Cross Currency LIBOR Models: Backward vs. Forward Algorithm

(A Cross Currency Markov Functional Model)

Christian Fries

13.11.2004

(Version 1.0)

www.christian-fries.de/finmath

Interest Rates

Interest Rates (term structure)

Zero-Coupon Bond: Basic Objekt. Interest Rates are seen as derived quantities.

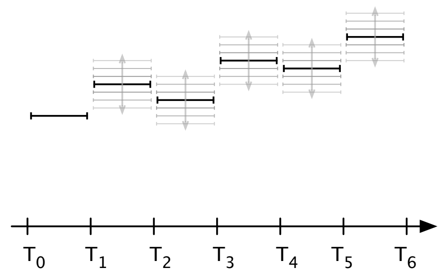
$$P(T_2) : [0, T_2] \times \Omega \mapsto \mathrm{I\!R} \qquad \text{(stochastic process)}$$

$P(T_2; t) : \Omega \mapsto \mathrm{I\!R}$ (random variable):
Value of a guaranteed payment of 1 (unit currency) at time T_2 as seen in $t < T_2$.

Remark: $P(T_2; T_2) = 1$

Forward-Rate (aka LIBOR):

$$\frac{P(T_1)}{P(T_2)} =: (1 + L(T_1, T_2) \cdot (T_2 - T_1)).$$



Interest Rate Curve (forward rates)

$L(T_1, T_2; t) : \Omega \mapsto \mathrm{I\!R}$ (random variable):
Interest rate of 1 invested in T_1 with payment in T_2 as seen in $t \leq T_1$.

Given $T_1 < T_2 < \ldots < T_n$ we write $L_i := L(T_i, T_{i+1})$ for the *stochastic process* $t \mapsto L(T_i, T_{i+1}; t)$.

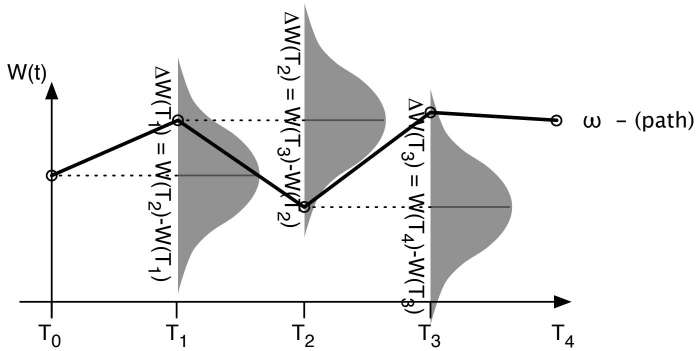
$\Rightarrow$ *term structure*: Family of stochastic processes: $L_1, L_2, L_3, \ldots, L_{n-1}$.

Modeling

Modeling (Itô stochastic process)

Underlying process is modeled as an Itô process:
E.g. lognormal processes for $L_i \rightarrow$ LIBOR Market Model:

$$\mathrm{d}L_i(t) = L_i(t)\mu(t)\mathrm{d}t + L_i(t)\sigma(t)\mathrm{d}W_i,$$



Time discretization
Given $0 = t_0 < t_1 < t_2 < t_3 < \ldots$ use

Euler Scheme:

$$L_i(t_{j+1}) = L_i(t_j) + \mu(t_i) \cdot L_i(t_j)\Delta t_i \; + \; \sigma(t_i) \cdot L_i(t_j) \cdot \underbrace{\Delta W_i}_{\sim \mathcal{N}(0,\sqrt{\Delta t_i})},$$

or

Euler Scheme of Log-Process:

$$L_i(t_{j+1}) = L_i(t_j) \cdot \exp\Big((\mu(t_i) - \frac{1}{2}\sigma(t_j)^2) \cdot \Delta t_i \; + \; \sigma(t_i) \cdot \underbrace{\Delta W_i}_{\sim \mathcal{N}(0,\sqrt{\Delta t_i})}\Big),$$

Pricing

Modeling (Itô stochastic process)

Example: European option value: European option value is a known function of the underlying process(es) at some future time T_{n+1}:

$$V(T_{n+1}) = \max\Big((L_n(T_n) - K) \cdot (T_{n+1} - T_n), 0\Big) \qquad \text{in } T_{n+1}.$$

Universal Pricing Theorem: Today's value $V(0)$ (the cost of replication) of a derivative is given by the expectation with respect to "some" measure $\mathbb{Q}^N$:

$$\frac{V(0)}{N(0)} = \mathrm{E}^{\mathbb{Q}^N}\left(\frac{V(T_n)}{N(T_n)}\right),$$

where N is some reference product (the so called Numéraie) and $\mathbb{Q}^N$ is such that for all financial products the above holds (this characterizes $\mathbb{Q}^N$)

For the calculation of the expectation $\mathrm{E}^{\mathbb{Q}^N}$ it is sufficient to know how the (drift of the) process looks like:

Definition (Martingal): A stochastic process M_t is called $\mathbb{P}$-Martingal, if

i) $M_s = \mathrm{E}^{\mathbb{P}}(M_t \mid \mathcal{F}_s) \quad \forall s \leq t$ ii) $\mathrm{E}^{\mathbb{P}}(|M_t|) < \infty \quad \forall t$

Martingal property for Itô processes: Let X denote an Itô stochastic process

$$\mathrm{d}X(t) = \mu(t)\mathrm{d}t + \sigma(t)\mathrm{d}W(t)$$

with $\mathrm{E}((\int_0^T \sigma_t^2 \mathrm{d}s)^{1/2}) < \infty$. Then

$$X \text{ is a Martingal} \;\Leftrightarrow\; X \text{ has zero drift, i.e. } \mu(t) \equiv 0.$$

Example: Calculating the Drift for the LIBOR Market Model

Brace, Gatarek, Musiela; Miltersen, Sandmann, Sondermann; etc.

LIBOR Market Model

Model:

$$\mathrm{d}L_i(t) = L_i(t)\mu_i(t)\mathrm{d}t \;+\; L_i(t)\sigma_i(t)\mathrm{d}W_i(t) \qquad \text{for } i = 0, \ldots, n-1,$$

Choose some reference product:

E.g. the T_n Bond: $N(t) = P(T_n; t)$ Note: $P(T_n; T_i) = \prod_{k=i}^{n-1} (1 + \delta_k L_k)^{-1}$.

Remember Universal Pricing Theorem / Martingale Property:

$$\frac{V(0)}{N(0)} = \mathrm{E}^{\mathbb{Q}^N}\left(\frac{V(T_i)}{N(T_i)}\right),$$

To do:

What does the process L_i look like under the probability measure $\mathbb{Q}^N$?
$\Leftrightarrow$ How does μ_i look like?

LIBOR Market Model: Derivation of the Drift under the Pricing Measure (1/3)

Choice of Numeraire: $N(t) = P(T_n;t)$ (T_n Bond).

Consider the N-relative Prices of *tradable assets*:

$$\frac{P(T_i)}{P(T_n)} = \prod_{k=i}^{n-1} \underbrace{\frac{P(T_k)}{P(T_{k+1})}}_{=1+\delta_k L_k} = \prod_{k=i}^{n-1} (1+\delta_k L_k) \qquad i = 1,\ldots,n-1.$$

In an arbitrage free model N-relative prices are drift free in:

$$\text{Drift}\left[\frac{P(T_i)}{P(T_n)}\right] = \text{Drift}\left[\prod_{k=i}^{n-1} (1+\delta_k L_k)\right] = 0 \qquad i = 1,\ldots,n-1 \text{ in } Q^{P(T_n)},$$

Product rule for Ito processes:

$$\begin{aligned}
&\mathrm{d}\left(\prod_{k=i}^{n-1} (1+\delta_k L_k)\right) \\
&= \prod_{k=i}^{n-1} (1+\delta_k L_k) \cdot \sum_{j=i}^{n-1} \left(\frac{\delta_j \mathrm{d}L_j}{(1+\delta_j L_j)} + \sum_{\substack{l \geq j+1 \\ l \leq n-1}} \frac{\delta_j \mathrm{d}L_j}{(1+\delta_j L_j)} \cdot \frac{\delta_l \mathrm{d}L_l}{(1+\delta_l L_l)} \right).
\end{aligned}$$

LIBOR Market Model: Derivation of the Drift under the Pricing Measure (2/3)

From

$$\underset{Q^{P(T_n)}}{\text{Drift}} \left[\prod_{k=i}^{n-1} (1 + \delta_k L_k) \right] = 0 \qquad i = 1, \ldots, n-1$$

we find

$$\sum_{j=i}^{n-1} \underset{Q^{P(T_n)}}{\text{Drift}} \left[\frac{\delta_j \mathrm{d} L_j}{(1 + \delta_j L_j)} + \sum_{\substack{l \geq j+1 \\ l \leq n-1}} \frac{\delta_j \mathrm{d} L_j}{(1 + \delta_j L_j)} \cdot \frac{\delta_l \mathrm{d} L_l}{(1 + \delta_l L_l)} \right] = 0 \qquad i = 1, \ldots, n-1$$

and thus

$$\underset{Q^{P(T_n)}}{\text{Drift}} \left[\frac{\delta_j \mathrm{d} L_j}{(1 + \delta_j L_j)} + \sum_{\substack{l \geq j+1 \\ l \leq n-1}} \frac{\delta_j \mathrm{d} L_j}{(1 + \delta_j L_j)} \cdot \frac{\delta_l \mathrm{d} L_l}{(1 + \delta_l L_l)} \right] = 0 \qquad j = 1, \ldots, n-1.$$

LIBOR Market Model: Derivation of the Drift under the Pricing Measure (3/3)

With

$$\mathrm{d}L_j = L_j \mu_j \mathrm{d}t + L_j \sigma_j \mathrm{d}W_j \quad , \quad \mathrm{d}L_j \cdot \mathrm{d}L_l = L_j L_l \sigma_j(t) \sigma_l(t) \rho_{j,l} \mathrm{d}t$$

the drift equation

$$\underset{\mathbb{Q}^{P(T_n)}}{\mathrm{Drift}} \left[\frac{\delta_j \mathrm{d}L_j}{(1+\delta_j L_j)} + \sum_{\substack{l \geq j+1 \\ l \leq n-1}} \frac{\delta_j \mathrm{d}L_j}{(1+\delta_j L_j)} \cdot \frac{\delta_l \mathrm{d}L_l}{(1+\delta_l L_l)} \right] = 0 \qquad j = 1, \ldots, n-1$$

becomes

$$\mu_j \frac{\delta_j L_j}{(1+\delta_j L_j)} + \sum_{\substack{l \geq j+1 \\ l \leq n-1}} \frac{\delta_j L_j}{(1+\delta_j L_j)} \cdot \frac{\delta_l L_l}{(1+\delta_l L_l)} \cdot \sigma_j(t) \sigma_l(t) \rho_{j,l} = 0.$$

Thus, under the *pricing measure* $\mathbb{Q}^{P(T_n)}$:

$$\mu_j = - \sum_{\substack{l \geq j+1 \\ l \leq n-1}} \frac{\delta_l L_l}{(1+\delta_l L_l)} \cdot \sigma_j(t) \sigma_l(t) \rho_{j,l}$$

Conclusion:

$$\mathrm{d}L_j = \sum_{j+1 \leq l \leq N-1} \frac{-\delta_l L_j L_l}{(1+\delta_l L_l)} \cdot \sigma_j(t) \sigma_l(t) \rho_{j,l} \, \mathrm{d}t \; + \; L_j \sigma_j(t) \, \mathrm{d}W_j \quad ; \qquad \text{where } < \mathrm{d}W_i, \mathrm{d}W_j > = \rho_{i,j} \mathrm{d}t$$

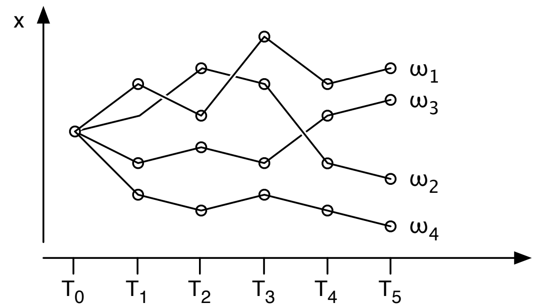
Free parameters (→ have to be fitted to market option prices):

$$\sigma_1(t), \ldots, \sigma_{N-1}(t) \quad \text{(Volatility)} \quad , \quad \rho_{i,j} \quad \text{(Correlation)}$$

Discretization & Implementation (I)

Monte-Carlo / Forward Algorithm

Monte-Carlo / Forward Algorithm



Universal Pricing Theorem:

$$\frac{V(0)}{N(0)} = \mathrm{E}^{\mathbb{Q}^N}\left(\frac{V(T_n)}{N(T_n)}\right)$$

Assume V consists of finite number of payments

X_i paid in T_i, X_i is $\mathcal{F}_{T_i}$-measurable random variable, $i = 1, \ldots, n$.

Value

Value of payment: $\mathrm{E}^{\mathbb{Q}^N}\left(\frac{X_i}{N(T_i)}\right) \quad \Rightarrow \quad \mathrm{E}^{\mathbb{Q}^N}\left(\frac{V}{N}\right) = \sum_{i=1}^{n} \mathrm{E}^{\mathbb{Q}^N}\left(\frac{X_i}{N(T_i)}\right)$

Monte-Carlo Simulation:

Sample space $\tilde{\Omega} = \{\omega_1, \ldots, \omega_m\}$, e.g. $m \approx 10000$

$$\mathrm{E}^{\mathbb{Q}^N}\left(\frac{X_i}{N(T_i)}\right) \approx \frac{1}{m} \sum_{\omega \in \tilde{\Omega}} \frac{X_i(\omega)}{N(T_i;\omega)}$$

$$\Rightarrow \qquad V(0) = N(0) \cdot \mathrm{E}^{\mathbb{Q}^N}\left(\frac{V}{N}\right) \approx N(0) \cdot \frac{1}{m} \sum_{\omega \in \tilde{\Omega}} \sum_{i=1}^{n} \frac{X_i(\omega)}{N(T_i;\omega)}$$

Example Product: Bermudan Option

Bermudan Option

Bermudan Option: Given multiple exercise dates $T_1 < T_2 < T_3 < \ldots < T_n$ at each time T_i holder has the choice between

[exercise] - choose the value $U(T_i)$ of some underlying financial product

[hold] - choose to exercise later, ie. an Bermudan Option with exercise dates $\{T_{i+1}, \ldots, T_n\}$.

$\rightarrow$ Option on option ... on option.

Value of Bermudan Option according to optimal exercise

$$V_{\{T_i,\ldots,T_n\}}(T_i) = \max\{U(T_i)\ ,\ V_{\{T_{i+1},\ldots,T_n\}}(T_i)\},$$

where

$$V_{\{T_{i+1},\ldots,T_n\}}(T_i) = N(T_i) \cdot \mathrm{E}^{\mathbb{Q}^N}\left(\frac{V_{\{T_{i+1},\ldots,T_n\}}(T_{i+1})}{N(T_{i+1})} \;\middle|\; \mathcal{F}_{T_i} \right)$$

Bermudan Option: Given multiple exercise dates $T_1 < T_2 < T_3 < \ldots < T_n$ at each time T_i holder has the choice between

[exercise] - choose the value $U(T_i)$ of some underlying financial product

[hold] - choose to exercise later, ie. an Bermudan Option with exercise dates $\{T_{i+1}, \ldots, T_n\}$.

$\rightarrow$ Option on option ... on option.

Value of Bermudan Option according to optimal exercise

$$V_{\{T_i,\ldots,T_n\}}(T_i) = \max\{U(T_i)\ ,\ \boxed{V_{\{T_{i+1},\ldots,T_n\}}(T_i)\},}$$

where

$$\boxed{V_{\{T_{i+1},\ldots,T_n\}}(T_i) = N(T_i) \cdot \mathrm{E}^{\mathbb{Q}^N}\left(\frac{V_{\{T_{i+1},\ldots,T_n\}}(T_{i+1})}{N(T_{i+1})} \;\middle|\; \mathcal{F}_{T_i} \right)}$$

Requires calculation of conditional expectation at some future time

Example Product:

Bermudan Power Reverse Dual Currency

Underyling:

$$\max(\ \min(\ \frac{FX(t)}{FX(0)} \cdot A\% - B\%\ ), \text{cap}\ ), \text{floor}) \quad - \quad L(t)$$

where $A\%$, $B\%$, cap, floor are constants.

Product: A Bermudan Option on the above.

Requires:

- Cross Currency Model
  - Depends on FX rate
  - Depends on Interest Rates
- Estimator for Conditional Expectation
  - Bermudan Option

Cross Currency LIBOR Market Model

Cross Currency LIBOR Market Model

Model Framework:

Dynamic under $\mathbb{Q}^N$ (with Numéraire (reference product) $N(T_i) := \prod_{j=0}^{i-1}(1 + L_j(T_j) \cdot \delta_j)$:

$$\begin{aligned} \mathrm{d}L_j(t) &= \quad L_j(t) \cdot \mu_j(t)\,\mathrm{d}t + \ L_j(t) \cdot \sigma_j(t)\,\mathrm{d}W_j \qquad j = 1,\ldots,N \\ \mathrm{d}FX(t) &= FX(t) \cdot \mu^{FX}(t)\,\mathrm{d}t + FX \cdot \sigma^{FX}(t)\,\mathrm{d}W^{FX} \\ \mathrm{d}\tilde{L}_j(t) &= \quad \tilde{L}_j(t) \cdot \tilde{\mu}_j(t)\,\mathrm{d}t + \ \tilde{L}_j(t) \cdot \tilde{\sigma}_j(t)\,\mathrm{d}\tilde{W}_j \qquad j = 1,\ldots,N \end{aligned}$$

where ($\beta(t) := \max\{i \mid T_i \leq t\}$)

$$\mu_j(t) = \sum_{\beta(t)\leq l\leq j} \frac{L_l\delta_l}{(1+L_l\delta_l)} \cdot \sigma_j(t)\sigma_l(t)\rho_{j,l} \qquad \int_{T_i}^{T_{i+1}} \mu^{FX}(t)\mathrm{d}t = \log\left(\frac{1+L_i(T_i)\delta_i}{1+\tilde{L}_i(T_i)\delta_i}\right)$$

$$\tilde{\mu_j}(t) = \sum_{\beta(t)\leq l\leq j} \frac{\tilde{L}_l\delta_l}{(1+\tilde{L}_l\delta_l)} \cdot \tilde{\sigma}_j(t)\tilde{\sigma}_l(t)\rho_{j,l} - \tilde{\sigma}_j(t)\sigma^{FX}(t)\rho_{\tilde{j},FX}(t)$$

Free parameters:

- Rich volatility structure possible: $\sigma_1(t),\ldots,\sigma_{N-1}(t)$, $\sigma^{FX}(t)$, $\tilde{\sigma}1(t),\ldots,\tilde{\sigma}_{N-1}(t)$
- Rich correlation structure possible:

$\rho_{i,j}(t)\mathrm{d}t =< \mathrm{d}W_i, \mathrm{d}W_j >,$	$\rho_{\tilde{i},\tilde{j}}(t)\mathrm{d}t =< \mathrm{d}W_{\tilde{i}}, \mathrm{d}W_{\tilde{j}} >,$	inter currency
$\rho_{i,FX}(t)\mathrm{d}t =< \mathrm{d}W_i, \mathrm{d}W^{FX} >,$	$\rho_{\tilde{i},FX}(t)\mathrm{d}t =< \mathrm{d}W_{\tilde{i}}, \mathrm{d}W^{FX} >$	currency to FX
$\rho_{i,\tilde{j}}(t)\mathrm{d}t =< \mathrm{d}W_i, \mathrm{d}W_{\tilde{j}} >,$		cross currency

To do: Calibrate the model, the choose free model parameters such that given/known option prices are reproduced
$\rightarrow$ inverse problem (not always trivial)

• Characteristics

- Rich correlation and volatility structure:
  best for correlation sensitive products

- Markovian only in high dimensions:
  Monte-Carlo Simulation seems natural. Lattice Implementation not trivial.

- Needs extensions to calibrate smile-surface (ie. to calibrate to more than one interest rate option per maturity)

Discretization & Implementation (2)

Lattice / Backward Algorithm

Lattice / Backward Algorithm

Universal Pricing Theorem:

$$\frac{V(T_i)}{N(T_i)} = \mathsf{E}^{\mathbb{Q}^N}\left(\frac{V(T_{i+1})}{N(T_{i+1})} \mid \mathcal{F}_{T_i}\right) \qquad \Rightarrow \qquad V(T_i) = \mathsf{E}^{\mathbb{Q}^N}\left(V(T_{i+1}) \cdot \frac{N(T_i)}{N(T_{i+1})} \mid \mathcal{F}_{T_i}\right)$$

Assume V (and $\frac{N(T_i)}{N(T_{i+1})}$) solely depend on some underlying process L (e.g. an interest rate), i.e.

$$V(T_i) = V(T_i, L(T_i))$$

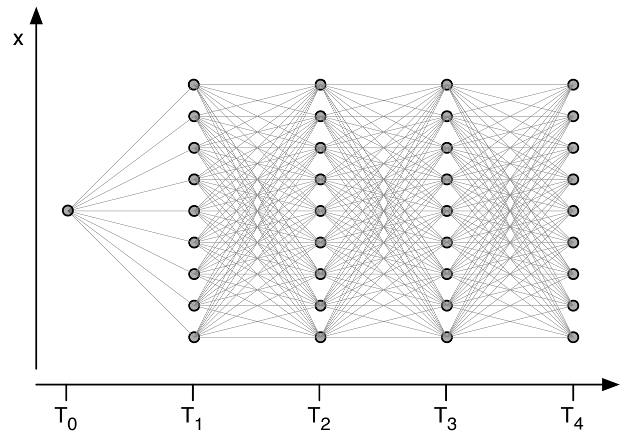
Expectation conditioned to state variable L

$$V(T_i, L^*) = \mathsf{E}^{\mathbb{Q}^N}\left(V(T_{i+1}, L(T_{i+1})) \cdot \frac{N(T_i)}{N(T_{i+1})} \mid L(T_i) = L^*\right)$$

Lattice Method:
Sample state space

$$\mathsf{E}^{\mathbb{Q}^N}\left(V(T_{i+1}, L(T_{i+1})) \cdot \frac{N(T_i)}{N(T_{i+1})} \mid L(T_i) = L^*\right)$$
$$\approx \sum_k V(T_{i+1}, L_k) \cdot \frac{N(T_i)}{N(T_{i+1})} \cdot q_{i,k}$$

where $q_{i,k}$ denotes the (discrete state) transition probability $L^* \to L_k$ from T_i to T_{i+1}.



Cross Currency Markov Functional Model

Lattice / Backward Algorithm

Paper available on the web:
Fries, Christian P.; Rott, Marius G.:
“Cross Currency and Hybrid Markov Functional Models (2004)”

Cross Currency Markov Functional Model

Markov Functional Modeling:
The financial quantities are functions of *some* underyling markov process.

Given driving Markov processes:

$$\begin{aligned} \mathrm{d}x &= \sigma_x(t)\mathrm{d}W_1 \\ \mathrm{d}y &= \mu(t,x,y,z)\mathrm{d}t + \sigma_y(t)\mathrm{d}W_2 \\ \mathrm{d}z &= \sigma_z(t)\mathrm{d}W_3 \end{aligned}$$

Postulate

- The (*domestic*) LIBOR $L(T_k) = L_k(T_k)$ (seen upon its maturity) is a (deterministic) function of $x(T_k)$: $L_k(T_k) = L(T_k, x(T_k))$.
- The FX rate $FX(T_k)$ is a deterministic function of $y(T_k)$ only.
- The (*foreign*) LIBOR $\tilde{L}(T_k) = \tilde{L}_k(T_k)$ (seen upon its maturity) is a (deterministic) function of $z(T_k)$.

$\Rightarrow$ This *forces* a cross-currency LIBOR model onto a computational feasible (3D) lattice (backward algorithm possible).

Markov Functional Model

Markov Functional Modeling:

The financial quantities are functions of *some* underyling markov process.

1-D (single currency) LIBOR Markov functional model:

Driving Process	**LIBOR functional**
$\mathrm{d}x = \sigma_x(t)\mathrm{d}W_1$	$L_k(T_k) = L(T_k, x(T_k))$

Where is the pricing measure? What is the Numéraire?

The approach is reversed here: We choose $N(t) := P(T_n; t)$ – the T_n-Bond – as Numéraire.

Then: We postulate that the driving processes are given under the pricing measure $\mathbb{Q}^N$ and define the Numéraire process

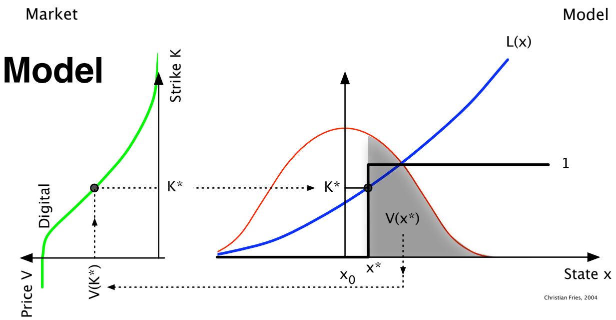
$$\frac{P(T_{i+1}; T_i)}{N(T_i)} := \mathrm{E}^{\mathbb{Q}^N}\left(\frac{1}{N(T_{i+1})} \mid \mathcal{F}_{T_i}\right) \quad \Rightarrow \quad N(T_i) := \frac{1}{(1 + L(T_k)\Delta T_i) \cdot \mathrm{E}^{\mathbb{Q}^N}\left(\frac{1}{N(T_{i+1})} \mid \mathcal{F}_{T_i}\right)}$$

$\rightarrow$ The numéraire is defined (via backward induction) as a functional of x too.

The Universal Pricing Theorem holds *per definition*:

$$\frac{V(T_i)}{N(T_i)} = \mathrm{E}^{\mathbb{Q}^N}\left(\frac{V(T_{i+1})}{N(T_{i+1})} \mid \mathcal{F}_{T_i}\right)$$

Cross Currency Markov Functional Model: Calibration of (domestic) LIBOR Fct’al



Calibration of the (domestic) LIBOR Markov Functional Model

Degrees of Freedom:

- The functional $\xi \mapsto L_i(\xi)$.
- The volatility $\sigma(t)$ of the driving process $x(t)$

Calibration (e.g.) means: The model should reproduce *all* European Options on L_i, i.e. all Product V with

$$V(K;T_{n+1}) = \max\Big((L_n(T_n) - K) \cdot (T_{n+1} - T_n), 0\Big) \qquad \text{in } T_{n+1}.$$

Calibration to continuous family of European Options on L_i (Caplet smile) can be obtained (almost) explicitly from market prices:

Lemma (Breeden & Litzenberger, etc.): For the probability distribution ϕ_{L_i} of L under the measure $\mathbb{Q}^N$, $N = P(T_{n+1})$ we have

$$\underbrace{\int_0^K \phi_{L_i}(\kappa)\mathrm{d}\kappa}_{\text{model distribution}} = \underbrace{N(0) \cdot \frac{\partial}{\partial K} V(K;T_{n+1})}_{\text{market price}}$$

$\Rightarrow$ Determination of the functional $\xi \mapsto L_i(\xi)$ can be done through a simple inversion (numerically feasable 1D root finding).

Calibration of the FX Functional

Degrees of Freedom:

- The functional $\eta \mapsto FX(\eta)$.
- The volatility $\sigma_y(t)$ of the driving process $y(t)$ – $\mathrm{d}y = \mu(t, x, y, z)\mathrm{d}t + \sigma_y(t)\mathrm{d}W_2$

Calibration (e.g.) means: The model should reproduce *all* European FX Options, i.e. all Product V with

$$V(K; T_n) = \max\left((FX(T_n) - K), 0\right).$$

1. Drift of the underlying process y

Example: Choose the functional as $FX(\eta) = \exp(a \cdot \eta)$ $\rightarrow$ log-normal model for the FX.

Then the drift $\mu(t, x, y, z)$ has to fullfill

$$\mu(T_i, \xi, \eta)\Delta T_i = \frac{1}{a} \log\left(\frac{1 + L_i(\xi)\Delta T_i}{1 + \tilde{L}_i(\eta)\Delta T_i}\right) - \frac{a \cdot \sigma_y(T_i)}{2}\Delta T_i$$

(Note: Same drift adjustment as for the cross currency LIBOR Market Model).

Conclusion

- Cross Currency MF Model has: Efficient Calibration (fitting the model to the market)
    - Calibrate to a one parameter family of domestic interest rate option per maturity (e.g. full caplet smile calibration)
    - Calibrate to a one parameter family of foreign interest rate option per maturity (e.g. full caplet smile calibration)
    - Calibrate to "some" (a least one ;-) FX options per maturity
- Cross Currency MF Model needs:
    - A fast algorithm for conditional expectation

$$\mathrm{E}^{\mathbb{Q}^N}\left(f\Big(x(T_{i+1}), y(T_{i+1}), z(T_{i+1})\Big) \;\Big|\; (x(T_i), y(T_i), z(T_i)) = (x^*, y^*, z^*)\right)$$

Consider: Tree, PDE, Full Numerical Integration, Fourier or Wavelet Methods

