 + f_k \tau_k)} \cdot \sum_{l=i}^{k-1} \sum_{m=k}^{n-1} P_{l+1} P_{m+1} N_l N_m \tau_l \tau_m (f_l - f_m) \quad (91)$$

highlights that it is a weighted average over inhomogeneities of the yield curve.



In fact, for a flat yield curve, all of the terms $(f_l - f_m)$ are identically zero and the mapping matrix $Z^{f \rightarrow s}$ is equivalent to the so-called constant-weights approximation.

In practice the yield curve is never entirely flat which makes it necessary to compute the swap rate coefficients via the full derivative calculation (86).

When floating and fixed schedules differ, we have to compute the partial dependencies of the swap rates' floating payments, floating payment discount factors, and fixed payment discount factors individually, but we still obtain a transformation of the form (87), i.e

$$C^f \rightarrow C^s = Z^{f \rightarrow s} \cdot C^f \cdot Z^{f \rightarrow s}^\top .$$

We now have a map between the instantaneous FRA/FRA covariance matrix and the instantaneous swap/swap covariance matrix.

Unfortunately, though, the map involves the state of the yield curve at any one given point in time via the matrix Z .

The price of a European swaption does not just depend on one single realised state or even path of instantaneous volatility. It is much more appropriate to think about some kind of *path integral average volatility*.

Using arguments of factor decomposition and equal probability of up and down moves (in logarithmic coordinates), it can be shown [JR00, Kaw02, Kaw03] that the specific structure of the map allows us to approximate the effective implied swaption volatilities by simply using today's state of the yield curve for the calculation of the mapping matrix Z :

$$\hat{\sigma}_{S_i}(t, T) = \sqrt{\sum_{k=i, l=i}^{n-1} Z_{ik}^{f \rightarrow s}(0) \cdot \frac{\int_t^T \sigma_k(t') \sigma_l(t') \rho_{kl}(t') dt'}{T - t} \cdot Z_{il}^{f \rightarrow s}(0)} \quad (92)$$

This approximate equivalent implied volatility can now be used in a Black swap formula to produce a price *without the need for a single simulation*. In practice, the formula (92) works remarkably well:

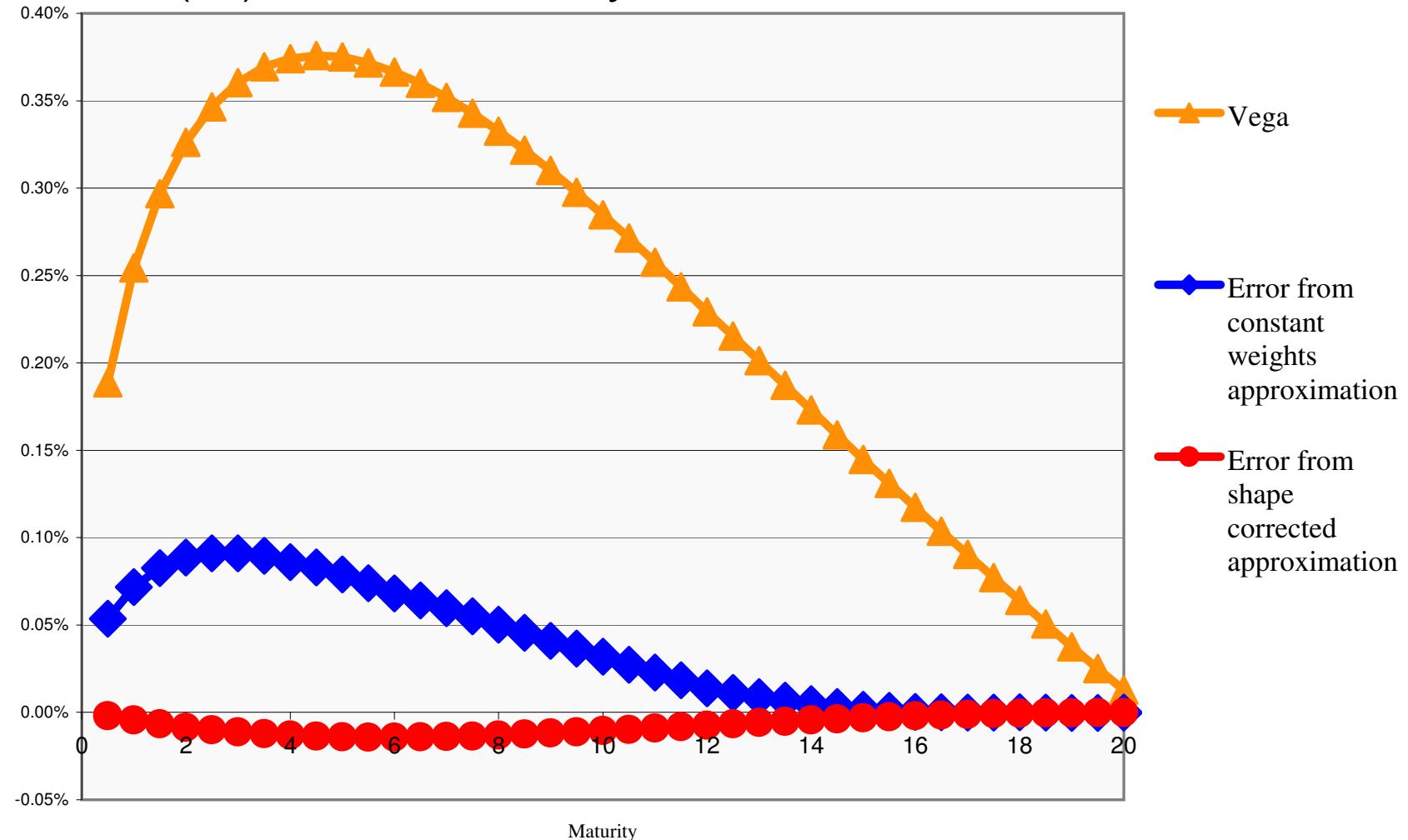


Figure IX.2.



We can now design a non-iterative calibration procedure that connects the step-wise covariance matrices of the logarithms of the realisations of the forward rates directly to the calibration volatilities of a set of European swaptions (including caplets):

- For any given time step from t to T , populate the time-unscaled FRA/FRA covariance matrix

$$C_{kl}^f = \frac{\int_{t'=t}^T \sigma_k(t') \sigma_l(t') \rho_{kl}(t') dt'}{T - t} . \quad (93)$$

- Next, map this matrix into a time-unscaled swap/swap covariance matrix using the Z matrix calculated from the initial state of the yield curve

$$C^S = Z \cdot C^f \cdot Z^\top . \quad (94)$$



Note:

- ◊ This swap rate/swap rate covariance matrix is associated with forward swap rates that expire at times equal to or later than T .
 - ◊ Its diagonal elements are the mean square volatilities of the n swap rates over the time step $t \rightarrow T$.
 - ◊ For $t = 0$ and $T = t_1$, the diagonal element C_{11}^s represents the square of the FRA-covariance-matrix-implied Black volatility of the first swaption, which, if the model was already calibrated, should be equal to the market implied volatility of the swaption expiring at time t_1 denoted by $\sigma_{S_1}^{\text{market}}$.
- Since variances are additive, we have

$$C_{ij}^S(0, t_k) \cdot t_k = C_{ij}^S(0, t_{k-1}) \cdot t_{k-1} + C_{ij}^S(t_{k-1}, t_k) \cdot (t_k - t_{k-1}) \quad (95)$$

for $k \geq \max(i, j)$.



- In other words, we can compute the time-integrated (smaller) covariance matrix for a set of swaptions expiring at a later date by adding a subset of the (larger) time-integrated covariance matrix to an earlier date and the time-integrated covariance matrix from that earlier date to the later date.
- This additive feature of covariances means that we can accomplish calibration of each swaption individually by rescaling the whole swap rate covariance matrix such that the diagonal elements, when averaged to the expiry date of any individual swaption, match the square of the respective market given implied volatility.
- For this purpose, define the diagonal matrix Ξ by

$$\Xi_{gh} = \frac{\hat{\sigma}_{S_h}^{\text{market}}}{\hat{\sigma}_{S_h}(0, t_h)} \cdot \delta_{gh} \quad (96)$$

with δ_{gh} being the Kronecker symbol (which is zero unless $g = h$ when it is one) and $\hat{\sigma}_{S_h}(0, t_h)$ calculated from the FRA instantaneous volatility parametrisation through equation (92).



- The calibrated swap rate/swap rate covariance matrix for any time step $t \rightarrow T$ is thus given by

$$C_{\text{calibrated}}^S = \Xi \cdot Z \cdot C^f \cdot Z^\top \cdot \Xi . \quad (97)$$

- When Z is invertible, we can therefore define the calibration matrix

$$M := Z^{-1} \cdot \Xi \cdot Z \quad (98)$$

and express the entire calibration procedure as the simple operation

$$C_{\text{calibrated}}^f = M \cdot C_{\text{parametric}}^f \cdot M^\top . \quad (99)$$

- **Note:** The matrix M only depends on the initial yield curve and known volatility parameters, but *not* on the time step over which we want to construct the covariance matrix $C_{\text{calibrated}}^f$, and is therefore the same for all time steps!



In order to use the matrix $C_{\text{calibrated}}^f$ for the evolution of the yield curve over the time step $t \rightarrow T$ from a set of standard normal variates, we now simply need to compute a pseudo-square root $A_{\text{calibrated}}^f$ such that

$$C_{\text{calibrated}}^f = A_{\text{calibrated}}^f A_{\text{calibrated}}^{f \top} \quad (100)$$

just as we would have done without calibration to swaptions.

In practice, a user may wish to specify not exactly as many swaptions as there are forward rates to calibrate to. Instead, it may be desirable to specify fewer, or even more than n swaption volatilities. In this case, the swap rate coefficient matrix Z may be over- or underdetermined. Either way, it is still possible to find a matrix M that can be used in equation (99). To find it, let us first consider the singular value decomposition [PTVF92] of the transpose of Z :

$$Z^\top = U \cdot W \cdot V^\top \quad (101)$$

In the underdetermined case, the diagonal matrix W will have some zero entries on the diagonal. Let us define W' as the diagonal matrix whose diagonal elements are the inverse of the corresponding elements in W where they are nonzero, and zero otherwise. The matrix product $(W'W)$ then has unit elements wherever W has nonzero entries, and formally constitutes a *projection* matrix by virtue of the fact that its repeated application to any target vector has the

same result as a single multiplication, i.e.

$$(W'W)^k \cdot X = (W'W) \cdot X \quad \forall k \geq 1 \text{ and } \forall X. \quad (102)$$

The calibration procedure in the present framework amounts to the identification of $A_{\text{calibrated}}^f$ that satisfies

$$Z \cdot A_{\text{calibrated}}^f = \Xi \cdot Z \cdot A_{\text{parametric}}^f \quad (103)$$

but remains as close to the original $A_{\text{parametric}}^f$ as possible, i.e. to find the matrix $A_{\text{calibrated}}^f$ that meets equation (103) and simultaneously minimises

$$\left\| A_{\text{calibrated}}^f - A_{\text{parametric}}^f \right\| \quad (104)$$

for some suitable matrix norm. Denote the Moore-Penrose inverse [Alb72] of Z as $Z^{\widetilde{-1}}$, and write the product of $Z^{\widetilde{-1}}$ times Z itself as Q :

$$Q := U^{\top -1} W' V^{\top} \cdot V W U^{\top} \quad (105)$$

By the aid of the orthogonality conditions satisfied by the constituents U and V of the singular value decomposition of Z , and by the fact that $(W'W)$ is a projection⁶, both Q and the matrix

$$P := \mathbf{1} - Q \quad (106)$$

are also projection operators. In fact, P is the projection onto the kernel of Z . The desired matrix $A_{\text{calibrated}}^f$ can thus be found by adding the projection of $A_{\text{parametric}}^f$ onto the kernel of Z and the Moore-Penrose solution to equation (103), i.e.

$$A_{\text{calibrated}}^f = P \cdot A_{\text{parametric}}^f + Z^{\widetilde{-1}} \cdot \Xi \cdot Z \cdot A_{\text{parametric}}^f. \quad (107)$$

Since $A_{\text{parametric}}^f$ appears as the last multiplicand in both of the summands on right hand side, we can rewrite this as

$$\begin{aligned} A_{\text{calibrated}}^f &= (\mathbf{1} - UW'WU^\top + UW'V^\top \Xi Z) \cdot A_{\text{parametric}}^f \\ &= (\mathbf{1} - UW' (WU^\top - V^\top \Xi Z)) \cdot A_{\text{parametric}}^f \\ &= (\mathbf{1} - UW' (1 - V^\top \Xi V) WU^\top) \cdot A_{\text{parametric}}^f. \end{aligned} \quad (108)$$

⁶ A projection operator is any operator T for which $T^n = T$ for all $n \geq 1$.



This means, the sought calibration matrix M is given by

$$M = \mathbf{1} - U \cdot W' \cdot \left(\mathbf{1} - V^\top \cdot \Xi \cdot V \right) \cdot W \cdot U^\top. \quad (109)$$

The key to this calibration procedure in the underdetermined case is that a *minimal* solution to the raw calibration problem (103) is combined with as much of the original covariance information as possible that has no effect on the calibration problem. In more formal terms, we combine the minimal solution to the calibration problem with the projection of the desired covariance structure onto the calibration kernel.

When Z is overdetermined, the correction matrix M cannot achieve calibration to all the desired market prices. Instead, the calibration procedure based on the linear algebraic operations above will result in a least squares fit in some suitable norm by virtue of the use of the singular value decomposition of Z .

Within the limits of the approximation (92), the operation given in equation (99) will provide calibration to European swaption prices whilst retaining as much calibration to the caplets as is possible without violating the overall FRA/FRA correlation structure too much.

X. Non-parametric volatility specification

Assume that we have a set of n forward rate fixing times (starting at the end of the accrual period of the spot Libor rate), and that the n associated forward rates span the forward yield curve unambiguously.

Also, assume that we are satisfied with a discretisation of the term structure of instantaneous covariance at the same level of temporal resolution as the forward rates themselves.

This means we can choose all correlation coefficients between all forward rates over all time steps as well as all volatility coefficients for all forward rates over all time steps at liberty to match any given set of calibration instruments.

**Note:**

- The interest rates markets provide sufficient liquidity for the hedging of the level of volatility
→ Hedging volatility is possible, and thus a market model should be calibrated to implied volatilities of relevant available contracts in the market.
- The interest rates markets provide practically no direct hedge against the level of forward rate correlations.
→ Efficiently hedging correlation is practically impossible, and thus a market model should not try to calibrate correlation figures in a closed calibration routine that disconnects the trader from any control over whether the resulting correlation levels are realistic.

Numerical calibration of correlation coefficients is dangerous.



Instead of a numerical calibration of a constant correlation structure, a reasonable parametrisation of the term structure of instantaneous correlation may be preferred. This way, we can allow for the correlation of two adjacent forward rates to gradually decrease as calendar time moves into the future.

In the following, we allow for stochastic stub evolution. However, since calibration instruments are not affected by this extension, it is not reasonable to calibrate the stochastic stub volatility. In practice, it suffices to set it to the last volatility the corresponding just-fixed canonical rate experienced immediately prior to its fixing time.

This means, the number of free volatility coefficients $\bar{\sigma}_{ij}$ we are at liberty to choose for the discrete evolution set

$$t_i \rightarrow t_{i+1} \quad \text{for } i = 1..n$$

is given by

$$\sum_{i=1}^n \sum_{j=i}^n 1 = \frac{1}{2}n(n+1). \quad (110)$$



However, the number of correlation coefficients $\bar{\rho}_{ijk}$, which we choose in a reasonable fashion and *not* make the subject of any calibration procedure is

$$\sum_{i=1}^n \sum_{j=i}^n \sum_{k=j}^n 1 = \frac{1}{6}n(n+1)(n+2). \quad (111)$$

In practice, we will never have $\frac{1}{2}n(n+1)$ calibration instruments of different expiry and/or tenor.

It is thus prudent to choose a *skeleton* or *reference volatility structure* $\sigma^{\text{reference}}(t, T) = \sigma^{\text{reference}}(T - t)$ that is time-homogeneous and aim for calibration that is close to the reference structure.

The reference volatility structure serves a second purpose: it can be used to determine and freeze the reference correlation structure

$$\bar{\rho}_{ijk} := \frac{\int_{t_i}^{t_{i+1}} \sigma^{\text{reference}}(t, t_j) \rho_{jk}(t) \sigma^{\text{reference}}(t, t_k) dt}{\sqrt{\int_{t_i}^{t_{i+1}} (\sigma^{\text{reference}}(t, t_j))^2 dt \cdot \int_{t_i}^{t_{i+1}} (\sigma^{\text{reference}}(t, t_k))^2 dt}} \quad (112)$$

wherein the parametric instantaneous correlation function can be chosen to be something like (60).

In practice, when the functional forms for $\sigma^{\text{reference}}(t, T)$ and $\rho_{jk}(t)$ don't permit analytical evaluation of (112), a four point Simpson rule is generally sufficiently accurate to represent the chosen correlation structure well enough.

The above procedure also gives us initial guesses for the *volatility coefficients* (aka piecewise constant instantaneous volatility levels):

$$\bar{\sigma}_{ij} \leftarrow \bar{\sigma}_{ij}^{\text{reference}} := \sqrt{\int_{t_i}^{t_{i+1}} (\sigma^{\text{reference}}(t, t_j))^2 dt} / (t_{i+1} - t_i) \quad (113)$$

XI. Global calibration to the full swaption matrix

For all caplets and swaptions⁷, an efficient and accurate approximate pricing formula of the form (92) can be found that links the Libor market model's covariance structure directly to the corresponding implied volatility of the respective calibration instrument.

Thus, calibration to a specific swaption # l expiring at time t_{S_l} is given if the ***primary calibration objective to reprice the market***

$$(\hat{\sigma}_{S_l}^{\text{atm}})^2 = \sum_{i=1}^{i[t_{S_l}]} \sum_{j,k} Z_{lj} \cdot \bar{\sigma}_{ij} \cdot \bar{\rho}_{ijk} \cdot \bar{\sigma}_{ik} \cdot Z_{lk} \cdot (t_{i+1} - t_i) \Bigg/ t_{S_l} \quad (114)$$

is satisfied with

$$Z_{lj} = \left. \frac{\partial S_l}{\partial f_j} \frac{f_j}{S_l} \right|_{t=0} . \quad (115)$$

⁷We shall treat caplets as a special case of swaptions from here on.

Perfect calibration will be given when this equation is exactly satisfied.

Whenever perfect calibration is not possible, an optimal fit can be defined as the set of volatility coefficients $\bar{\sigma}_{ij}$ that makes the difference between the left hand side and the right hand side minimal.

The matching of a given reference structure defined by individual coefficients $\bar{\sigma}_{ij}^{\text{reference}}$ can be specified as the linear system

$$\bar{\sigma}_{ij} = \bar{\sigma}_{ij}^{\text{reference}}. \quad (116)$$

Equally, time-homogeneity conditions can be specified as

$$\bar{\sigma}_{ij} = \bar{\sigma}_{i+1,j+1}. \quad (117)$$

Clearly, in general one can define that each of equations (114), (116), and (117) is to be matched as well as possible by demanding that a norm of the difference between the left and the right hand side is minimal. Typically, one would use a weighted sum of the squared differences as a suitable norm in this context.

In a more generic case of the calibration problem, we may have the market prices, or rather their associated implied variances, v_i for $i = 1..n_c$ instruments, and a pricing function $f_i(\varsigma)$ of the volatility coefficient vector ς as defined above, i.e. the right hand side of equation (114).

The requirement to meet all of the desirable features of the calibrated volatility vector ς (such as time-homogeneity and proximity to a set of reference values) as well as possible can be expressed as the minimisation of the penalty function

$$\frac{1}{2} \cdot \varsigma^\top \cdot M_0 \cdot \varsigma + \frac{1}{2} \cdot (\varsigma - \varsigma_{\text{reference}})^\top \cdot (\varsigma - \varsigma_{\text{reference}}) \quad \forall \quad \varsigma \Big|_{f(\varsigma) = \mathbf{v}} \quad (118)$$

for some suitably chosen symmetric matrix M_0 . In other words, the minimisation is to be carried out such that the calibration conditions to the right of the vertical bar are met. Naturally, perfect minimisation of the expression (118) is achieved when the L_2 norm

$$\| (M_0 + \mathbf{1}) \cdot \varsigma - \varsigma_{\text{reference}} \| \quad (119)$$

vanishes.



As an example, let us assume that we have four equal time steps and thus initially four forward rates. Define the volatility coefficients vector $\varsigma \in \mathbb{R}^{n_\varsigma}$ as

$$\varsigma := (\bar{\sigma}_{11}, \bar{\sigma}_{21}, \bar{\sigma}_{31}, \bar{\sigma}_{41}, \bar{\sigma}_{22}, \bar{\sigma}_{32}, \bar{\sigma}_{42}, \bar{\sigma}_{33}, \bar{\sigma}_{43}, \bar{\sigma}_{44})^\top.$$

If we only wish to calibrate such that we remain as closely as possible to the reference values $\varsigma^{\text{reference}}$, the matrix M_0 consists of zero elements, and $M := M_0 + \mathbf{1}$ is equal to the identity matrix $\mathbf{1}$.

However, if we wish to calibrate as time-homogeneously as possible, i.e. attempt to incorporate the conditions $\bar{\sigma}_{i,j} = \bar{\sigma}_{i+1,j+1}$, the matrix M becomes

$$M = (M_0 + \mathbf{1}) = \begin{pmatrix} 2 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 3 & 0 & 0 & -1 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 3 & 0 & 0 & -1 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 3 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 2 \end{pmatrix}. \quad (120)$$



This can be generalised to incorporate further desirable features of the calibrated volatility structure with respectively associated weights of importance in a list of ***secondary calibration objectives***:-

- Proximity to reference structure of volatility.

Minimise $(\bar{\sigma}_{ij} - \bar{\sigma}_{ij}^{\text{reference}})^2$ for all i, j , i.e.

$$(\boldsymbol{\varsigma} - \boldsymbol{\varsigma}_{\text{reference}})^\top \cdot (\boldsymbol{\varsigma} - \boldsymbol{\varsigma}_{\text{reference}})$$

Weighting coefficient w_r .

- Time-homogeneity.

Minimise $(\bar{\sigma}_{ij} - \bar{\sigma}_{i+1, j+1})^2$ for all i, j , i.e.

$$\boldsymbol{\varsigma}^\top \cdot M_h \cdot \boldsymbol{\varsigma} .$$

Weighting coefficient w_h .



- Volatility structure smoothness in calendar time.

Minimise $(\bar{\sigma}_{ij} - \bar{\sigma}_{i+1,j})^2$ for all i, j , i.e.

$$\boldsymbol{\varsigma}^\top \cdot M_{\textcolor{blue}{s}} \cdot \boldsymbol{\varsigma} .$$

Weighting coefficient $w_{\textcolor{blue}{s}}$.

- Volatility homogeneity of neighbouring forward rates.

Minimise $(\bar{\sigma}_{ij} - \bar{\sigma}_{i,j+1})^2$ for all i, j , i.e.

$$\boldsymbol{\varsigma}^\top \cdot M_{\textcolor{blue}{n}} \cdot \boldsymbol{\varsigma} .$$

Weighting coefficient $w_{\textcolor{blue}{n}}$.

The calibration condition $f(\boldsymbol{\varsigma}) = \boldsymbol{v}$ is nonlinear and thus difficult to preserve in any minimisation of (118).

However, since our primary calibration criterion is the matching of the market given variances, we ought to design a procedure whose main focus is on convergence to those, and only if there are any residual degrees of freedom in that procedure should we exploit them for the sake of the given reference, homogeneity, and smoothness preferences such as the minimisation of expression (118).

We therefore start the development of our global calibration algorithm from the local linearisation of the calibration conditions as the governing equation of an iterative procedure that leads from the current estimate $\boldsymbol{\varsigma}_k$ to the next estimate $\boldsymbol{\varsigma}_{k+1}$:

$$\mathbf{f}(\boldsymbol{\varsigma}_k) + J(\boldsymbol{\varsigma}_k) \cdot (\boldsymbol{\varsigma}_{k+1} - \boldsymbol{\varsigma}_k) - \boldsymbol{v} = 0 . \quad (121)$$

Hereby, $J(\boldsymbol{\varsigma}) \in \mathbb{R}^{n_c \times n_c}$ is the Jacobi matrix of the primary calibration criterion

$$J(\boldsymbol{\varsigma}) = \frac{\partial (\mathbf{f}(\boldsymbol{\varsigma}))}{\partial (\boldsymbol{\varsigma})} . \quad (122)$$

Due to the specific nature of our calibration problem, the function $f(\varsigma)$ happens to be symmetrically bilinear in the elements of ς . As a consequence, we have

$$f(\varsigma) = \frac{1}{2} J(\varsigma) \cdot \varsigma , \quad (123)$$

and we can make use of this fact for optimisation.

From a geometric point of view, equation (121) means that the projection of the update vector

$$\Delta\varsigma_k := \varsigma_{k+1} - \varsigma_k \quad (124)$$

onto each of the row vectors of $J(\varsigma_k)$ must match the associated entry in the difference vector

$$\varepsilon_k := \mathbf{v} - f(\varsigma_k) , \quad (125)$$

i.e. the product of the increment vector $\Delta\varsigma_k$ with the Jacobi matrix J_k must equal the difference vector ε_k :

$$J_k \cdot \Delta\varsigma_k - \varepsilon_k = 0 \quad (126)$$



Since there are typically far fewer calibration instruments than free volatility coefficients, this leaves us a great deal of freedom in each Newton-Raphson step.

In order to use this freedom for the sake of the desirable features of the volatility structure, we complement the penalty function (118) expressed for the update vector $\varsigma_{k+1} = \Delta\varsigma_{k+1} + \varsigma_k$ by the hard constraint (126) weighted by a *Lagrange multiplier*⁸ to obtain the penalty function

$$\begin{aligned} & \frac{w_r}{2} \cdot (\Delta\varsigma_k + \varsigma_k - \varsigma_{\text{reference}})^\top \cdot (\Delta\varsigma_k + \varsigma_k - \varsigma_{\text{reference}}) \\ & + \frac{w_h}{2} \cdot (\Delta\varsigma_k + \varsigma_k)^\top \cdot M_h \cdot (\Delta\varsigma_k + \varsigma_k) \\ & + \frac{w_s}{2} \cdot (\Delta\varsigma_k + \varsigma_k)^\top \cdot M_s \cdot (\Delta\varsigma_k + \varsigma_k) \\ & + \frac{w_n}{2} \cdot (\Delta\varsigma_k + \varsigma_k)^\top \cdot M_n \cdot (\Delta\varsigma_k + \varsigma_k) \\ & + (J_k \cdot \Delta\varsigma_k - \varepsilon_k)^\top \cdot \lambda_k . \end{aligned} \tag{127}$$

⁸ The method shown has similarities with the one published in [Wu03] but was developed entirely independently.

Note: the above penalty function is specific to the k -th step of the iterative Newton-Raphson algorithm that we use to satisfy the primary calibration criterion.

Expression (127) is minimal with respect to all possible values of the vector $\Delta\varsigma_k$ when

$$M \cdot \Delta\varsigma_k + M \cdot \varsigma_k - w_r \cdot \varsigma_{\text{reference}} - J_k^\top \cdot \boldsymbol{\lambda}_k = 0, \quad (128)$$

where we have used

$$M := w_r \mathbf{1} + w_h M_h + w_s M_s + w_n M_n. \quad (129)$$

For $w_r \neq 0$, the symmetric matrix M is positive definite and thus equation (128) can formally be solved for $\Delta\varsigma_k$:

$$\Delta\varsigma_k = M^{-1} \cdot J_k^\top \cdot \boldsymbol{\lambda}_k - \varsigma_k + w_r \cdot M^{-1} \cdot \varsigma_{\text{reference}} \quad (130)$$



Since the last term on the right hand side remains constant throughout the whole Newton-Raphson process, we define

$$\mathbf{r} := w_r M^{-1} \boldsymbol{\varsigma}_{\text{reference}} \quad (131)$$

and precompute it. Also, let us define the matrix H_k^\top with $H_k \in \mathbb{R}^{n_c \times n_s}$ as the solution of the linear system

$$M \cdot H_k^\top = J_k^\top. \quad (132)$$

In this notation, we now have

$$\Delta \boldsymbol{\varsigma}_k = H_k^\top \cdot \boldsymbol{\lambda}_k - \boldsymbol{\varsigma}_k + \mathbf{r} \quad (133)$$

Substituting this into the constraint that the increment $\Delta \boldsymbol{\varsigma}_k$ must satisfy the Newton-Raphson condition (126), we obtain

$$J_k \cdot H_k^\top \cdot \boldsymbol{\lambda}_k - J_k \cdot \boldsymbol{\varsigma}_k + J_k \cdot \mathbf{r} - \boldsymbol{\varepsilon}_k = 0. \quad (134)$$

Using the definitions

$$G_k = J_k \cdot H_k^\top \quad (135)$$

$$\mathbf{y}_k = \boldsymbol{\varepsilon}_k + J_k \cdot (\boldsymbol{\varsigma}_k - \mathbf{r}) \quad (136)$$

we therefore obtain the vector of Lagrange multipliers $\boldsymbol{\lambda}_k$ as the solution of the linear system from the symmetric matrix $G_k \in \mathbb{R}^{n_c \times n_c}$

$$G_k \cdot \boldsymbol{\lambda}_k = \mathbf{y}_k . \quad (137)$$

Once we have computed $\boldsymbol{\lambda}_k \in \mathbb{R}^{n_c}$, the updated vector $\boldsymbol{\varsigma}_{k+1}$ is given by

$$\boldsymbol{\varsigma}_{k+1} = \boldsymbol{\varsigma}_k + \Delta \boldsymbol{\varsigma}_k = H_k^\top \cdot \boldsymbol{\lambda}_k + \mathbf{r} . \quad (138)$$

The key for fast calibration is the efficient solution of the involved large linear systems. A very useful tool for this purpose is the Iterative Template Library [LLS].



To summarise, here is the sequence of calculations that need to be carried out in the k -th iteration:

- Compute $J_k = J(\varsigma_k)$.
- Compute $f_k = \frac{1}{2}J_k \cdot \varsigma_k$.
- Solve $M \cdot H_k^\top = J_k^\top$ for H_k^\top . This is best done by solving for the rows of H_k individually using a conjugate gradient iterative linear system solver [LLS] taking advantage of the fact that M is symmetric and positive definite.
- Compute $y_k := v + f_k - J_k \cdot r$.
- Solve $G_k \cdot \lambda_k = y_k$ for λ_k . This should be done with a safe solver algorithm such as the Moore-Penrose pseudo-inverse [Alb72, PTVF92].
- Set $\varsigma_{k+1} = H_k^\top \cdot \lambda_k + r$.

Keep iterating until the maximum norm of f_k is sufficiently small (e.g. $< 0.1\%$).

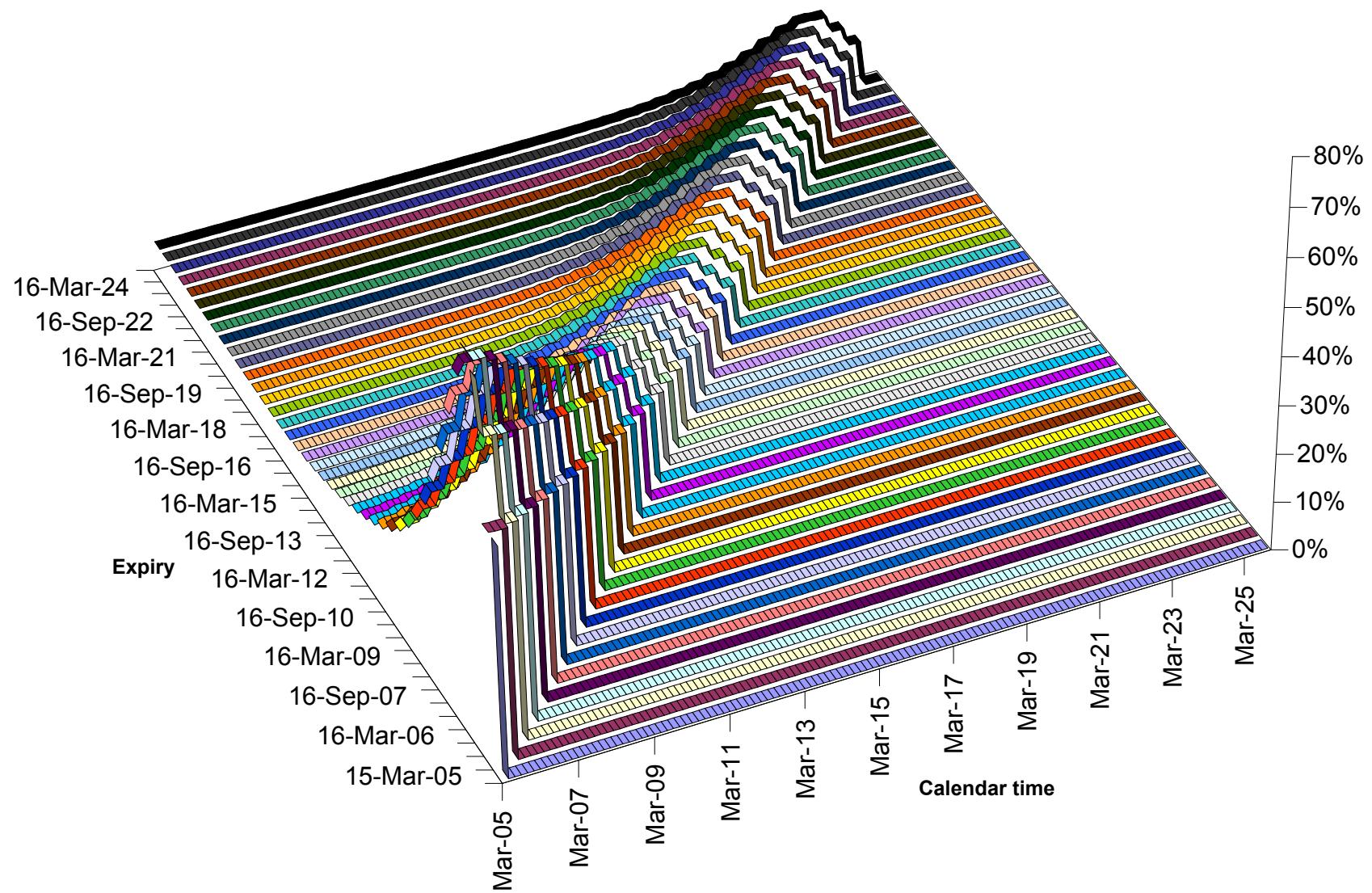


Figure XI.1. Instantaneous volatilities when only a reference structure is given (no calibration at all).

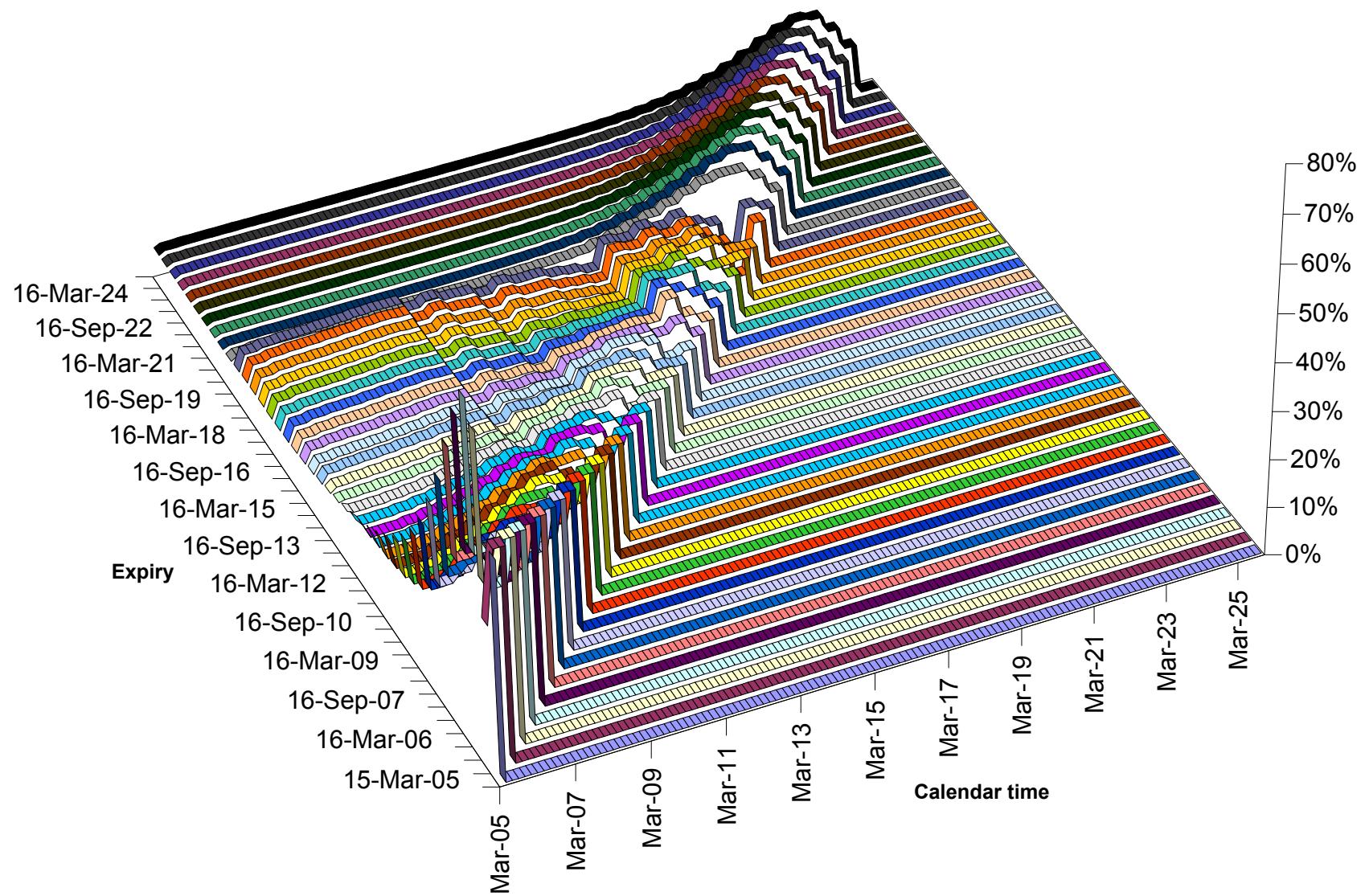


Figure XI.2. Instantaneous volatilities with reference structure and partial calibration ($w_r = 1, w_h = w_s = w_n = 1/10$).

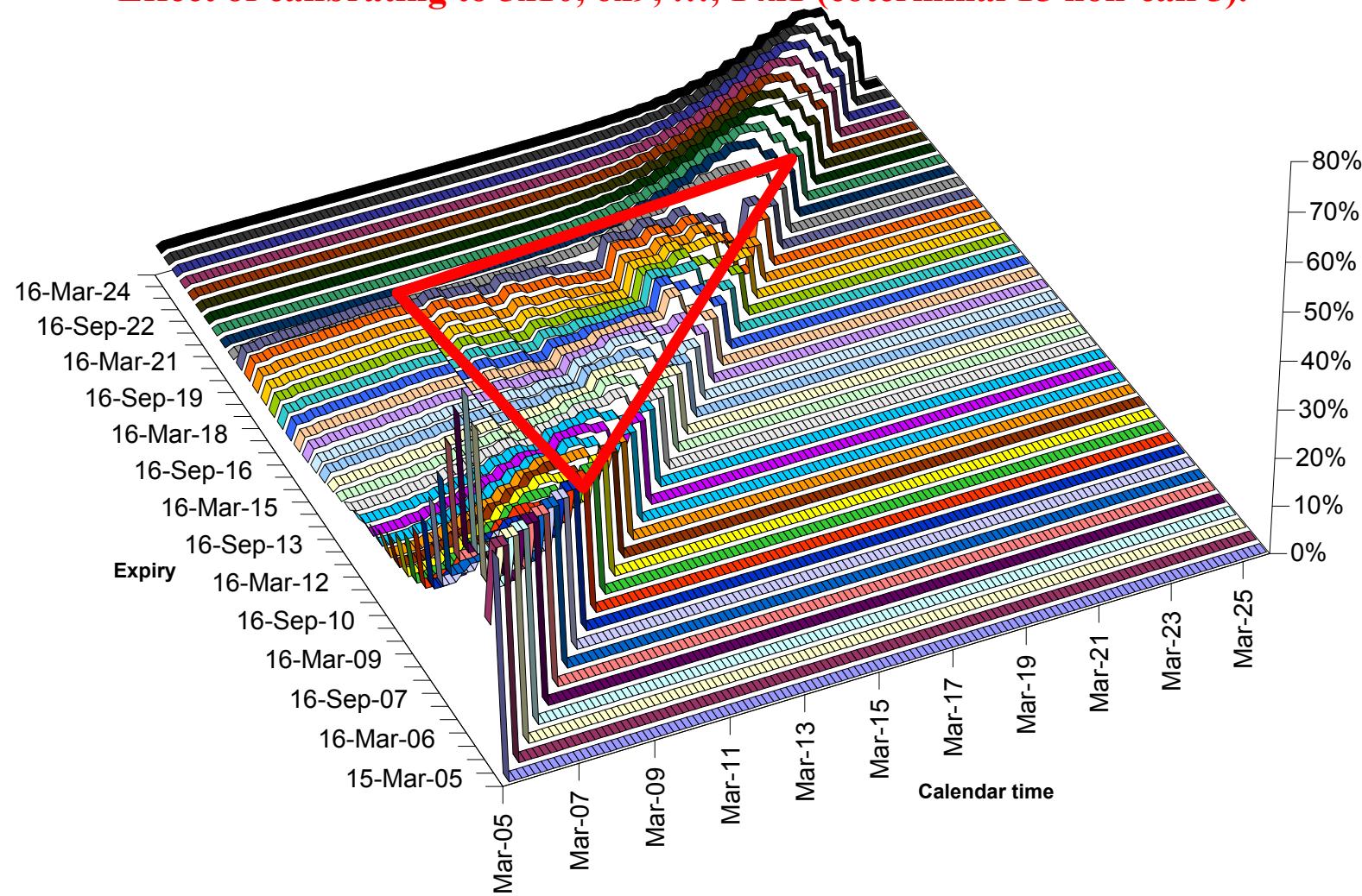
Effect of calibrating to 5x10, 6x9, ..., 14x1 (coterminal 15 non-call 5).

Figure XI.3. Instantaneous volatilities with reference structure and partial calibration ($w_r = 1, w_h = w_s = w_n = 1/10$).

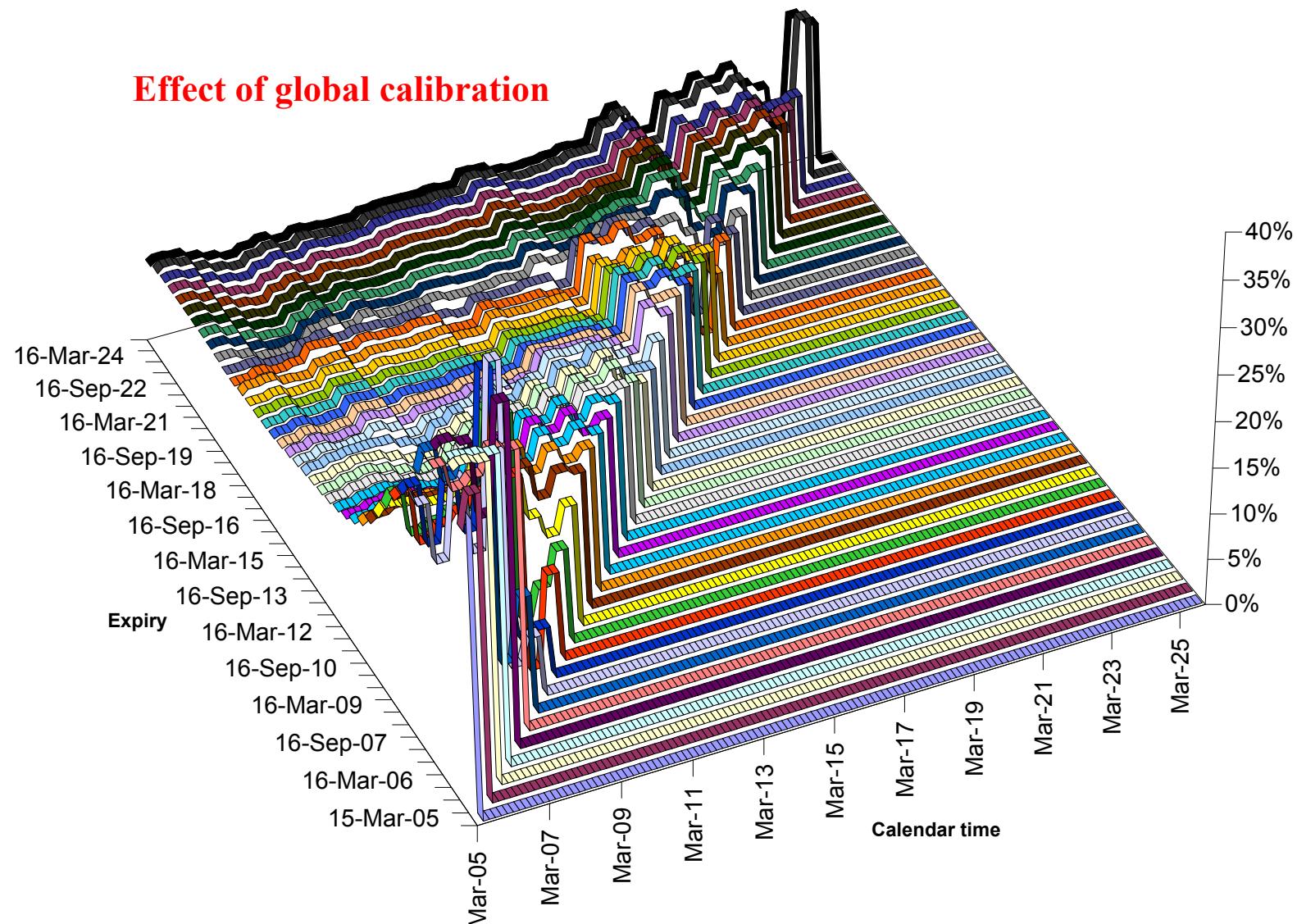


Figure XI.4. Instantaneous volatilities with reference structure and global calibration ($w_r = 1, w_h = w_s = w_n = 10$).



XII. Bermudan Monte Carlo

For interest rate derivatives, the most common types of early exercise deals nominally contain the *prerogative of early termination*.

When the concept of a prerogative of early termination is combined with an associated penalty payment that may be contingent on market observables and be payable in either direction of the contract, any early exercise specification can be accommodated.

Examples:

- Cancellable swap: swap + Bermudan swaption
- Option on option on option: three period cancellable irregular swap with bespoke cashflow schedule
- Bermudan best-of call $(\max(S_1, S_2) - K)_+$: cancellable European option paying $(\max(S_1, S_2) - K)_+$ at maturity with contingent penalty payment equal to $(\max(S_1, S_2) - K)_+$ payable to the prerogative holder.



Contingent claims that allow exercise at a given set of discrete points in time during the life of the contract are frequently referred to as *Bermudan*⁹

For a specific Bermudan contract, the expected profit or loss to the option holder depends on the strategy chosen by the holder of the prerogative.

A risk-neutral and perfectly rational investor will aim to find the strategy that maximises his profit.

For a set of discrete exercise opportunities t_j for $j = 1..m$, a rational investor can consider his exercise strategy as a set of m exercise decision indicator functions that all depend on the set of the prevailing financial state variables $x(t)$ in the respective associated filtration.

Define the exercise decision function $\mathcal{E}_j(x(t_j))$ such that the right to terminate a contract at time t_j is exercised if

$$\mathcal{E}_j(x(t_j)) > 0 .$$

⁹ There are various different approaches for the valuation of such derivatives [And00, LS98, BG97a, BD96, BG97b]. I present here my own tried-and-tested pet method.



This means that today's value of the contract for the given exercise strategy function vector \mathcal{E} is given by

$$V_0(\mathbf{x}(0)) = \mathbb{E}[\mathbf{1}_{\{\mathcal{E}_1(\mathbf{x}(t_1)) > 0\}} \cdot H_1(\mathbf{x}(t_1)) + \mathbf{1}_{\{\mathcal{E}_1(\mathbf{x}(t_1)) \leq 0\}} \cdot V_1(\mathbf{x}(t_1))] . \quad (139)$$

H_1 : numéraire-denominated net present value of all the cashflows occurring if the given right is exercised at the first exercise opportunity t_1 .

V_1 : the value of the derivative contract if the prerogative is not exercised at t_1 .

The exercise strategy dependent value of the contract is given by the recursive definition

$$V_{j-1}(\mathbf{x}(t_{j-1})) = \mathbb{E}[\mathbf{1}_{\{\mathcal{E}_j(\mathbf{x}(t_j)) > 0\}} \cdot H_j(\mathbf{x}(t_j)) + \mathbf{1}_{\{\mathcal{E}_j(\mathbf{x}(t_j)) \leq 0\}} \cdot V_j(\mathbf{x}(t_j))] \quad (140)$$

by setting $t_0 = 0$.

In order to find the fair value of the given contingent claim in a risk-neutral measure, let us view V_j as the *objective function* of an optimisation problem that is to find the exercise decision functions $\mathcal{E}_{j+1}, \mathcal{E}_{j+2}, \dots, \mathcal{E}_m$ that maximise V_j .

Fortunately, by virtue of the structure of equation (140) and by the aid of the tower law¹⁰, the recursive nature of the optimisation of the objective function decomposes into a sequence of non-recursive optimisations.

This can be seen by the fact that the optimisation of \mathcal{E}_j does not depend at all on knowledge as to whether one should ever exercise prior to t_j at all: the optimisation of \mathcal{E}_j only depends on the imminent and later exercise decision functions, not on earlier ones. This means that the last exercise decision function \mathcal{E}_m can be optimised entirely without the influence of any other strategic considerations other than the one at t_m . Once we have knowledge of the optimal function \mathcal{E}_m , however, the optimisation of \mathcal{E}_{m-1} , in turn, becomes a well defined optimisation problem in which only \mathcal{E}_{m-1} needs to be varied (since the truly optimal \mathcal{E}_m is already known) until V_{m-1} is maximised. Thus, the fair value of the derivative contract can be computed in a procedure in which a sequence of exercise strategy functions are optimised in their reverse order in time. The reverse order highlights the connection to conventional tree or finite differencing schemes: the fair value can only be computed with a method of *backwards induction* type.

It is, in general, not possible on a finite computer to implement an algorithm that can find the very optimal out of all possible exercise decision functions.

One can, however, devise methods that are able to come very close.

¹⁰also known as the law of iterated conditional expectations

A very simple, yet equally very robust, approach to achieve this is to rely on a suitable parametrisation of the sought exercise decision function and thus to change the nature of the optimisation problem from the search for the optimal function (which is really a problem of functional analysis and thus much harder to implement numerically) to the search for the optimal parameters. The implementation of the *prerogative-of-early-termination* algorithm can thus make use of the great numerical simplification that is given by the provision of a suitable parametrisation of the exercise decision function. Also, it is in general the case that not all state variables are needed for a good approximation to the optimal exercise decision function. Indeed, in most applications (in fact all I have seen so far), a projection of the state space variables onto a small set of financially relevant coordinates, say the vector \mathbf{f}_j associated with time t_j , suffices for a highly accurate optimisation.

In the following, the exercise decision function associated with exercise at time t_j is from here on denoted as

$$\mathcal{E}_j(\mathbf{f}_j; \boldsymbol{\lambda}_j)$$

with $\boldsymbol{\lambda}_j$ being a vector of parameters solely responsible for exercise at time t_j .

Note that we make no assumption that the parametric form of the exercise decision function, the number of parameters, the choice of financial coordinates, or the number of financial coordinates is the same for different exercise times. In practice though, due to most callable derivative structures having a large degree of time homogeneity in their contract definition, the functional and parametric form may be the same for all exercise dates.

The *prerogative-of-early-termination* algorithm leaves two very important tasks up to the designer of the functional and parametric form of the exercise decision function: a good approximation to the fair value can only be achieved

1. with a suitable choice of financial coordinates and
2. a reasonably flexible parametric form that allows for a good fit to the truly optimal exercise frontier.

Both of these choices depend typically very heavily on the contract structure, but, luckily, rarely on the actual *model* employed for the valuation. This means one can use the same parametrisation with a Hull-White model, or a Libor market model, or indeed any other reasonably well calibrated model!

Also, in all cases (the author has) seen so far it is possible to find a suitable exercise decision functional form that is robust with respect to the magnitude of the market parameters which is a crucial feature for the *prerogative-of-early-termination* algorithm to be a viable method in an integrated pricing system.

The algorithm

A suitable valuation algorithm for a prerogative of early termination can be separated into two main parts: the *training phase* and the *main evaluation phase*. The training phase itself consists of two stages: *the generation of the training set* and the *parameter optimisation stage*. And finally, the parameter optimisation as presented here also comprises two parts: the *Low-discrepancy Point search* (LP-search) [Sob79], and a *downhill-simplex optimisation* [PTVF92]:

- **Training**
 - ◊ **Generation of the training set**
 - ◊ **Parameter optimisation**
 - ➔ **Low-discrepancy Point search**
 - ➔ **Downhill-simplex optimisation**
- **Main evaluation**



The training phase

The starting point of the *prerogative-of-early-termination* methodology is the generation of a training set.

In a simplified way, one can think of the training set as a presimulated set of Monte Carlo evolution paths.

In practice, however, only the information relevant for the subsequent parameter optimisation is generated along any one path.

Generation of the training set

Let the

- number of paths in the training set be denoted as n ,

and the

- number of exercise time horizons as m .



Then, for each path indexed by the integer i , start the evaluation of the sequence of cashflow events (indexed by the integer j) at the beginning of the product description timeline, and aggregate the sum of cashflows divided by the value of the numéraire asset out to each exercise time horizon:

$$a_{ij} := \sum_{l=1}^j \frac{c_{il}}{N_{il}}. \quad (141)$$

Hereby, c_{il} stands for the value of the cashflow occurring in path $\#i$ at event $\#l$, and N_{il} stands for the value of the numéraire asset in the same path at the same point in time.

When an exercise of the prerogative occurs on path $\#i$ at a given time horizon $\#j$, the value of the path is the aggregated path value a_{ij} , amended by the penalty payment p_{ij} that is to be made there and then in case of early termination.



We denote the value of the path $\#i$ conditional on cancellation occurring at time horizon $\#j$ by

$$\pi_{ij} := a_{ij} + \frac{p_{ij}}{N_{ij}}. \quad (142)$$

Naturally, it is possible for no termination to occur at all for any one given path $\#i$.

Let the aggregate of all the cashflow payments divided by their associated numéraire asset values be denoted as the complete path continuation value q_i .

In addition to the cancellation path values π_{ij} and the continuation path values q_i , we also need to compute the financial coordinate values associated with each path at each exercise time horizon.

In the following, we denote the financial coordinate value in path $\#i$ at event $\#j$ for the coordinate $\#s$ as f_{ijs} .

Even though the choice of exactly two financial coordinates helps greatly in the visualisation of the exercise boundary, the *prerogative-of-early-termination* algorithm does not depend on there being exactly two financial coordinates.

The number of financial coordinates may vary from one exercise date to the next, and we therefore let r_j from now on represent their number at time t_j .

The generation of the training set thus entails the following:

- for each of the n paths in the training set (indexed by i)
 - ◊ for each of the m event time horizons (indexed by j)
 - ★ Compute the numéraire value N_{ij} ;
 - ★ Compute the cashflow value c_{ij} ;
 - ★ Compute the aggregated path value to this event $a_{ij} = a_{i,j-1} + \frac{c_{ij}}{N_{ij}}$;
 - ★ Compute the cancellation penalty payment value p_{ij} ;
 - ★ Store the cancellation path value $\pi_{ij} := a_{ij} + \frac{p_{ij}}{N_{ij}}$;
 - ★ Store the financial coordinate values $f_{ijs} \forall s = 1..r_j$;
 - ◊ Store the complete continuation value $q_i := a_{im}$.

The training set is therefore concisely described by a set of

$$n \cdot (m \cdot (r + 1) + 1)$$

numbers.

Since, for some models, the number of state space variables is much greater than the number of important financial coordinates for any one given product, storing all of the underlying financial variables for each path and each event horizon can easily pose a prohibitively large memory requirement.

By only storing the preprocessed relevant information associated with each path and each event horizon, the storage needs can be greatly reduced and thus is no issue for the implementation of the algorithm.



The objective function

Given a specified exercise decision function $\mathcal{E}_j(f_j; \lambda_j)$ associated with the prerogative at time t_j , the aim of the training phase is thus to find the parameter values $\lambda_j = \lambda_j^*$ such that the objective function

$$\hat{V}_j := \frac{1}{n} \sum_{i=1}^n \left[\mathbf{1}_{\{\mathcal{E}_j(f_{ij}; \lambda_j) > 0\}} \cdot \pi_{ij} + \mathbf{1}_{\{\mathcal{E}_j(f_{ij}; \lambda_j) \leq 0\}} \cdot q_i \right] \quad (143)$$

takes on its maximal value over all possible choices for λ_j .

This optimisation has to be carried out individually for each exercise time, and the optimisations over all j have to be done in reverse order, i.e.

$$j = m, (m-1), (m-2), \dots, 2, 1.$$

The key to the efficiency of the algorithm is that, following each optimisation of a specific parameter vector λ_j to find the optimal values λ_j^* , *the continuation variable values q_i for $i = 1, \dots, n$ are updated given the then known values λ_j^* .*

In other words, after the optimisation procedure has converged to the values λ_j^* , we override the continuation variable values

$$q_i := \mathbf{1}_{\{\mathcal{E}_j(f_{ij}; \lambda_j^*) > 0\}} \cdot \pi_{ij} + \mathbf{1}_{\{\mathcal{E}_j(f_{ij}; \lambda_j^*) \leq 0\}} \cdot q_i. \quad (144)$$

In practice, this of course means that we update each continuation variable value only if the just optimised exercise decision function \mathcal{E}_j indicates that for path # i exercise should occur at time t_j .

The benefit of this scheme is that the optimisation of m exercise decision functions' parameters is a task whose computation time grows only linearly in m since none of the later exercise decision functions are ever evaluated again once they have been optimised.



Strictly speaking, this procedure results in a value for \hat{V}_0 that is upwards biased due to the fact that the same set of paths is used for the optimisation of all exercise decision functions, though the bias diminishes quickly with increasing number of paths in the training set.

Fortunately, this poses no major problem since we can obtain an unbiased estimate for the fair value of the derivative contract by invoking a second Monte Carlo simulation that uses the optimised exercise decision functions but is based on a completely new set of simulated paths.

To avoid any upwards bias altogether, the *prerogative-of-early-termination* method consists of two parts, and it is the second part, the *main valuation phase*, that comprises the required simulation using a new set of paths generated to avoid an upwards bias in the final estimate for the derivative's value.

Parameter optimisation on the training set

As an example, let us assume that the financial variables of relevance of the cancellable derivative contract at hand span a two-dimensional coordinate system. Let us also assume that the training set consists of 2047 paths resulting in the same number of event coordinate pairs at each event date.

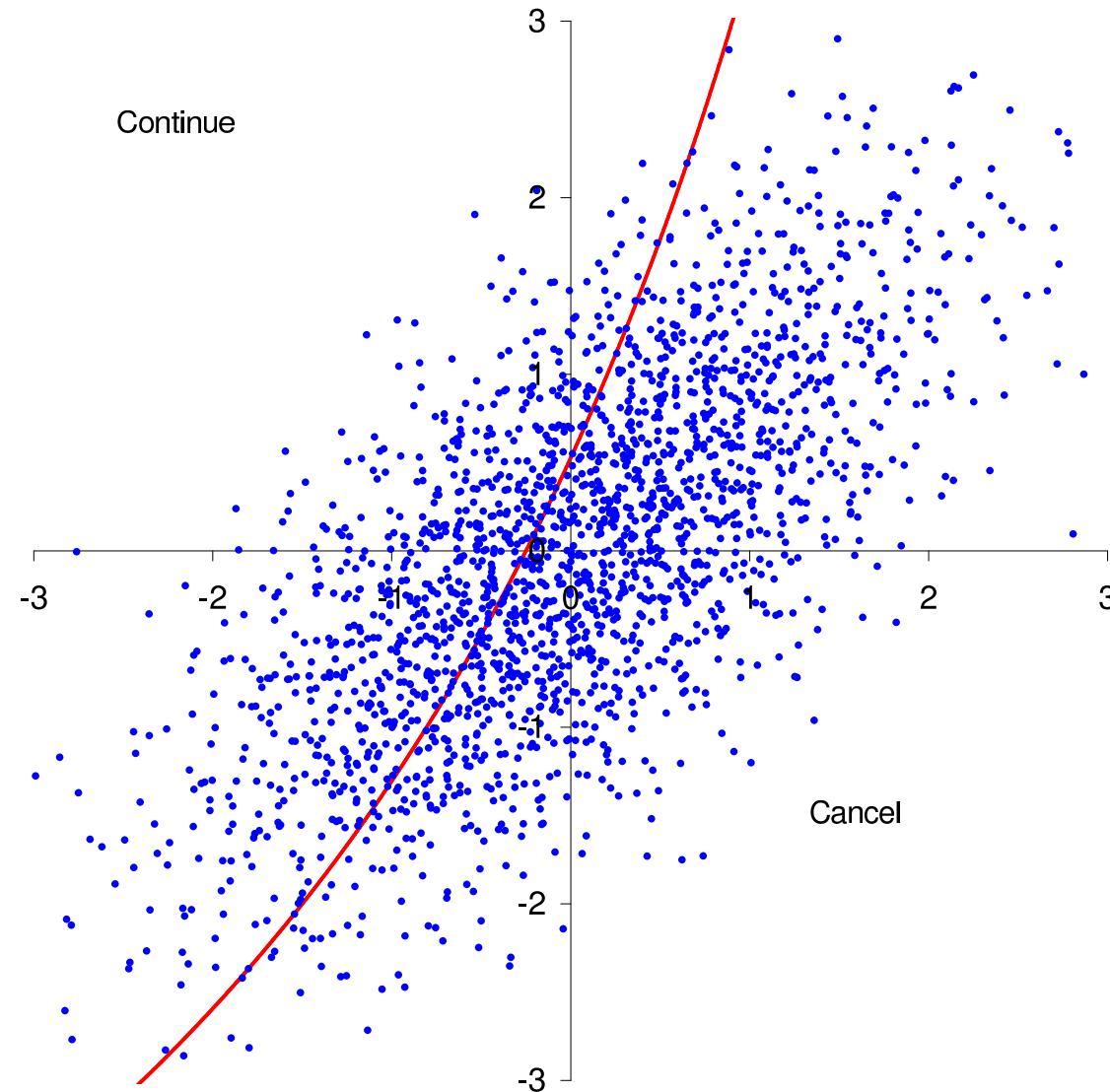


Figure XII.1. An example of the relevant financial variable values at a given exercise date for a training set of 2047 paths and an assumed truly optimal exercise boundary.



To simplify the illustration, we further assume that the continuation value of each path is +1 above the indicated truly optimal (but unknown to us) exercise boundary, and -1 below it.

The easiest way to specify an exercise decision function is to declare that cancellation should occur whenever a single indicating index breaches a certain threshold value known as *the trigger level*.

Here, we choose a one-dimensional trigger function such that cancellation occurs if the value of the first of the two selected financial coordinates is above the trigger level.

This means that we effectively use an exercise boundary that would appear as a vertical line in the diagram, and that our optimisation algorithm is simply to find the best location along the abscissa for the vertical trigger line.

Since the actual (truly optimal) exercise boundary is not perfectly vertical in our two-dimensional representation in figure XII.1, we expect the value of the objective function even for the optimal trigger level not to match the value given by the truly optimal exercise rule.

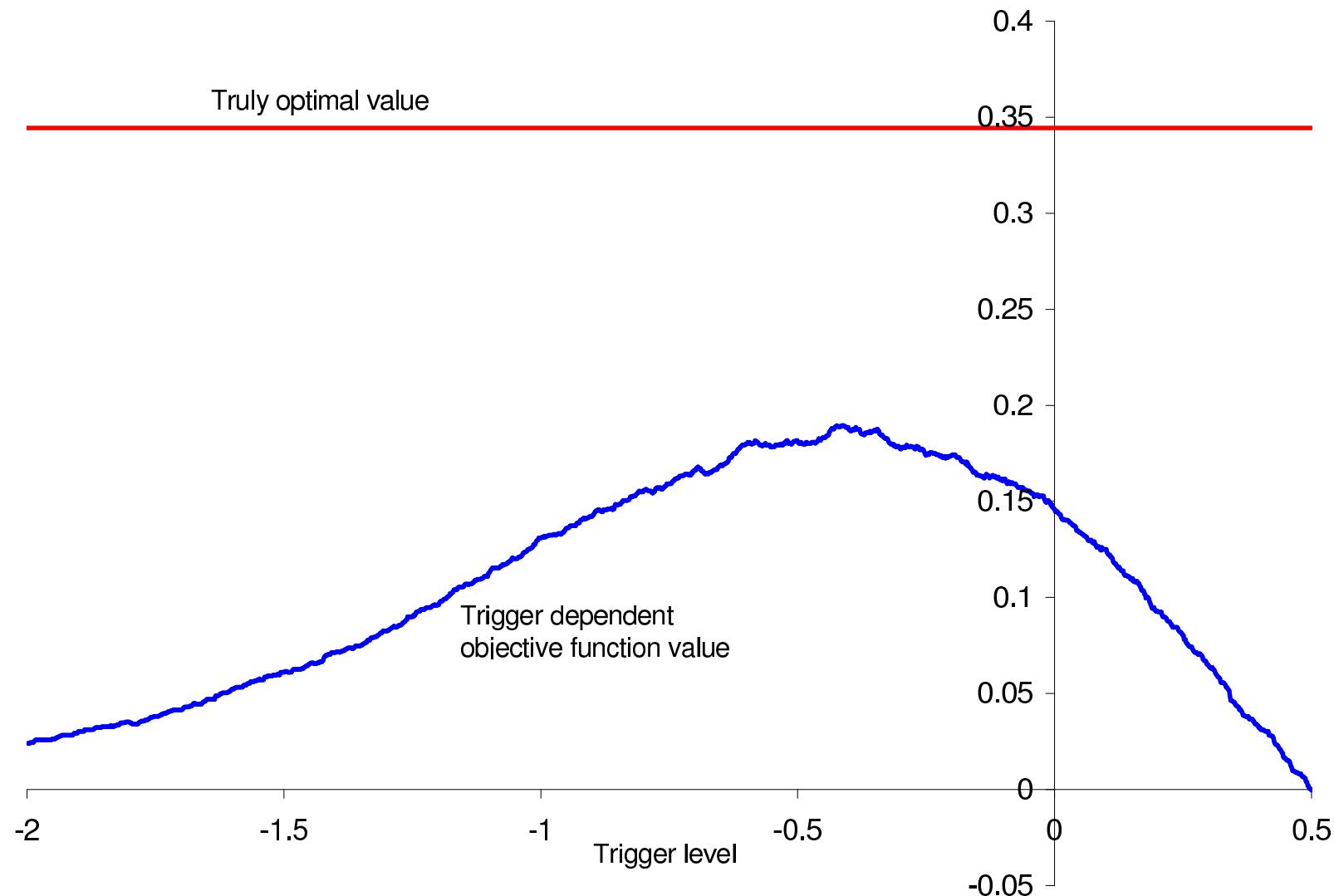


Figure XII.2. The dependence of the objective function defined using a one-dimensional trigger rule for the exercise decision function.

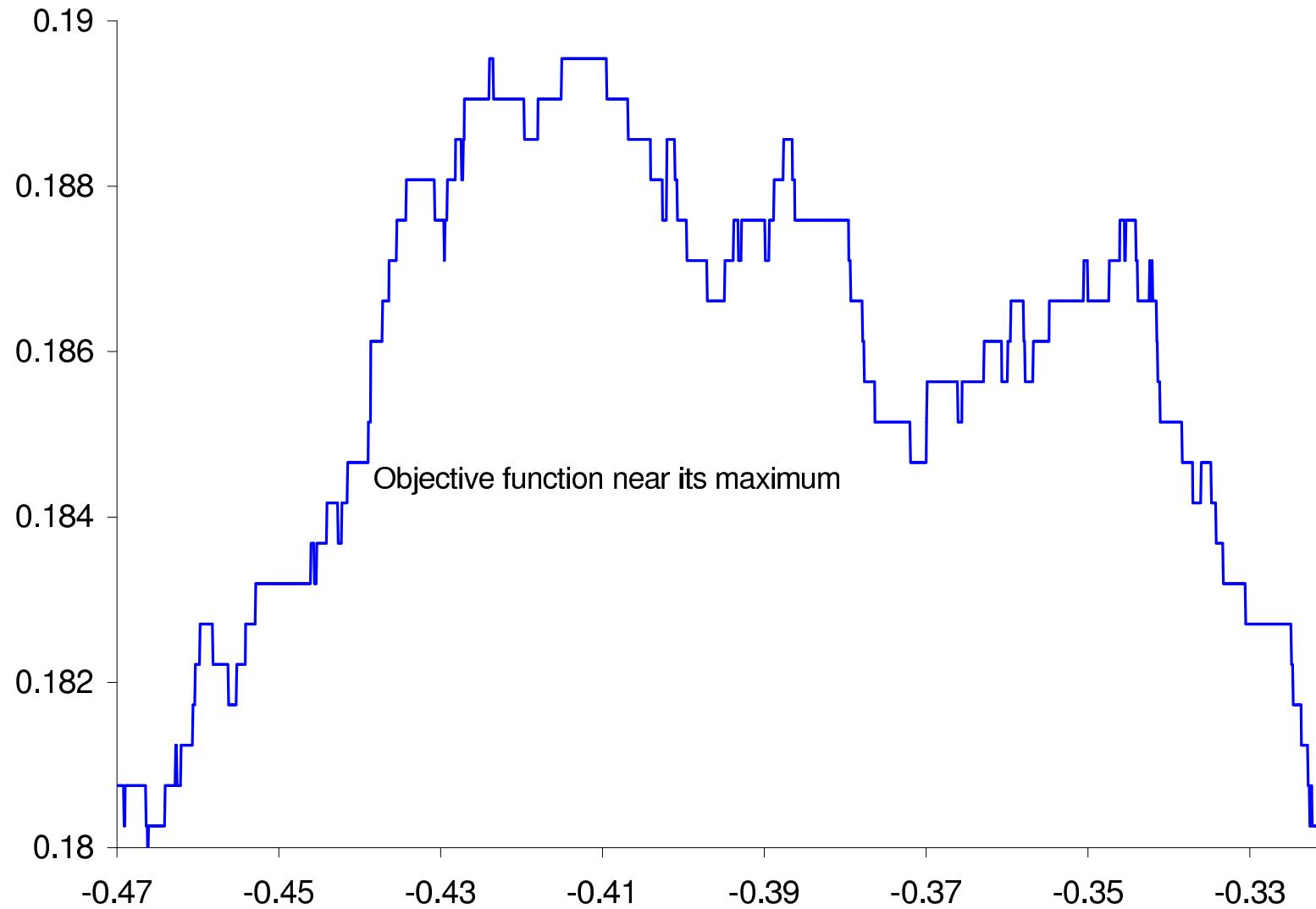


Figure XII.3. An enlargement of the objective function shown in figure XII.2.



There are two interesting features to note here:

1. The objective function appears to be *piecewise constant*.
2. The maximum of the function is not well defined.

Both observations can be readily explained:

Recall that the objective function is defined using a small set of discretely sampled training paths, and thus the distribution of the financial coordinate value pairs and the associated continuation values are subject to the usual error we incur in any Monte Carlo simulation.

This explains why there is more than one local maximum near what we would expect to be the location of the overall global maximum. In the limit of infinitely many paths in the training set, the function would converge to having a single local maximum. In other words, the fact that the maximum is not clearly defined is simply due to the inevitable presence of noise.

The piecewise constant nature of the objective function is also not difficult to understand:

It is caused by the fact that the objective function can only ever change value as and when the trigger level slides over the coordinate value of one of the event points in the training set.

Note: the locally noisy nature of the global maximum and the piecewise constant nature of the objective function are both genuinely intrinsic features of the *prerogative-of-early-termination* valuation methodology and not just artefacts of our choice of a one-dimensional exercise decision function.

The above mentioned features of the objective function have an important consequence with respect to the choice of the optimisation algorithm: *since the objective function is piecewise constant, its local gradient is always zero and thus methods based on an analytically computed gradient are not applicable!*



Of course, one can, instead of computing the gradient analytically, use a finite differencing algorithm for the approximation of the slope.

However, the use of a finite differencing algorithm requires knowledge of a suitable length scale to be used for the numerical differentiation.

Unfortunately, since the *prerogative-of-early-termination* method has to be robust with respect to changes of the product under consideration, market parameters, and other details, it would be very difficult indeed to design a finite differencing algorithm for the numerical approximation of the local gradient of sufficient reliability.

In addition to that, the fact that the global maximum is always subject to some local noise makes it practically impossible to use one of the many optimisation methods based on the local gradient that can be found in the literature.

To summarise:

- We cannot use a local gradient method easily.
- There may appear to be more than one local maximum.
- We cannot use a numerical approximation for the gradient based on any single estimate of the appropriate length scale.

All is not lost, though!

We can use a method that *changes its numerical length scale of spatial exploration in each and every iteration, first roaming the parameter space rather coarsely and then refining its search in the most promising area for the global maximum!*

A very simple algorithm with this feature is known as the *downhill-simplex algorithm* [PTVF92].

A simplex in d dimensions is essentially the volume spanned by $(d + 1)$ linearly independent corner points.

In two dimensions, a simplex is simply a triangle. In three dimensions, a simplex is a tetrahedron, and so forth.

To start the downhill-simplex algorithm, we need $(d + 1)$ initial guesses for the parameter vector λ .



Since it is in practice hard enough to demand the provision of *one* initial guess, we need a systematic method to generate those required $(d+1)$ simplex starting points.

One way to overcome this problem is to use a so-called *LP-search* [Sob79].

In this context, an LP-search is nothing other than, given an initial guess, the attempt to find other suitable points by sampling in the vicinity of the initial guess using a Sobol' vector sequence and some suitably selected joint distribution of all the parameter vector entries.

The implementation therefore requires the user to specify

- an initial guess for the exercise decision function parameter vector,
- a vector of standard deviations for each parameter,
- and the number of LP-search points that are to be used.

In practice, 63 LP-search points are usually enough when the exercise decision function has up to five parameters and a reasonable initial guess is provided.

The $(d + 1)$ simplex corner points can simply be chosen to be the $(d + 1)$ best guesses resulting from the initial LP-search.

The subsequent downhill-simplex algorithm is then invoked and continued until either the objective function converges on a relative scale (no further significant improvement of the function value possible) or until the simplex has been contracted by the algorithm to a comparatively small size (no further significant progress in parameter space).

The valuation phase

After the completion of the training phase, the *prerogative-of-early-termination* method continues with the main valuation.

This consists of an essentially independent Monte Carlo simulation using newly generated paths for the financial state variables. All exercise decisions are now taken on the basis of the fixed (optimised) parameter values.

Information loss due to inappropriate projection

The *prerogative-of-early-termination* method is very flexible to adapt to many different possible products and thus exercise domain types.

For instance, it is possible to design a function that is positive in several disjoint regions that are each individually well behaved singly connected exercise domains.

An application for such an exercise decision function parametrisation would be a Bermudan style option on the maximum of two assets against a fixed strike, i.e. an option with intrinsic value $(\max(S_1, S_2) - K)_+$.

However, since the flexibility is to some extent achieved by leaving the design of the exercise decision function parametrisation up to the financial engineer tayloring the product description, it is also possible to end up with noticeable undervaluation of the optionality in a given product due to an unfortunate mis-specification of the exercise decision function.



This is in itself not surprising.

A particularly dramatic example of this is that it is possible for an unfortunate mis-specification of an exercise decision function parametrisation with the *prerogative-of-early-termination* optimisation method to result in an option value that is *lower* than the value that can be obtained from a heuristically guessed set of parameters used *with the same exercise decision function*.

This may look at first as if the optimisation algorithm failed in its task and can be very confusing.

In order to explain how this phenomenon can arise, and to show that, when it happens, it really is simply due to the fact that a very important financial dimension has been ignored, we revisit the example in figure XII.1.

Recall that the maximum of the unconditional objective function as a function of the trigger level parameter in figure XII.3 was near the abscissa value -0.42.

This would mean that the approximate exercise boundary given the simple one-dimensional trigger function approach would result in the effective exercise strategy boundary given by a vertical line at abscissa value -0.42 in figure [XII.1](#).

Now, let us assume that instead of only one single exercise event, as we had so far in the example in figure [XII.1](#), there is now one further exercise event at a slightly *earlier* date.

We arbitrarily choose the earlier exercise event to cause early cancellation of all paths whose *second* coordinate is above -0.5 at that earlier date.

Since we assume some kind of two-dimensional diffusion process behind the path evolution, this means that out of all of our initially given 2047 sample paths, only a fraction ever arrives at the later event date we are interested in.

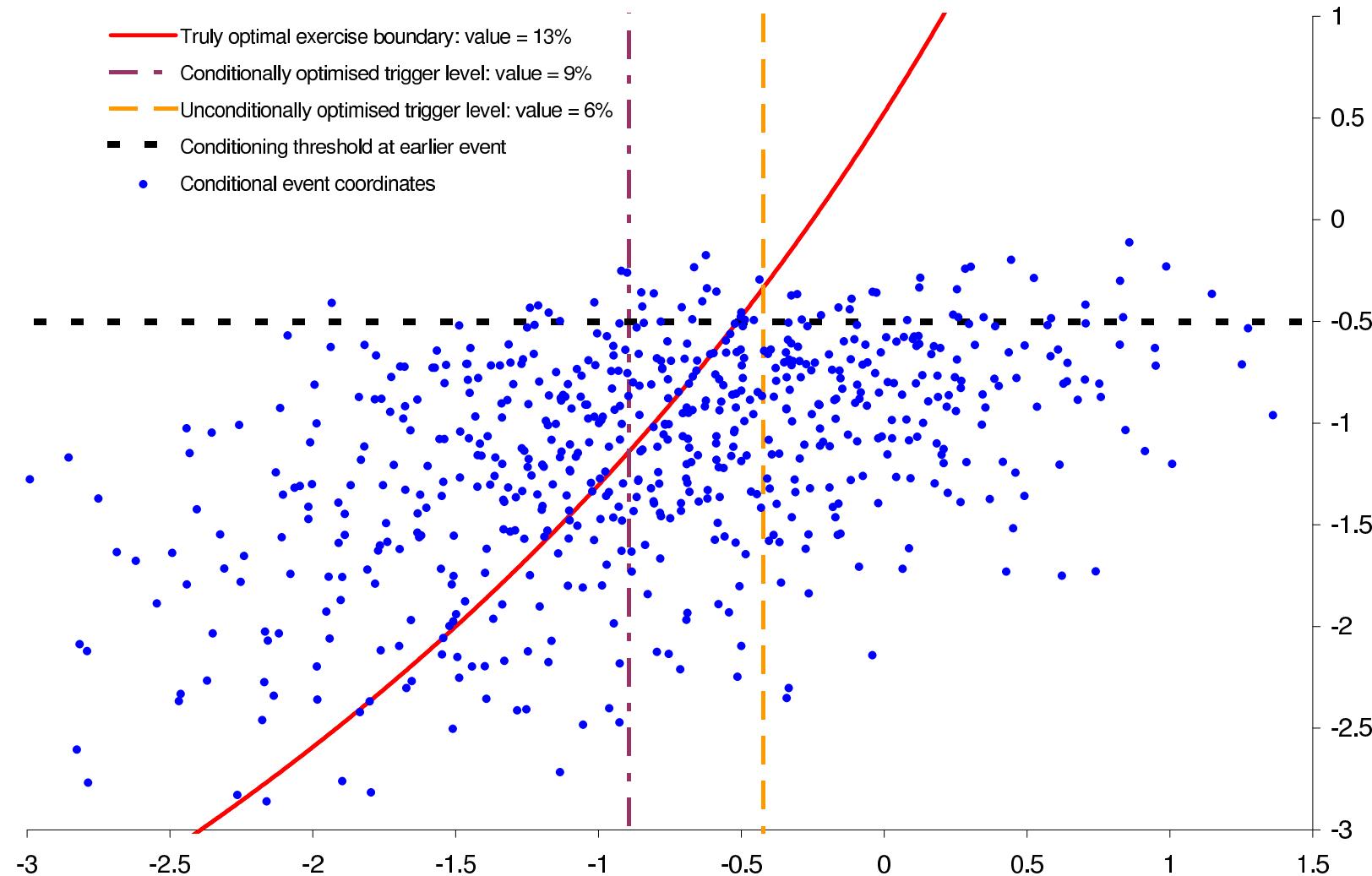


Figure XII.4. The value of the conditional objective function can be significantly lower when the parameter optimisation is carried out on an unconditional training set (backwards induction) when important financial variables were ignored (here: the second coordinate axis).

Only paths that prevail to the later (this) event date are marked as scattered dots.

Note that figure XII.4 is not exactly on the same scale as figure XII.1 which is why the scattered data set may appear somewhat distorted.

If you carefully compare the shape and location of several clusters you will recognise that it is indeed part of the same data set.

You may also note that some paths arrive at this event above the ordinate value -0.5, which is naturally caused by the diffusion process between the first and the second date.

The abscissa value of the right hand side vertical line is at -0.42 and corresponds to the one-dimensional trigger level that was obtained from the optimisation over the original, unconditional, training set.

The objective value associated with that trigger level for the unconditional training set, i.e. averaging over all paths in the training set without any filtering at an earlier event date, was approximately 19%.



However, if we restrict the averaging to those paths that *actually* arrive at the later event date due to the fact that many are already cancelled at an earlier date, we obtain a conditional objective function value of only around 6% when the one-dimensional trigger level is set at -0.42.

If, on the other hand, we now optimise the trigger level over the *conditional* training set, i.e. only over those paths that actually make it to the second event date, we find that a conditional trigger level of around -0.9 gives us a conditional objective function value of 9%.

In comparison, the *true* conditional objective function value, i.e. the value we obtain when using the actually optimal exercise boundary given by the solid line is 13%.

Whilst this example is somewhat contrived, it highlights the following problem: since the parameter optimisation of the *prerogative-of-early-termination* function is for efficiency reasons done in a backward induction fashion at each event date on the event set that takes no notice of earlier exercise opportunities, it is possible for the optimisation procedure to produce suboptimal parameter values *when the exercise decision function disregards important financial coordinates.*

It is important to emphasise, though, that this kind of problem is extremely rare and only occurs when either a single, comparatively simple, exercise decision function is used for a set of exercise events with strongly varying exercise boundary features due to the nature of the contract changing its characteristics in a major way throughout the life of the deal, or, when an extremely important financial coordinate has been disregarded completely in the design of the exercise decision function.

An example where the above described phenomenon has been observed in practice was a callable swap whose structured coupons are determined by an accreting ratchet formula, whereby the exercise decision function was indeed initially set to be a one-dimensional trigger rule.

Since the accreting factor is a strongly path-dependent variable for this kind of product, it proved to be essential that the exercise decision function took it into account.

Changing the exercise decision function to a two-dimensional rule with one of the financial coordinates being the accreting factor immediately remedied the issue.

An alternative solution to the phenomenon described in this section would be to design a variation of the *prerogative-of-early-termination* optimisation procedure that always optimises on the *conditional* training set.

Since this approach would result in a growth of the calculation time quite easily from being linear in the number of event dates, to being quadratic in the number of event dates, it is at present not envisaged to implement this alternative.



Applied example: Bermudan swaption.

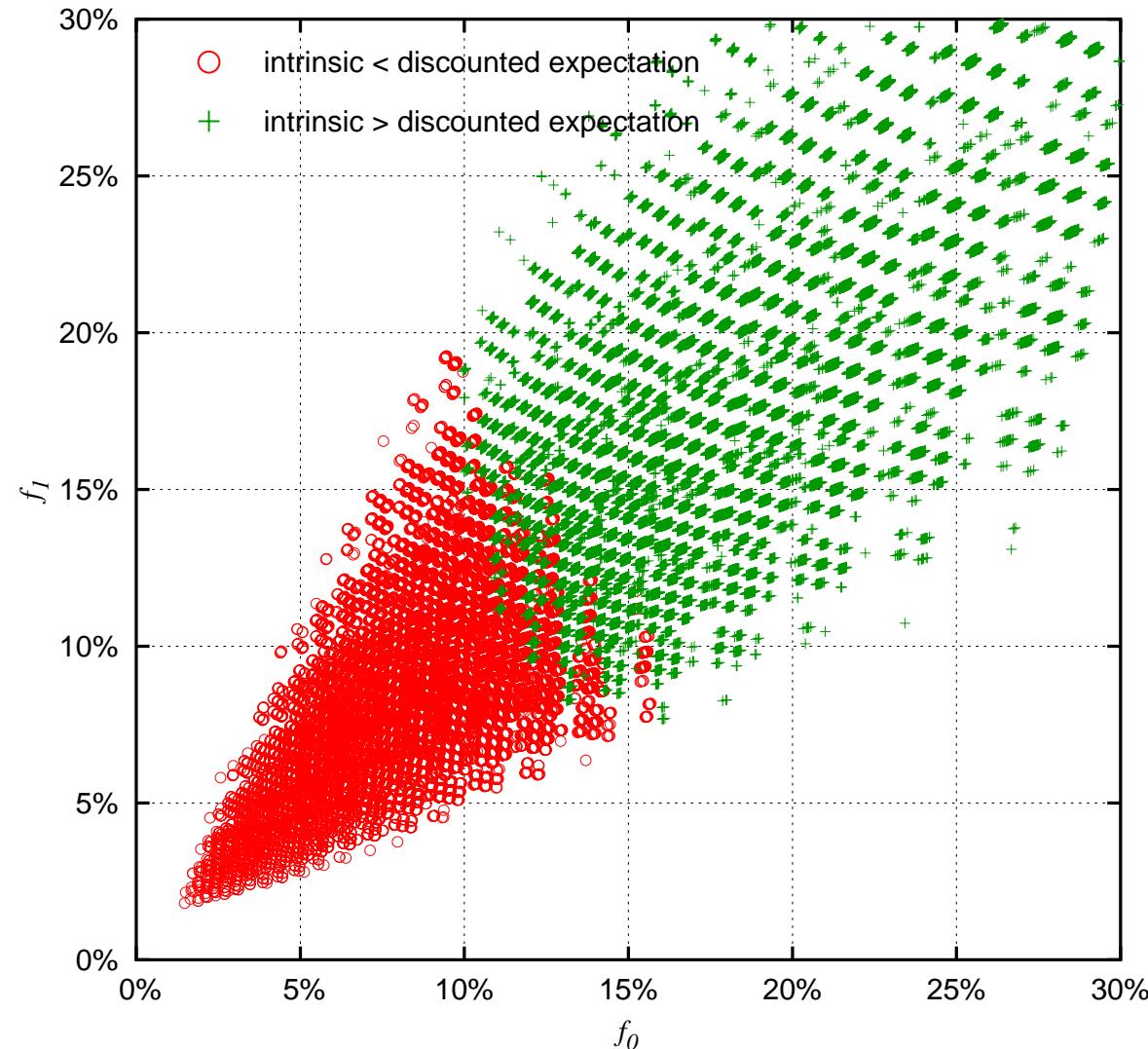


Figure XII.5. Bermudan swaption exercise domain computed with a non-recombining tree in Libor/Libor projection.

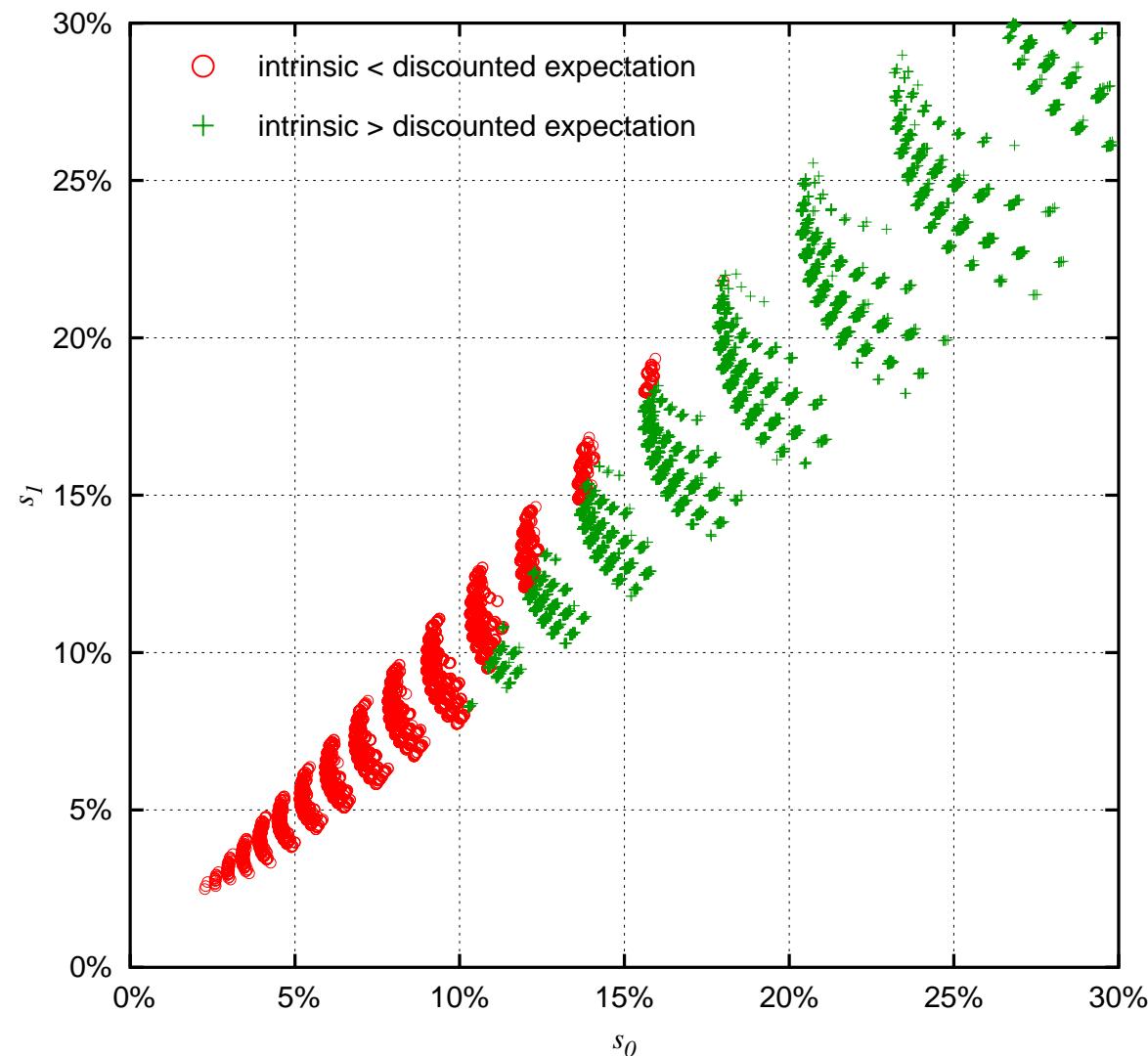


Figure XII.6. Bermudan swaption exercise domain computed with a non-recombining tree in swaprate/swaprate projection.

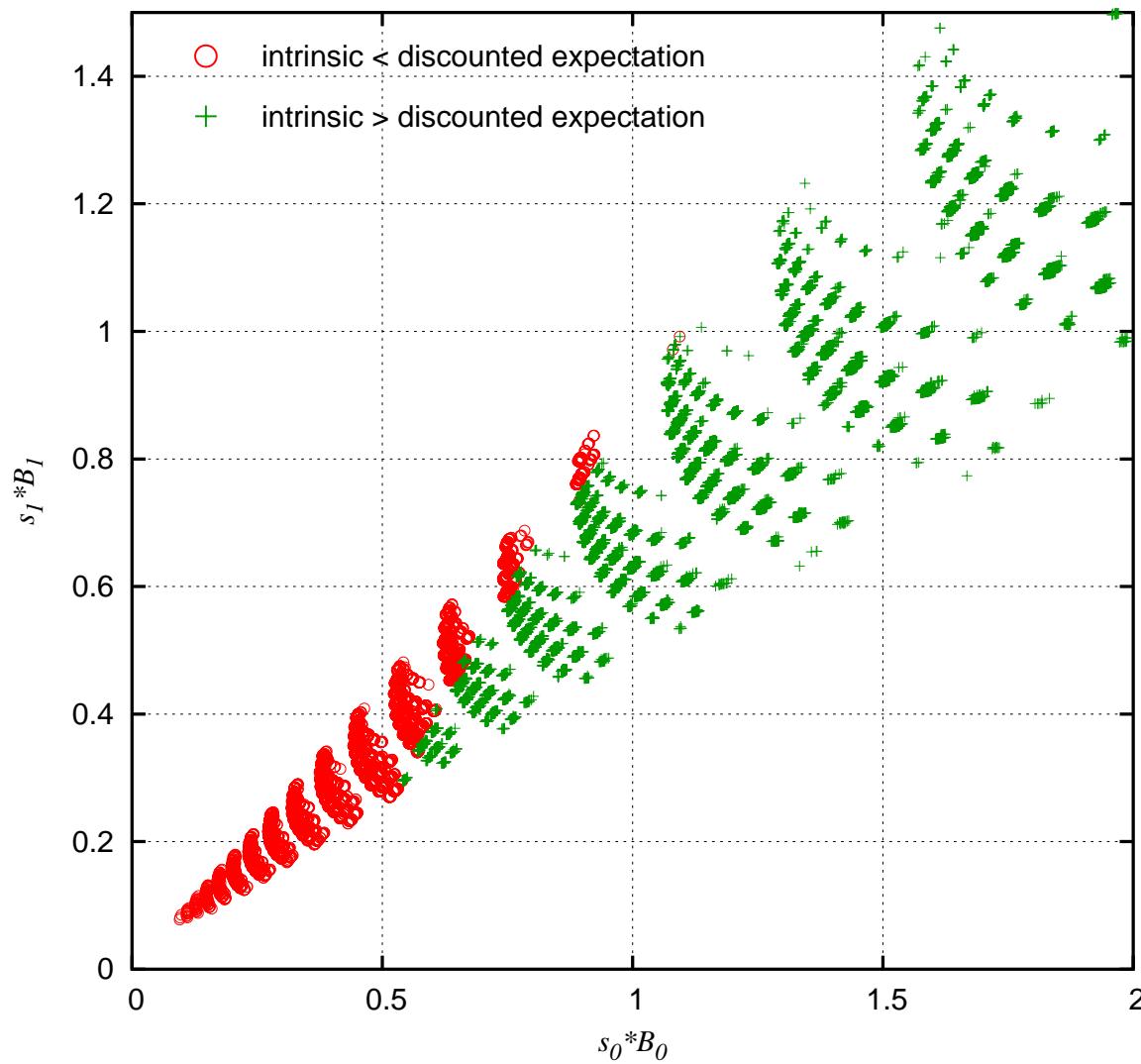


Figure XII.7. Bermudan swaption exercise domain computed with a non-recombining tree in floating-leg/floating-leg projection.

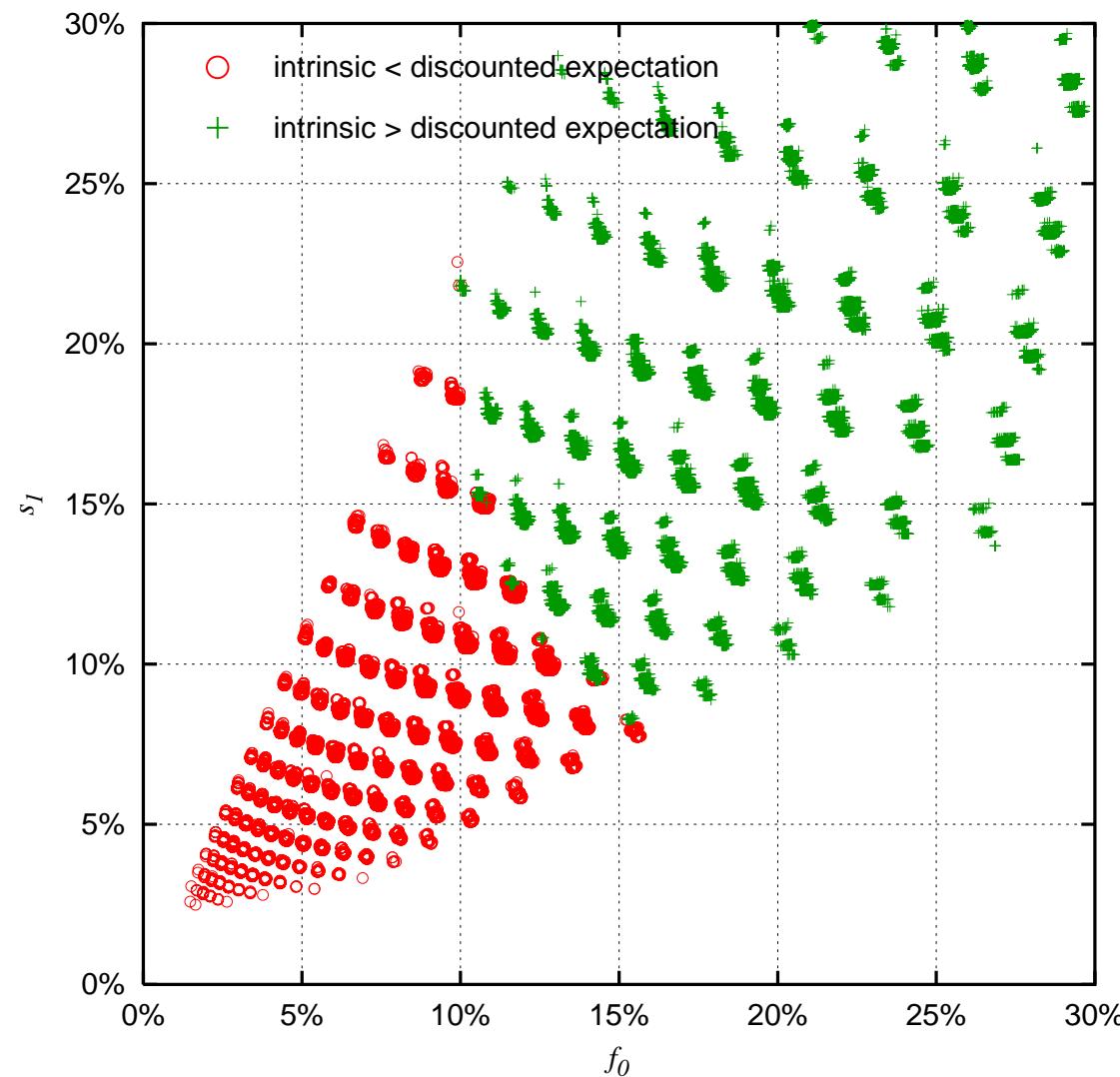


Figure XII.8. Bermudan swaption exercise domain computed with a non-recombining tree in floating-leg/swaprate projection.

Bermudan swaption exercise boundary parametrisation

Taking into account all of the heuristic observations about the shape of the exercise boundary in various projections for many different shapes of the yield curve and volatility structures, the following function can be chosen as the basis for the subsequent exercise decision strategy in the Monte Carlo simulation:

$$\mathcal{E}_i(f(t_i)) = \phi_{\pm} \cdot \left(f_i(t_i) - \left[p_{i1} \cdot \frac{s_{i+1}(0)}{s_{i+1}(t_i) + p_{i2}} + p_{i3} \right] \right) \quad (145)$$

with

$$\phi_{\pm} = \begin{cases} +1 & \text{for payer's swaptions} \\ -1 & \text{for receiver's swaptions} \end{cases} \quad (146)$$

This function is hyperbolic in s_{i+1} and depends on three coefficients, the initial (i.e. at the calendar time of evaluation or inception of the derivative contract) value of $f_i(0)$ and $s_{i+1}(0)$, and their respective evolved values as given by the simulation procedure.

For non-standard Bermudan swaptions that have payments in between exercise dates, we use the shortest swap rate from t_i to the next exercise time instead of f_i . The parametric exercise decision given an evolved yield curve is then simply to exercise if $\mathcal{E}_i > 0$.

At the very last exercise opportunity at time t_{n-1} we have exact knowledge if exercise is optimal, namely when the residual swap is in the money. This easily integrates into the parametric description given by equation (145) by setting $p_{(n-1)1}$ and $p_{(n-1)2}$ to zero and $p_{(n-1)3}$ to the strike:

$$p_{(n-1)1} = 0 \quad p_{(n-1)2} = 0 \quad p_{(n-1)3} = K \quad (147)$$

Note: For Bermudan swaptions the above shown method has been tested by *upper-bound methods* [JT02, Mey03a] and was found to be highly accurate. It has also been compared with other approaches ([Mey03b], page 202), and been found to be very competitive overall, both with respect to execution time and with respect to accuracy.

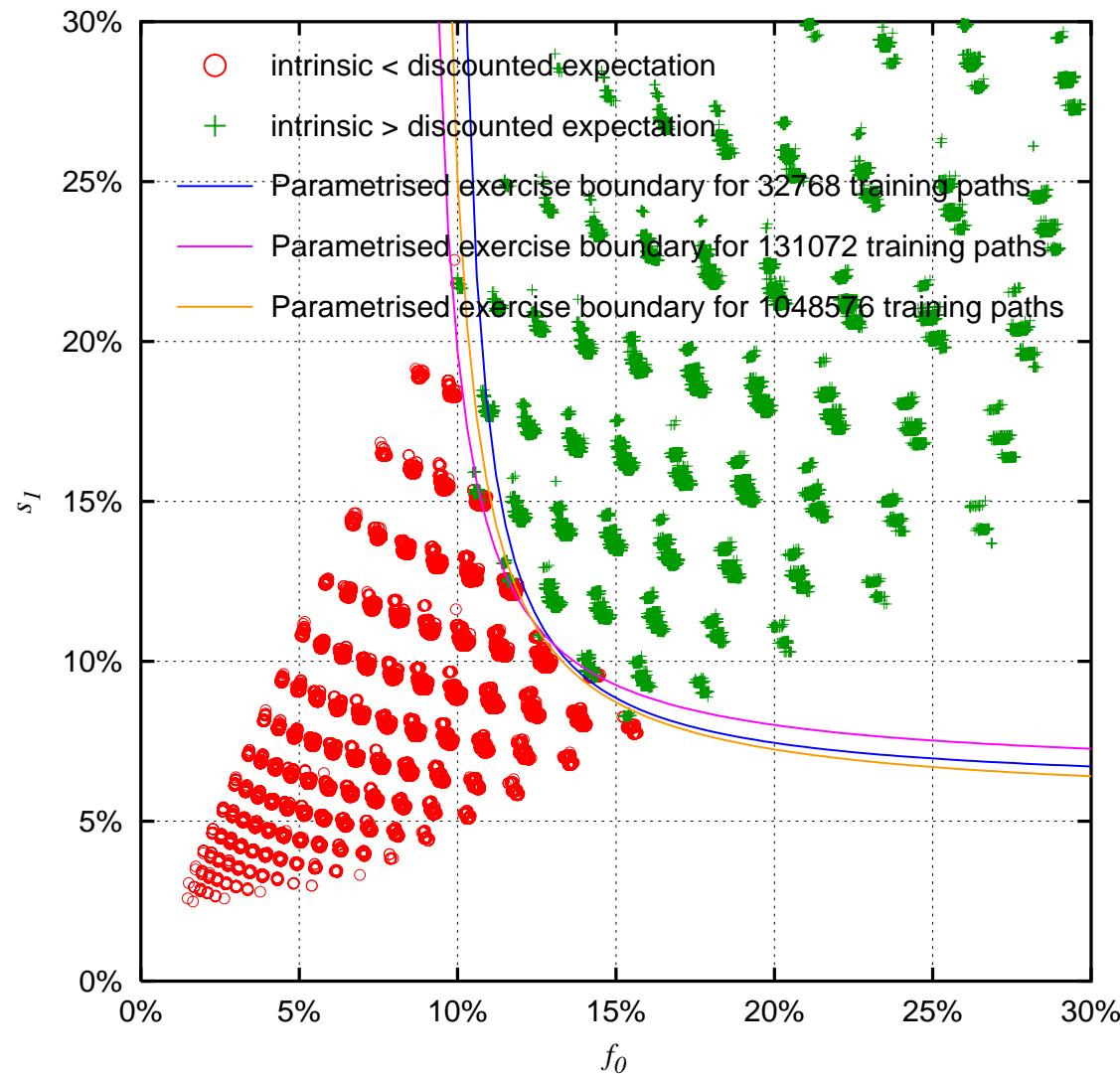


Figure XII.9. The exercise domain in the f_i - s_{i+1} projection of the evolved yield curve at $t_i = 2$, together with the parametrised exercise boundary resulting from training with different sizes of the training set.

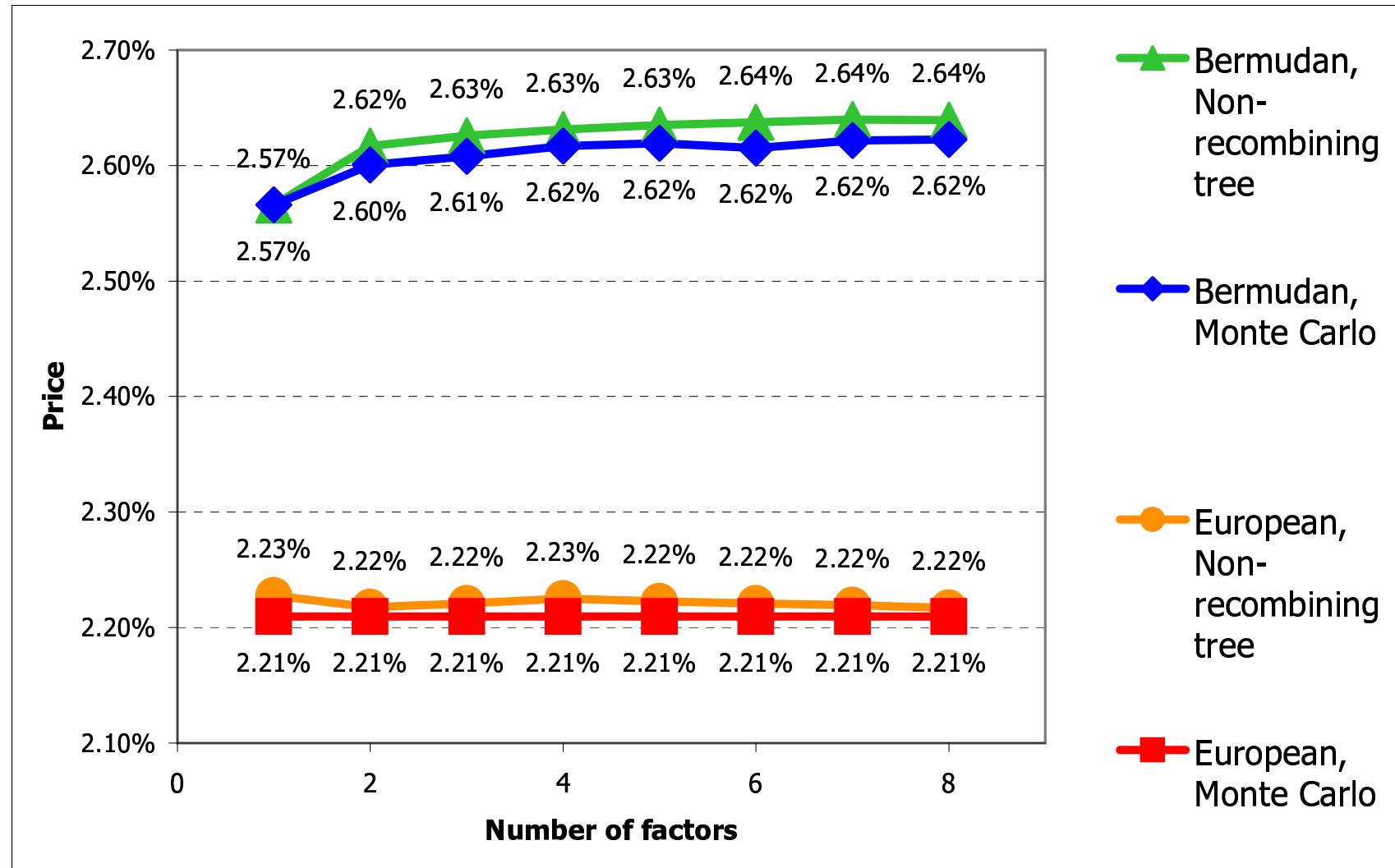


Figure XII.10. Bermudan swaption prices from the Monte Carlo model in comparison to those obtained from a non-recombining tree model for a 6-non-call-2 semi-annual payer's swaption.

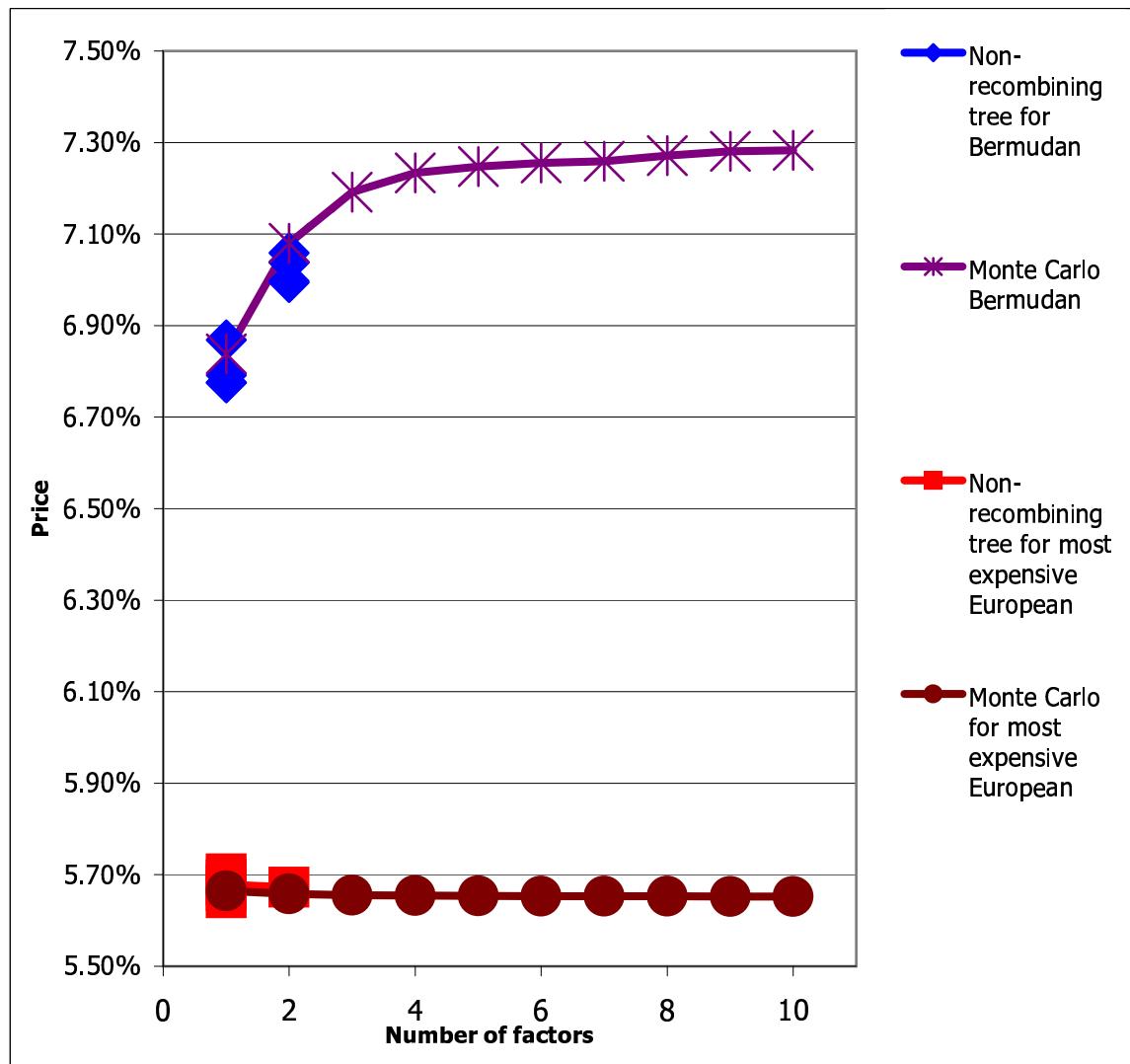


Figure XII.11. 15-non-call-5 annual payer's

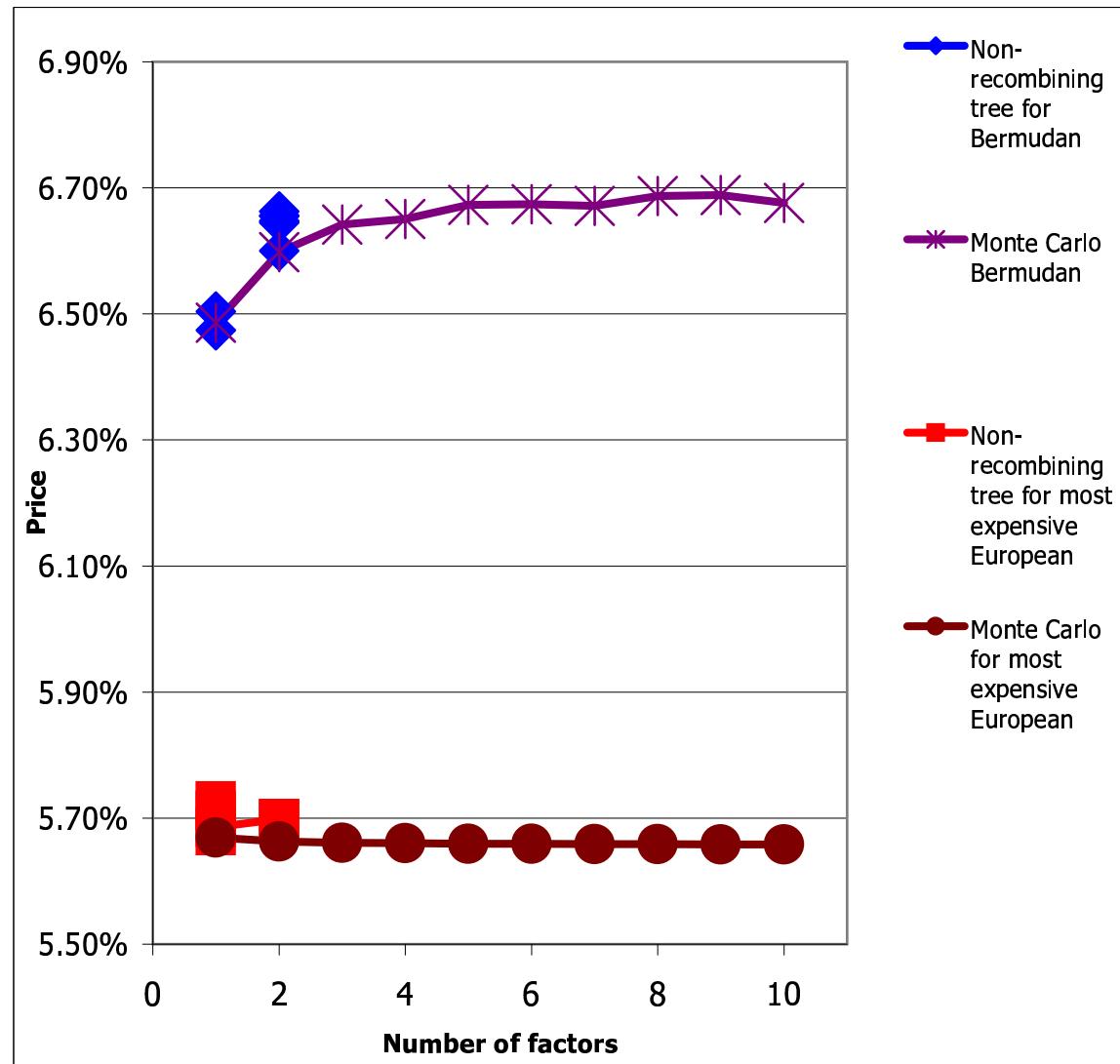


Figure XII.12. 15-non-call-5 annual receiver's

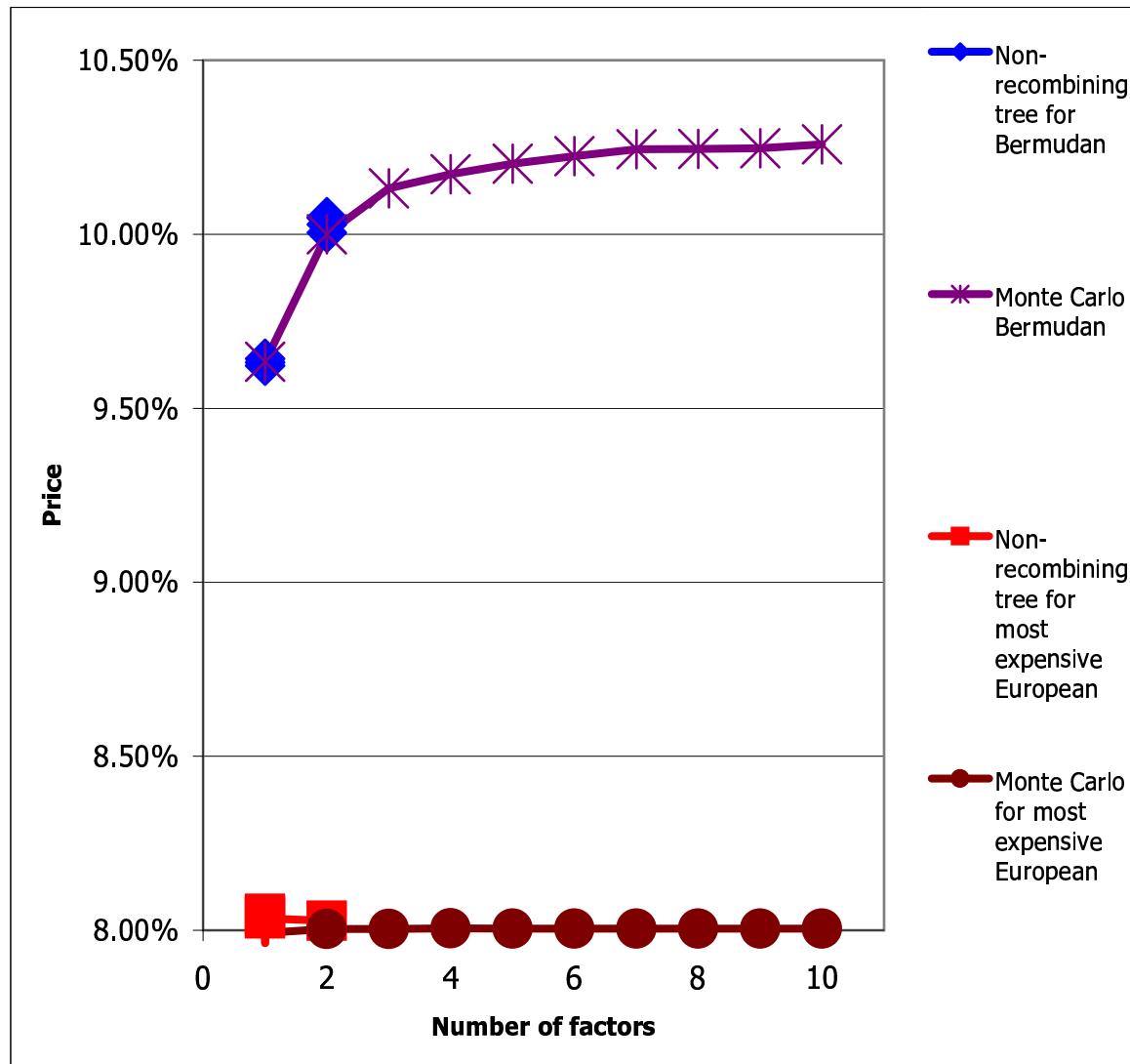


Figure XII.13. 15-non-call-5 annual payer's for steeply upwards sloping yield curve

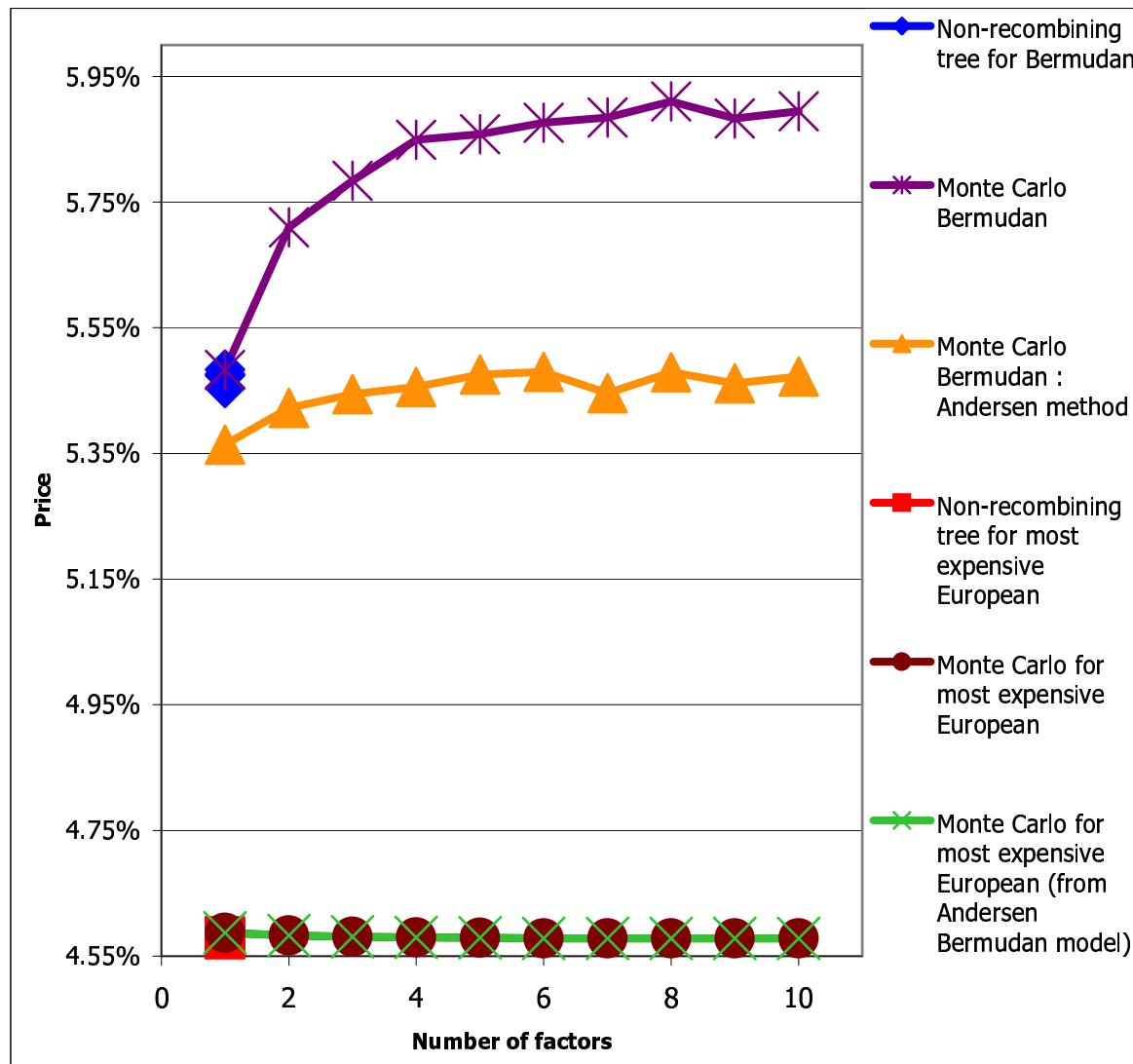


Figure XII.14. 20-non-call-10 semi-annual payer's in comparison to Andersen's method I [And00]

A useful parametric form for two-dimensional exercise decision function design:

$$\mathcal{E}(x, y) = a - y + c(x - b) + g\sqrt{(c(x - b))^2 + d^2} \quad (148)$$

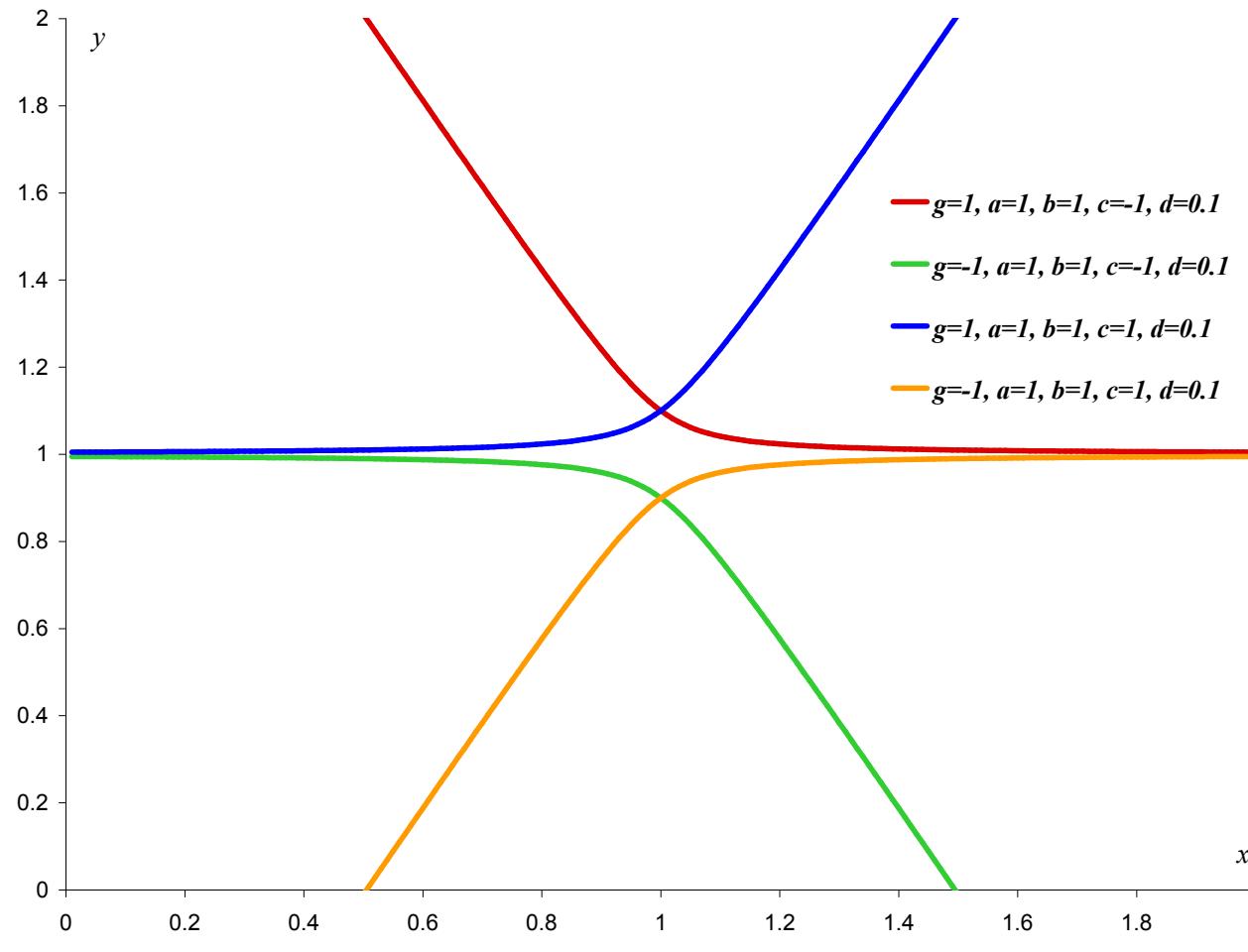


Figure XII.15.



Hints when computing delta or vega with Bermudan Monte Carlo methods:

- When using an exercise boundary, do not re-optimize the boundary. This is asymptotically correct and reduces the Monte Carlo noise both on delta and vega.
- Better even, if your implementation allows for it, enforce exercise on the same (modified for delta or vega purposes) path at the same point in time [Pit03b]



XIII. Cross-currency Libor market modelling

(The formulæ shown in this section are the result of joint work with Atsushi Kawai.)

Define $Q(t)$ as the domestic spot value of one foreign currency unit.

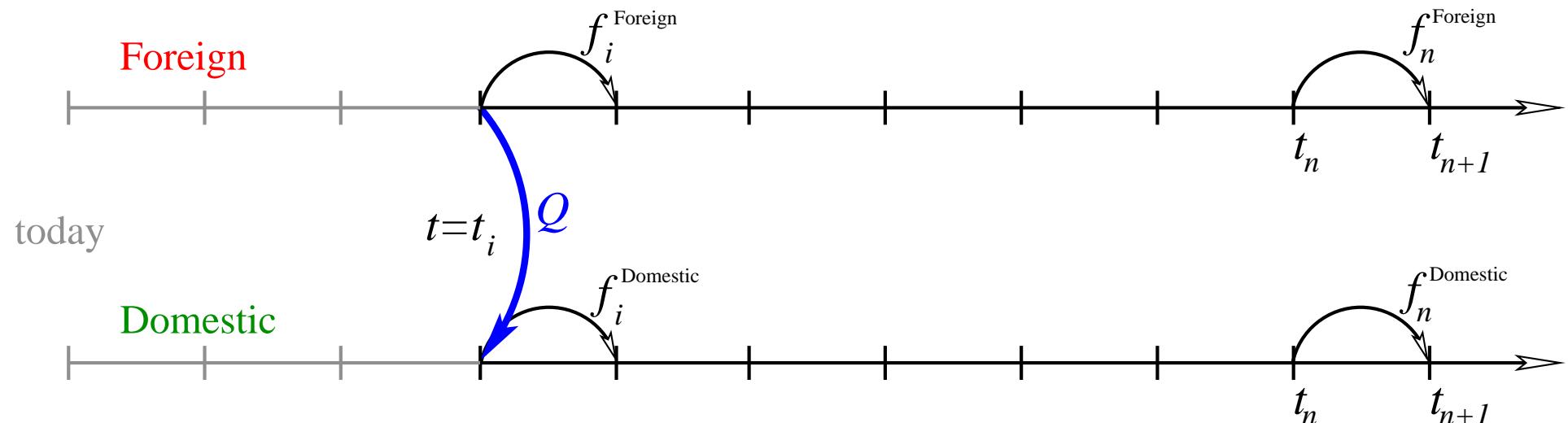


Figure XIII.1.

In the discretely rolled up money market account measure, the drifts of all domestic rates are for a cross-currency Libor market model the same as for a single currency model, i.e. they are given by equation (41).

In order to allow for the FX implied volatility profile to show a pronounced skew, we permit the spot FX process to be governed by a displaced diffusion:

$$\frac{d(Q + s_Q)}{Q + s_Q} = \mu_Q^{\text{DD}} dt + \sigma_Q^{\text{DD}} d\widetilde{W}_Q \quad (149)$$

The drift of the spot FX process in the spot measure is given by¹¹

$$\mu_Q^{\text{DD}} = \frac{Q}{Q + s_Q} (r^{\text{Domestic}} - r^{\text{Foreign}}) \quad (150)$$

The drift of foreign forward rates incurs a *quanto correction term*:

$$\mu_{f_j^{\text{For}}}^{\text{DD}} (f_j^{\text{For}}, s^{\text{For}}, t) = \sigma_{f_j^{\text{For}}}^{\text{DD}} \cdot \left[\sum_{k=i[t]+1}^j \frac{(f_k^{\text{For}} + s_k^{\text{For}}) \tau_k}{1 + f_k^{\text{For}} \tau_k} \sigma_{f_k^{\text{For}}}^{\text{DD}} \rho_{f_j^{\text{For}} f_k^{\text{For}}} - \frac{Q + s_Q}{Q} \sigma_Q^{\text{DD}} \rho_{Q f_k^{\text{For}}} \right]. \quad (151)$$

¹¹ Note that this is only formal notation since we do not really have well defined domestic and foreign spot interest rates r^{Domestic} and r^{Foreign} .



Cross-currency correlation specifications

It is in general possible to define arbitrary instantaneous term structures of correlation between domestic rates, foreign rates, and the spot FX process, as long as they are positive semi-definite at all points in time.

A simple scheme for cross currency correlation specifications that appears to work for all reasonable choices of parameters is to introduce only

three new correlation coefficients,

namely:

- ρ_{FD} . Correlation of domestic and foreign short rates.
- ρ_{QF} . Correlation of spot FX rate and foreign short rate.
- ρ_{QD} . Correlation of spot FX rate and domestic short rate.



Then, set

$$\rho_{f_j^F, f_k^D}(t) := \rho_{f_{i[t]}^F, f_j^F}(t) \cdot \rho_{FD} \cdot \rho_{f_{i[t]}^D, f_k^D}(t) . \quad (152)$$

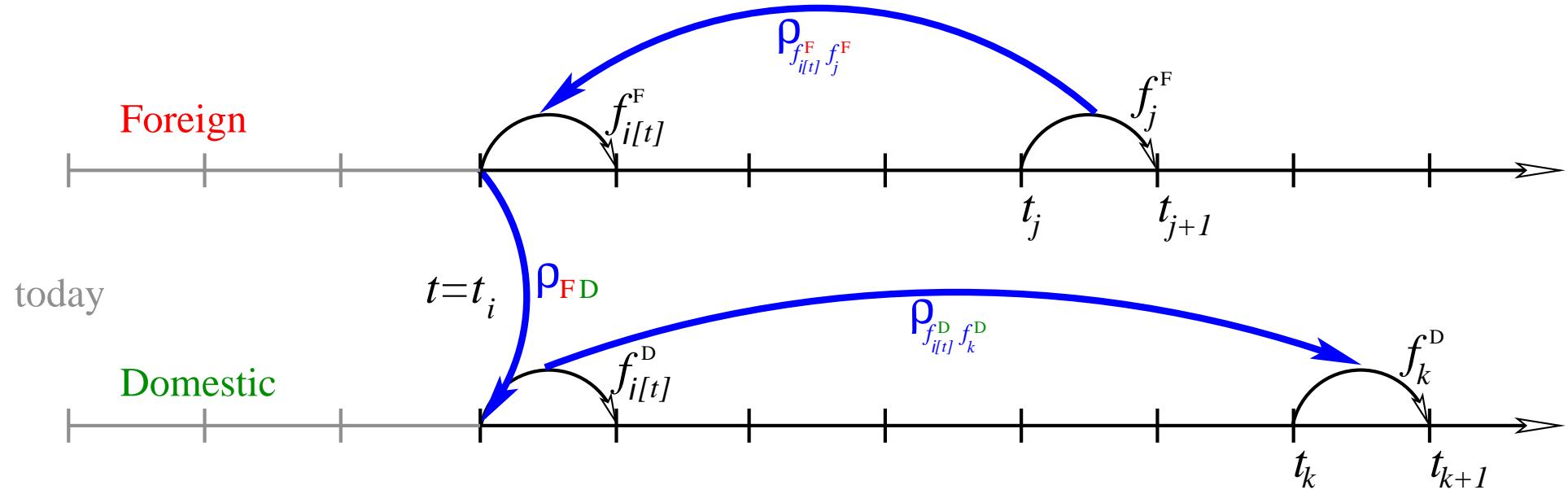


Figure XIII.2.

Equally, set

$$\rho_{Q, f_j^D}(t) := \rho_{QD} \cdot \rho_{f_{i[t]}^D, f_j^D}(t) \quad (153)$$

and

$$\rho_{Q, f_j^F}(t) := \rho_{QF} \cdot \rho_{f_{i[t]}^F, f_j^F}(t) . \quad (154)$$



The predictor-corrector scheme in the cross-currency Libor market model

- Domestic forward rates are handled as in a single currency Libor market model.
- Foreign forward rates are handled in *exactly the same way*.
- Spot FX rates require special attention.

Define

$$x_Q(t) := Q(t) + s_Q. \quad (155)$$

A formal solution of (149) and (150) is

$$x_Q(t_b) = \underbrace{x_Q(t_a) e^{-\frac{1}{2} \int_{t_a}^{t_b} \sigma_Q^{\text{DD}} du + \int_{t_a}^{t_b} \sigma_Q^{\text{DD}} d\widetilde{W}_Q(u)}}_{=: x_Q^{\text{GBM}}(t_b)} \cdot \underbrace{e^{\int_{x_Q(t_a)}^{x_Q(t_b)} (r^{\text{Domestic}} - r^{\text{Foreign}}) du}}_{\text{stochastic drift term}} \quad (156)$$

Recall that, in a simplified way, we can view the predictor-corrector scheme as the approximation of stochastic drift terms by a given frozen state of the yield curve, executed twice with different yield curves, and averaged.

For a frozen state of the yield curves and the FX rate,

$$\mathbf{x}^\top := (x_Q, \mathbf{f}^{\text{Domestic}}{}^\top + \mathbf{s}^{\text{Domestic}}{}^\top, \mathbf{f}^{\text{Foreign}}{}^\top + \mathbf{s}^{\text{Foreign}}{}^\top), \quad (157)$$

the drift term can be re-expressed:

$$\begin{aligned} e^{t_a \int_{x_Q}^{t_b} (r^{\text{Domestic}} - r^{\text{Foreign}}) du} &= e^{\frac{Q}{x_Q} \int_{t_a}^{t_b} (r^{\text{Domestic}} - r^{\text{Foreign}}) du} \\ &= \left(e^{\int_{t_a}^{t_b} (r^{\text{Domestic}} - r^{\text{Foreign}}) du} \right)^{\frac{Q}{x_Q}} \\ &= \left(\frac{P^{\text{Foreign}}(\mathbf{x})[t_a, t_b]}{P^{\text{Domestic}}(\mathbf{x})[t_a, t_b]} \right)^{\frac{Q}{x_Q}}. \end{aligned} \quad (158)$$

In general, the state-dependent discount factors in equation (158) may have to be evaluated with the stub formula (36).

The predictor-corrector scheme for the spot FX rate in the cross-currency Libor market model:

$$x_Q^{\text{Predictor}}(t_b) = x_Q^{\text{GBM}}(t_b) \cdot \left(\frac{P^{\text{Foreign}}(\mathbf{x}(t_b))[t_a, t_b]}{P^{\text{Domestic}}(\mathbf{x}(t_b))[t_a, t_b]} \right)^{\frac{x_Q(t_a) - s_Q}{x_Q(t_a)}} \quad (159)$$

$$x_Q^{\text{Corrector}}(t_b) = x_Q^{\text{GBM}}(t_b) \cdot \left(\frac{P^{\text{Foreign}}(\mathbf{x}(t_b))[t_a, t_b]}{P^{\text{Domestic}}(\mathbf{x}(t_b))[t_a, t_b]} \right)^{\frac{x_Q(t_b) - s_Q}{x_Q(t_b)}} \quad (160)$$

$$x_Q^{\text{Predictor-Corrector}}(t_b) = \sqrt{x_Q^{\text{Predictor}}(t_b) \cdot x_Q^{\text{Corrector}}(t_b)} . \quad (161)$$



XIV. Calibration of FX volatilities in a cross-currency Libor market model

(The formulæ shown in this section have been derived by Atsushi Kawai [KJ07].)

To fully calibrate a cross-currency Libor market model, both the domestic and the foreign yield curve must be calibrated first to their respective interest rate market information.

The final stage is then the calibration of instantaneous FX factor volatilities such that FX plain vanilla option prices meet their respective market values.

This last stage is the most difficult one since FX implied volatilities as predicted by the model depend on all of:-

- Levels of both yield curves.
- Volatilities in both currencies' interest rate markets.
- Correlations within both currencies' interest rate markets.
- Correlations between the FX factor and both currencies' interest rate markets.
- FX factor volatilities

Atsushi Kawai [KJ07] derived analytical formulæ for plain vanilla options using asymptotic approximations based on Itô-Taylor expansions.

These can be used in a fast root finding procedure to find the term structure of instantaneous FX factor volatility that calibrates the model to the market.

For a plain vanilla FX option on one foreign currency unit whose value today is Q domestic currency units, expiring and settling at the payment time T_N of the $(N - 1)$ -th discrete forward rate, we have the

asymptotic FX option formulæ in the cross-currency Libor market model :

- Call option with expiry T_N

$$\approx P^{\text{Domestic}}[0, T_N] \cdot \left(x_Q \Phi(h) - \varkappa_Q \Phi \left(h - \hat{\sigma}_{Q_{T_N}}^{\text{DD}} \sqrt{T_N} \right) \right) \quad (162)$$

- Put option with expiry T_N

$$\approx P^{\text{Domestic}}[0, T_N] \cdot \left(\varkappa_Q \Phi \left(-h + \hat{\sigma}_{Q_{T_N}}^{\text{DD}} \sqrt{T_N} \right) - x_Q \Phi(-h) \right) \quad (163)$$

where

$$Q_{T_N} = Q \cdot \frac{P^{\text{Foreign}}[0, T_N]}{P^{\text{Domestic}}[0, T_N]} \quad (164)$$

$$x_Q = Q_{T_N} + s_Q \quad (165)$$

$$\varkappa_Q = K + s_Q \quad (166)$$

$$h = \frac{\ln(x_Q/\varkappa_Q) + \frac{1}{2} \hat{\sigma}_{Q_{T_N}}^{\text{DD}} T_N}{\hat{\sigma}_{Q_{T_N}}^{\text{DD}} \sqrt{T_N}}, \quad (167)$$

and

$$\begin{aligned} \hat{\sigma}_{Q_{T_N}}^{\text{DD}} &= \frac{Q_{T_N}^2}{x_Q^2} \cdot \frac{v_1}{T_N} \cdot \left[1 + \left(\frac{1}{x_Q} - 2c_1 \right) g_0 + \left(c_2 + \frac{11}{12x_Q^2} - \frac{2c_1}{x_Q} \right) g_0^2 \right. \\ &\quad \left. + \left(c_3 + \frac{1}{12x_Q^2} + 2c_4 \right) Q_{T_N}^2 v_1 \right]. \end{aligned} \quad (168)$$

The involved coefficients are defined below. All quantities are defined as seen with today's yield curve.

Superscripts $(\cdot)^D$ and $(\cdot)^F$ stand for domestic and foreign values, respectively.

$$g_0 := Q_{T_N} - K \quad (169)$$

$$\gamma := 1 + \frac{1}{2} s_Q \left(\frac{1}{Q_{T_N}} + \frac{1}{Q} \right) \quad (170)$$

$$w_k^D := \frac{(f_k^D + s_k^D) \tau_k}{1 + f_k^D \tau_k} \quad y_k^D := w^D \cdot \frac{1 - s_k^D \tau_k}{1 + f_k^D \tau_k} \quad (171)$$

$$w_k^F := \frac{(f_k^F + s_k^F) \tau_k}{1 + f_k^F \tau_k} \quad y_k^F := w^F \cdot \frac{1 - s_k^F \tau_k}{1 + f_k^F \tau_k} \quad (172)$$

$$\begin{aligned}
v_1 &:= \gamma^2 c_{QQ} + \sum_{k=0}^{N-1} \sum_{j=0}^{N-1} w_k^D w_j^D c_{f_k^D f_j^D} + \sum_{k=0}^{N-1} \sum_{j=0}^{N-1} w_k^F w_j^F c_{f_k^F f_j^F} + 2\gamma \sum_{k=0}^{N-1} w_k^D c_{Q f_k^D} \\
&\quad - 2\gamma \sum_{k=0}^{N-1} w_k^F c_{Q f_k^F} - 2 \sum_{k=0}^{N-1} \sum_{j=0}^{N-1} w_k^D w_j^F c_{f_k^D f_j^F}. \tag{173}
\end{aligned}$$

$$v_{2,k}^D := \gamma c_{Q f_k^D} + \sum_{j=0}^{N-1} w_j^D c_{f_k^D f_j^D} - \sum_{j=0}^{N-1} w_j^F c_{f_k^D f_j^F} \tag{174}$$

$$v_{2,k}^F := \gamma c_{Q f_k^F} + \sum_{j=0}^{N-1} w_j^D c_{f_k^F f_j^D} - \sum_{j=0}^{N-1} w_j^F c_{f_k^F f_j^F} \tag{175}$$

$$v_{3,k}^{\text{D}} := - \sum_{j=k+1}^{N-1} w_j^{\text{D}} c_{f_k^{\text{D}} f_j^{\text{D}}} \quad (176)$$

$$v_{3,k}^{\text{F}} := -\gamma c_{Qf_k^{\text{F}}} + \sum_{j=0}^k w_j^{\text{F}} c_{f_k^{\text{F}} f_j^{\text{F}}} - \sum_{j=0}^{N-1} w_j^{\text{D}} c_{f_k^{\text{F}} f_j^{\text{D}}} \quad (177)$$

$$v_4 := \gamma c_{QQ} + \sum_{j=0}^{N-1} w_j^{\text{D}} c_{Qf_j^{\text{D}}} - \sum_{j=0}^{N-1} w_j^{\text{F}} c_{Qf_j^{\text{F}}} \quad (178)$$

$$v_5 := - \sum_{j=0}^{N-1} w_j^{\text{D}} c_{Qf_j^{\text{D}}} \quad (179)$$

$$c_{QQ} := \int_0^{T_N} \sigma_Q^{\text{DD}}{}^2(u) \, du \quad (180)$$



$$c_{Qf_k^N} := \int_0^{T_N} \sigma_Q^{\text{DD}}(u) \cdot \rho_{Qf_k^N}(u) \cdot \sigma_{f_k^N}^{\text{DD}}(u) \, du \quad \text{with } N \in \{\text{F}, \text{D}\} \quad (181)$$

$$c_{f_k^N f_j^{\text{D}}} := \int_0^{T_N} \sigma_{f_k^N}^{\text{DD}}(u) \cdot \rho_{f_k^N f_j^{\text{D}}}(u) \cdot \sigma_{f_j^{\text{D}}}^{\text{DD}}(u) \, du \quad \text{with } N, \text{D} \in \{\text{F}, \text{D}\} \quad (182)$$

$$c_1 := \frac{1}{2Q_{T_N}v_1^2} \left[v_1^2 - \gamma(\gamma-1)v_4^2 + \sum_{k=0}^{N-1} y_k^{\text{D}} v_{2,k}^{\text{D}} - \sum_{k=0}^{N-1} y_k^{\text{F}} v_{2,k}^{\text{F}} \right] \quad (183)$$

$$c_2 := -5c_1^2 + 2g_1 + 2g_3 + d_1 \quad (184)$$

$$c_3 := 3c_1^2 - 2g_1 + 2g_2 - 2g_3 - d_1 + d_2 \quad (185)$$

$$c_4 := \frac{1}{2Q_{T_N}^2 v_1^2} \left[\gamma(\gamma-1)v_4 v_5 + \sum_{k=0}^{N-1} y_k^{\text{D}} v_{2,k}^{\text{D}} v_{3,k}^{\text{D}} - \sum_{k=0}^{N-1} y_k^{\text{F}} v_{2,k}^{\text{F}} v_{3,k}^{\text{F}} \right] \quad (186)$$

$$g_1 := \frac{1}{6Q_{T_N}^2} \quad (187)$$

$$g_2 := \frac{1}{2Q_{T_N}^2 v_1^2} \left[+ \gamma (\gamma - 1) v_4 (\gamma c_{QQ} - v_4) + \sum_{k=0}^{N-1} y_k^D v_{2,k}^D - \sum_{k=0}^{N-1} y_k^F v_{2,k}^F \right] \quad (188)$$

$$g_3 := \frac{1}{6Q_{T_N}^2 v_1^3} \left[\gamma (\gamma - 1) v_4^2 ((2\gamma - 1)v_4 - 3v_1) + \sum_{k=0}^{N-1} y_k^D v_{2,k}^D (3v_1 + v_{2,k}^D) - \sum_{k=0}^{N-1} y_k^F v_{2,k}^F (3v_1 + v_{2,k}^F) \right] \quad (189)$$



$$\begin{aligned}
 d_1 := & \frac{1}{Q_{T_N}^2 v_1^3} \left[v_1^3 + \gamma^2 (\gamma - 1)^2 v_4^2 c_{QQ} - 2\gamma (\gamma - 1) v_1 v_4^2 \right. \\
 & + 2v_1 \sum_{k=0}^{N-1} \left(y_k^D v_{2,k}^D - y_k^F v_{2,k}^F \right) \\
 & - 2\gamma (\gamma - 1) v_4 \sum_{k=0}^{N-1} \left(y_k^D v_{2,k}^D c_{Qf_k^D} - y_k^F v_{2,k}^F c_{Qf_k^F} \right) \\
 & \left. + \sum_{j=0}^{N-1} \sum_{k=0}^{N-1} \left(\omega_{jk}^{DD} - 2\omega_{jk}^{DF} + \omega_{jk}^{FF} \right) \right]
 \end{aligned} \tag{190}$$

$$\omega_{jk}^{\aleph\beth} := y_j^{\aleph} \cdot v_{2,j}^{\aleph} \cdot c_{f_j^{\aleph} f_k^{\beth}} \cdot v_{2,k}^{\beth} \cdot y_k^{\beth} \quad \text{with} \quad \aleph, \beth \in \{\text{F}, \text{D}\} \tag{191}$$



$$\begin{aligned} d_2 := & \frac{1}{2Q_{T_N}^2 v_1^2} \left[v_1^2 + \gamma^2 (\gamma - 1)^2 c_{QQ}^2 - 2\gamma (\gamma - 1) v_4^2 \right. \\ & + \sum_{k=0}^{N-1} \left(y_k^D v_{2,k}^D - y_k^F v_{2,k}^F \right) - \gamma (\gamma - 1) \sum_{k=0}^{N-1} \left(y_k^D c_{Qf_k}^2 - y_k^F c_{Qf_k}^2 \right) \\ & \left. + \sum_{k=0}^{N-1} \sum_{j=0}^{N-1} \left(y_j^D c_{f_j^D f_k}^2 y_k^D - 2y_j^D c_{f_j^D f_k}^2 y_k^F + y_j^F c_{f_j^F f_k}^2 y_k^F \right) \right] \end{aligned} \quad (192)$$

Finally: some numerical results showing the accuracy of the analytical formulæ by comparison with Monte Carlo valuations.

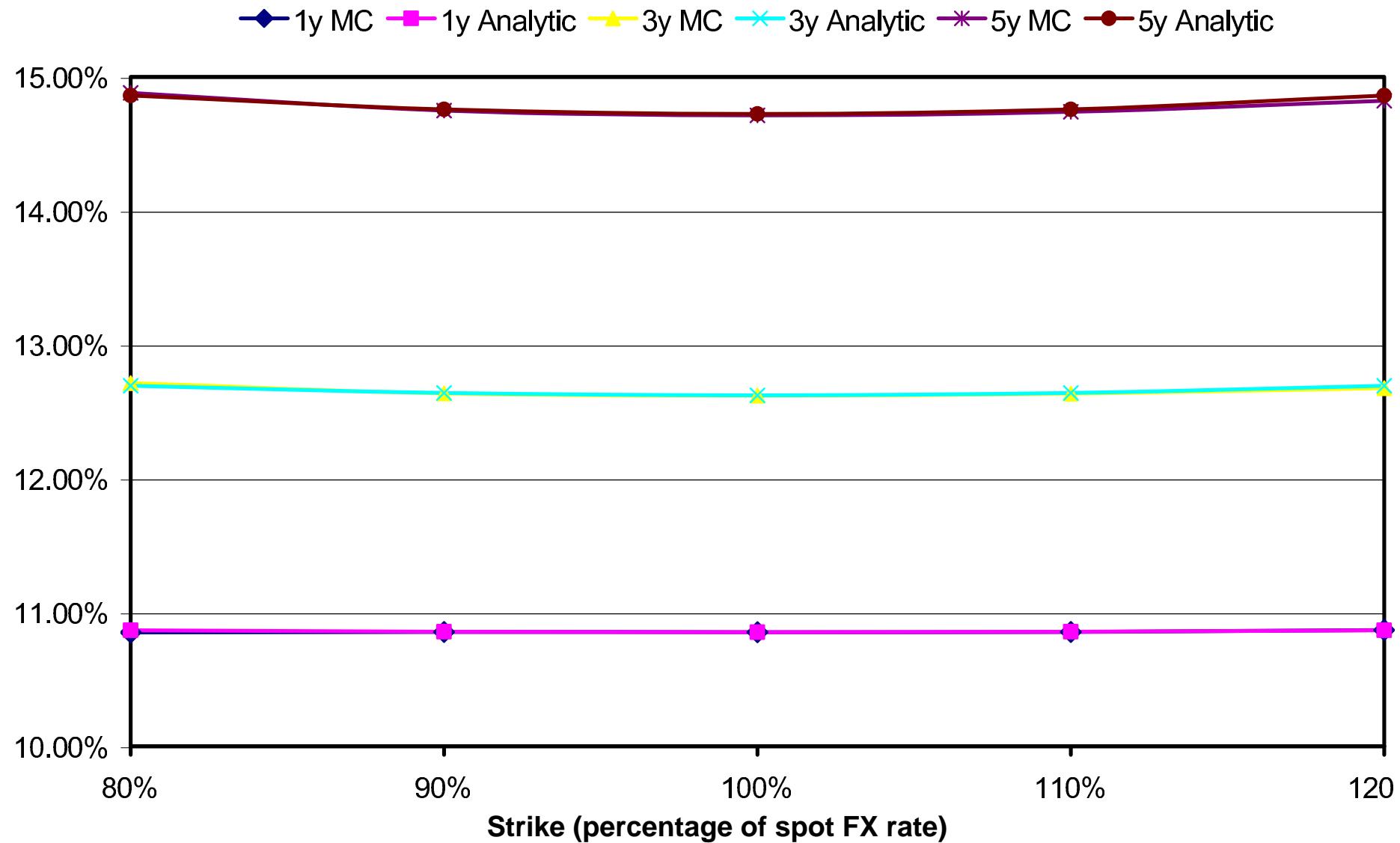


Figure XIV.1. Numerical and analytical implied volatilities for different maturities.

$$Q = 1, f_i^D = f_i^F = 5\%, s_Q = s_i^D = s_j^F = 0, \sigma_Q = 10\%, \sigma_i^D = \sigma_i^F = 40\%, \rho_{f_i^D, f_j^D} = \rho_{f_i^F, f_j^F} = 1, \rho_{Q, f_j^D} = 0.3, \rho_{Q, f_j^F} = -0.3, \rho_{f_j^D, f_j^F} = 0.3$$

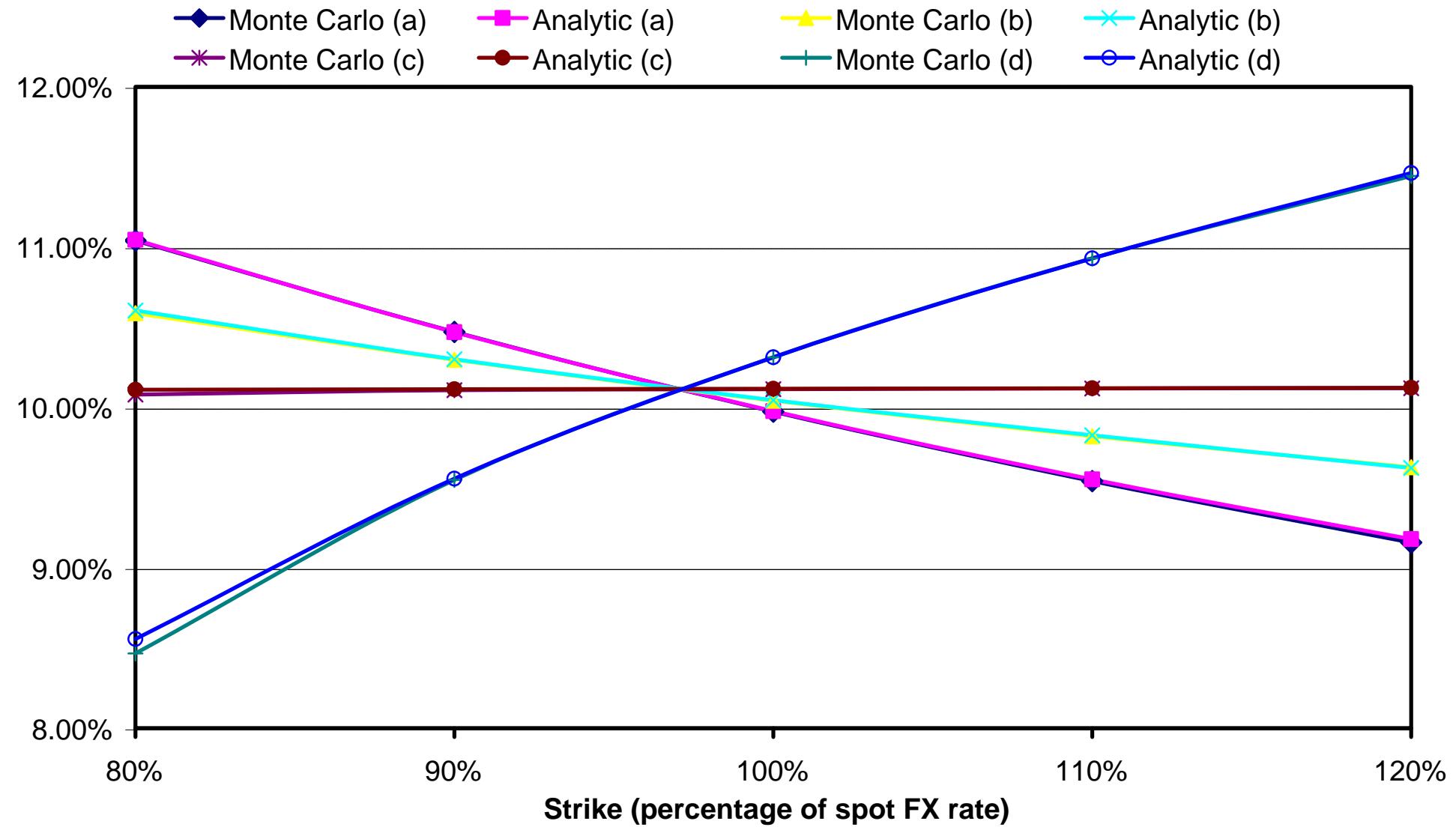


Figure XIV.2. Numerical and analytical 1 year implied volatilities with different FX skew settings:

(a) $s_Q = 8 \log_2(10) \cdot Q$ (almost normal), (b) $s_Q = Q$ (similar to square root distribution), (c) $s_Q = 0$ (almost lognormal), (d) $s_Q = -\log_2(3/2) \cdot Q$ (positive skew).

$Q = 1$, $f_i^D = 5\%$, $f_i^F = 2\%$, $s_i^D = s_j^F = 0$, $\sigma_Q^{DD} = 10\% \cdot \frac{Q}{Q+s_Q}$, $\sigma_i^D = 40\%$, $\sigma_i^F = 50\%$, $\rho_{f_i^D f_j^D} = \rho_{f_i^F f_j^F} = e^{-|t_i - t_j|/5}$, $\rho_{f_j^D f_j^F} = 0\%$, $\rho_{Q, f_j^D} = 0$, $\rho_{Q, f_j^F} = 0$

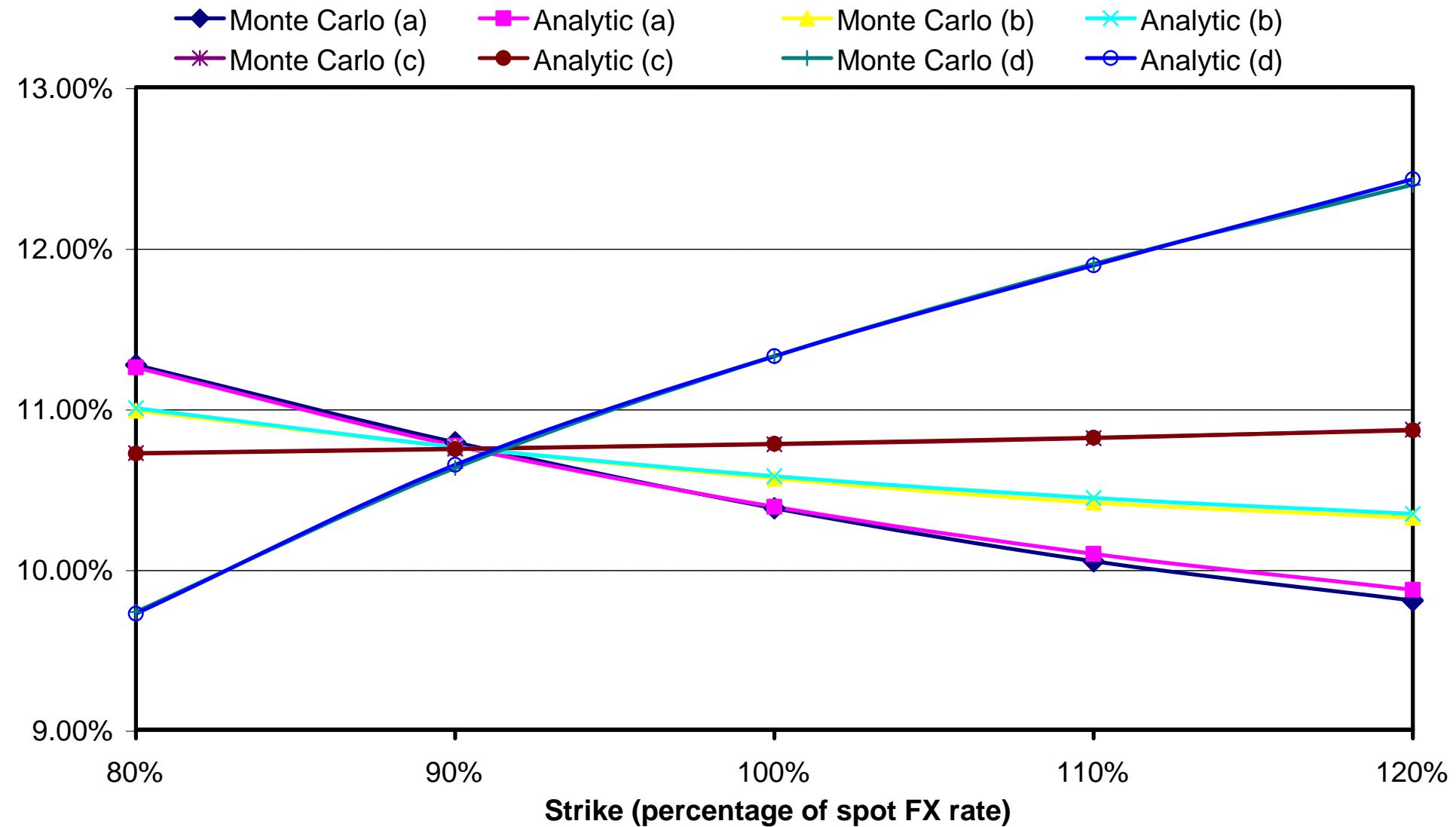


Figure XIV.3. Numerical and analytical 3 year implied volatilities with different FX skew settings:

(a) $s_Q = 8 \log_2(10) \cdot Q$ (almost normal), (b) $s_Q = Q$ (similar to square root distribution), (c) $s_Q = 0$ (almost lognormal), (d) $s_Q = -\log_2(3/2) \cdot Q$ (positive skew).

$Q = 1$, $f_i^D = 5\%$, $f_i^F = 2\%$, $s_i^D = s_j^F = 0$, $\sigma_Q^{DD} = 10\% \cdot \frac{Q}{Q+s_Q}$, $\sigma_i^D = 40\%$, $\sigma_i^F = 50\%$, $\rho_{f_i^D f_j^D} = \rho_{f_i^F f_j^F} = e^{-|t_i-t_j|/5}$, $\rho_{f_i^D, f_j^F} = 0\%$, $\rho_{Q, f_j^D} = 0$, $\rho_{Q, f_j^F} = 0$

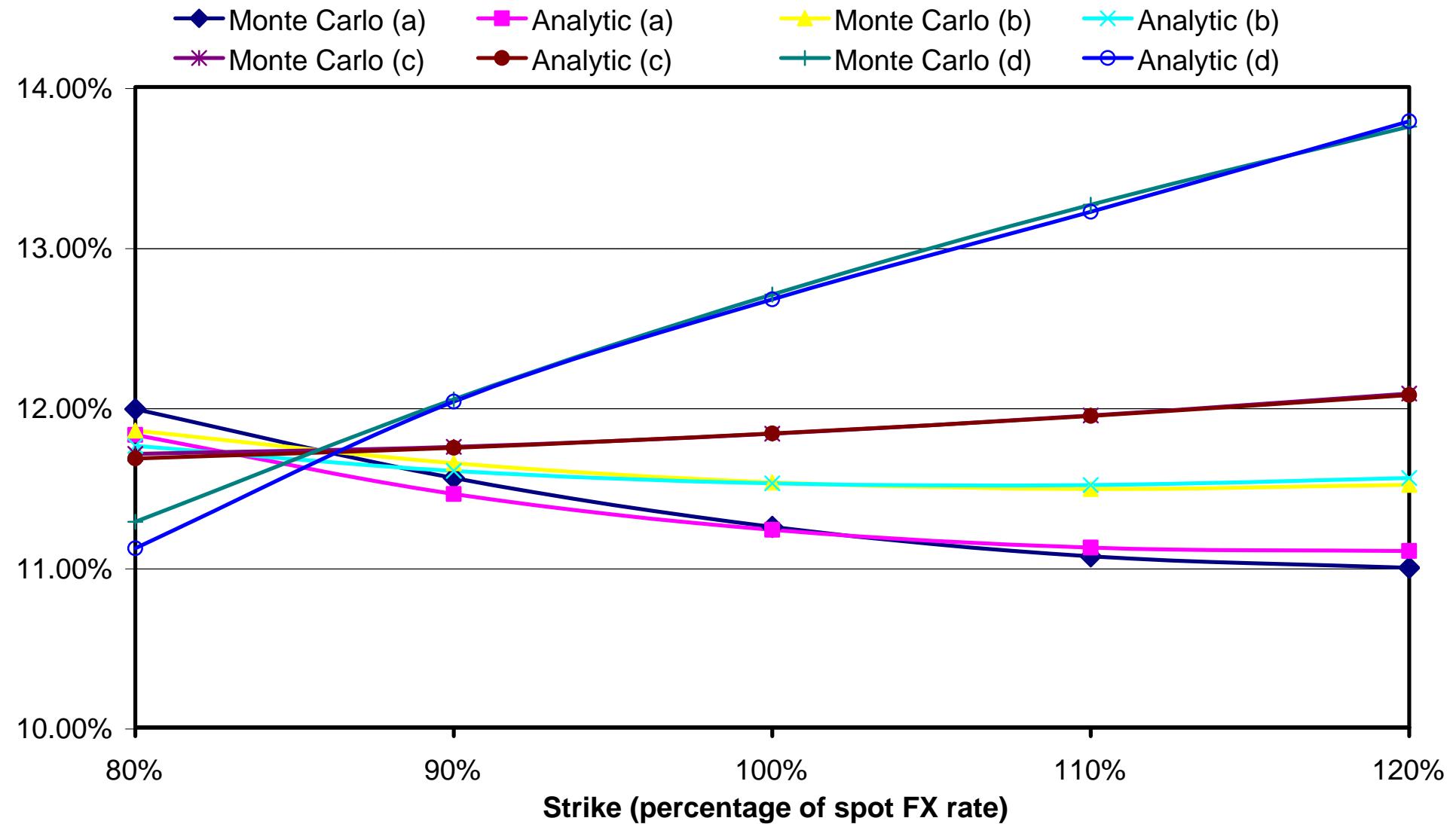


Figure XIV.4. Numerical and analytical 5 year implied volatilities with different FX skew settings:

(a) $s_Q = 8 \log_2(10) \cdot Q$ (almost normal), (b) $s_Q = Q$ (similar to square root distribution), (c) $s_Q = 0$ (almost lognormal), (d) $s_Q = -\log_2(3/2) \cdot Q$ (positive skew).

$Q = 1$, $f_i^D = 5\%$, $f_i^F = 2\%$, $s_i^D = s_j^F = 0$, $\sigma_Q^{DD} = 10\% \cdot \frac{Q}{Q+s_Q}$, $\sigma_i^D = 40\%$, $\sigma_i^F = 50\%$, $\rho_{f_i^D f_j^D} = \rho_{f_i^F f_j^F} = e^{-|t_i - t_j|/5}$, $\rho_{f_j^D f_j^F} = 0\%$, $\rho_{Q, f_j^D} = 0$, $\rho_{Q, f_j^F} = 0$

References

- [AA00] L. Andersen and J. Andreasen. Volatility Skews and Extensions of the Libor Market Model. *Applied Mathematical Finance*, 7(1):1–32, March 2000.
- [Alb72] A. Albert. *Regression and the Moore-Penrose Pseudo-Inverse*. Academic Press, 1972.
- [And00] L. Andersen. A simple approach to the pricing of Bermudan swaptions in the multifactor LIBOR market model. *Journal of Computational Finance*, 3(2):5–32, Winter 1999/2000.
- [BD96] M. Broadie and J. Detemple. American Option Valuation: New Bounds, Approximations, and a Comparison of Existing Methods. *Review of Financial Studies*, 9(4):1211–1250, 1996.
- [Bec80] S. Beckers. The constant elasticity of variance model and its implications for option pricing. *Journal of Finance*, XXXV(3):661–673, June 1980.
- [BG97a] M. Broadie and P. Glasserman. Monte Carlo methods for pricing high-dimensional American options: An overview. *Net Exposure: The Electronic Journal of Financial Risk*, 1(3):15–37, December 1997.
- [BG97b] M. Broadie and P. Glasserman. Pricing American-Style Securities Using Simulation. *Journal of Economic Dynamics and Control*, 21(8–9):1323–1352, 1997.
- [CIR81] J. C. Cox, J. E. Ingersoll, and S. A. Ross. The relation between forward prices and futures prices. *Journal of Financial Economics*, 9(4):321–346, December 1981.
- [CR76] J. C. Cox and S. A. Ross. The valuation of options for alternative stochastic processes. *Journal of Financial Economics*, 3:145–166, March 1976.
- [GZ99] P. Glasserman and X. Zhao. Fast greeks by simulation in forward Libor models. *Journal of Computational Finance*, 3(1):5–39, Fall 1999.
- [HP81] J. M. Harrison and S. Pliska. Martingales and stochastic integrals in the theory of continuous trading. *Stochastic processes and their applications*, 11:215–260, 1981.
- [Jäc02] P. Jäckel. *Monte Carlo methods in finance*. John Wiley and Sons, February 2002.
- [Jäc03] P. Jäckel. Mind the Cap. *Wilmott*, pages 54–68, September 2003. www.jaeckel.org/MindTheCap.pdf.
- [JK05] P. Jäckel and A. Kawai. The Future is Convex. *Wilmott*, pages 2–13, February 2005. www.jaeckel.org/TheFutureIsConvex.pdf.



- [JL03] S. Johnson and H. Lee. Capturing the Smile. *Risk*, 16(3), March 2003. www.numerix.com/news/announcements/0303_smile.pdf.
- [JR00] P. Jäckel and R. Rebonato. The link between caplet and swaption volatilities in a Brace-Gatarek-Musiela/Jamshidian framework: approximate solutions and empirical evidence. *Journal of Computational Finance*, 6(4):41–59, 2003 (submitted in 2000). www.jaeckel.org/LinkingCapletAndSwaptionVolatilities.pdf.
- [JT02] M. Joshi and J. Theis. Bounding Bermudan swaptions in a swap-rate market model. *Quantitative Finance*, 2(5):370–377, October 2002. www.quarchome.org/BermBounds.pdf.
- [Kaw02] A. Kawai. Analytical and Monte Carlo swaption pricing under the forward swap measure. *Journal of Computational Finance*, 6(1):101–111, 2002. www.maths.unsw.edu.au/statistics/files/preprint-2002-12.pdf.
- [Kaw03] A. Kawai. A new approximate swaption formula in the LIBOR market model: an asymptotic expansion approach. *Applied Mathematical Finance*, 10(1):49–74, March 2003. www.maths.unsw.edu.au/statistics/files/preprint-2002-13.pdf.
- [KJ07] A. Kawai and P. Jäckel. An asymptotic FX option formula in the cross currency Libor market model. *Wilmott*, pages 74–84, March 2007. First published in October 2006 at www.jaeckel.org/AnAsymptoticFXOptionFormulaInTheCrossCurrencyLiborMarketModel.pdf.
- [KN97] G. Kirikos and D. Novak. Convexity Conundrums. *Risk*, pages 60–61, March 1997. www.powerfinance.com/convexity.
- [KS91] I. Karatzas and S. E. Shreve. *Brownian motion and Stochastic Calculus*. Springer, 1991.
- [KS98] I. Karatzas and S. E. Shreve. *Methods of Mathematical Finance*. Springer, 1998.
- [LLS] A. Lumsdaine, L. Lee, and J. Siek. The iterative template library. www.osl.iu.edu/research/itl.
- [LS98] F. A. Longstaff and E. S. Schwartz. Valuing American options by simulation: A simple least squares approach. Working paper, The Anderson school, UCLA, 1998.
- [Mat01] K. Matsumoto. Lognormal swap approximation in the Libor market model and its application. *Journal of Computational Finance*, 5(1):107–131, 2001.
- [Mey03a] M. Meyer. Monte Carlo Pricing of Bermudan Options. martingale.berlios.de/topics/Bermudan/Index.html, 2003.
- [Mey03b] M. Meyer. *Monte Carlo simulation with Java and C++*. download.berlios.de/martingale/MCBook-1.2.pdf.gz, 2003. download.berlios.de/martingale/MCBook-1.2.pdf.gz.
- [Pit03a] V. Piterbarg. A Stochastic Volatility Forward Libor Model with a Term Structure of Volatility Smiles. Technical report, Bank of America, 2003. ssrn.com/abstract=472061.



- [Pit03b] V. Piterbarg. Computing deltas of callable Libor exotics in forward Libor models. *Journal of Computational Finance*, 7(3), 2003. ssrn.com/abstract=396180.
- [Pit03c] V. Piterbarg. Mixture of Models: A Simple Recipe for a ... Hangover ? Technical report, Bank of America, 2003. ssrn.com/abstract=393060.
- [PTVF92] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes in C*. Cambridge University Press, 1992. www.library.cornell.edu/nr/cbookcpdf.html.
- [PvR05] R. Pietersz and M. van Regenmortel. Generic Market Models. Technical report, ABN AMRO, January 2005. ssrn.com/abstract=605301.
- [Rub83] M. Rubinstein. Displaced diffusion option pricing. *Journal of Finance*, 38:213–217, March 1983.
- [Sch89] M. Schroder. Computing the Constant Elasticity of Variance Option Pricing Formula. *Journal of Finance*, 44(1):211–219, March 1989. home.ust.hk/jinzhang/ust/Schroder_JF.pdf.
- [Sch02] E. Schögl. A multicurrency extension of the lognormal interest rate market models. *Finance and Stochastics*, 6:173–196, 2002.
- [Sob79] I. M. Sobol'. On the Systematic Search in a Hypercube. *Siam J. Numer. Anal.*, 16(5), October 1979.
- [Wu03] L. Wu. Fast at-the-money calibration of the Libor market model using Lagrange multipliers. *Journal of Computational Finance*, 6(2):39–72, Winter 2002/2003.