Fixed Income Analysis
Calibration in lattice models — Part II

Calibration to the initial volatility structure
Pitfalls in volatility calibrations
Mean-reverting log-normal models (Black-Karasinski)
Brownian-path independent (BPI) models
Trinomial lattice models (Hull-White)

Jesper Lund
April 28, 1998

The volatility structure

- **Definition**: volatility of zero-coupon yields as a function of time to maturity.
- In a one-factor model, the zero-coupon rate is governed by
  \[ dR(t,T) = \mu_R(t,T)dt + \sigma_R(t,T)R(t,T)dW_t. \]  
  \[ (1) \]
- Here, \( \sigma_R(t,T) \) is the **proportional** volatility, and \( \sigma_R(t,T)R(t,T) \) is the **basis-point** volatility.
- From Ito's lemma, the volatility structure is given by
  \[ \sigma_R(t,T)R(t,T) = \sigma(r_t) \frac{\partial R(t,T)}{\partial r} = -\frac{\sigma(r_t)}{(T-t)P(t,T)} \frac{\partial P(t,T)}{\partial r} \]  
  \[ (2) \]
- Thus, the volatility structure depends on the effect of the short rate, \( r_t \), on bond prices, \( P(t,T) \).
- Mainly determined by the speed of **mean reversion**.
Calibration in the BDT model

- The BDT model is an approximation to the SDE

\[ d \log r_t = \left\{ b(t) + \frac{\sigma'(t)}{\sigma(t)} \log r_t \right\} dt + \sigma(t)dW_t^Q. \]  

(3)

- Note that the mean reversion coefficient is tied to \( \sigma(t) \), the short-rate volatility at time \( t \).

- Calibrating the BDT model to the initial yield and volatility curve:
  - The basic geometry of the the BDT binomial tree is unchanged.
  - The pair \( \{b(n\Delta), \sigma(n\Delta)\} \) is chosen to match the yield and volatility of the \((n + 1)\)-period bond.
  - Alternative parameterization: \( \{r(n,0), \delta_n\} \), where \( r(n,0) \) is the bottom node and \( \log \delta_n \) is the spacing for \( \log r(n,s) \).
  - We have two equations in two unknowns, but they are easy to solve numerically (Newton-Raphson) when the forward-induction technique is used.

Pitfalls in volatility calibration – 1

- The basic problem can be illustrated for the extended Vasicek model (in continuous time)

\[ dr_t = \kappa(t) \{\theta(t) - r_t\} dt + \sigma(t)dW_t^Q. \]  

(4)

- The pair \( \{\theta(t), \kappa(t)\} \) is chosen to fit the initial (time \( t = 0 \)) yield and volatility curves.

- The initial forward-rate volatility structure is given by:

\[ \sigma(0,T) = \sigma(0)e^{-\int_0^T \kappa(u)du}. \]  

(5)

- Basis-point volatility structure for zero-coupon rates:

\[ \sigma_R(0,T) = \frac{1}{T} \int_0^T \sigma(0,s)ds = \sigma(0) \int_0^T e^{-\int_0^s \kappa(u)du} ds. \]  

(6)

- Note that the shape only depends on \( \kappa(t) \), for \( 0 \leq t \leq T \).
Pitfalls in volatility calibration – 2

- At time $t$, we have a new volatility structure, and the payoffs from derivatives (e.g., call options) depend on the time $t$ volatilities.

- **Caveat**: in a Markovian model, the new volatility structure is completely determined from the initial volatility structure.

- New forward-rate volatilities:

  $$
  \sigma(t, T) = \sigma(t) e^{-\int_t^T \kappa(u) du} = \sigma(t) \frac{\sigma(0, T)}{\sigma(0, t)}.
  $$

  (7)

- New volatility structure for zero-coupon rates:

  $$
  \sigma_R(t, T) = \frac{\sigma(t)}{T - t} \cdot \frac{T \sigma_R(0, T) - t \sigma_R(0, t)}{\sigma(0, t)}
  $$

  (8)

- Apart from $\sigma(t)$, which is common for all maturities, (7) and (8) only depend on the initial volatility structure.

---

Pitfalls in volatility calibration – 3

- Constraining the evolution of the volatility structure in this way could have undesirable effects on derivatives prices.

- If the current volatility curve is humped, the future curve will be steeply downward sloping, although flat eventually.

- Hull and White’s recommendation: do not calibrate the model to the volatility structure (if anything, use cap prices instead).

- Additional problems for the BDT model:
  - Minor problem: no analytical solution for bond prices, so the dependencies are more difficult to analyze (and understand).
  - Major problem: mean reversion is tied to the future short-rate volatilities since $\kappa(t) = -\sigma'(t)/\sigma(t)$ in the BDT model.

- If the volatility structure is downward sloping (normal situation), we need BDT’s $\sigma(t)$ to be decreasing in $t$ — and this is an unrealistic property (in general).
Pitfalls in volatility calibration – 4

- Illustration of the BDT volatility problem, taken from a paper by Simon Schultz and Per Søgaard-Andersen, “Pricing Caps and Floors,” Finans/Invest, 4/93 [in Danish].
- Market prices (bid-ask) and three model prices for interest-rate caps (all prices are relative to bid-ask midpoint).
- The BDT and Hull-White (extended) Vasicek models are calibrated to a downward-sloping volatility structure.
- Since \( \sigma(t) \) in the BDT model is decreasing over time, the BDT cap prices generally are too low in the table below.

<table>
<thead>
<tr>
<th>Maturity</th>
<th>Bid</th>
<th>Ask</th>
<th>BDT</th>
<th>Vasicek</th>
<th>Black-76</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>0.95</td>
<td>1.05</td>
<td>0.94</td>
<td>1.00</td>
<td>0.97</td>
</tr>
<tr>
<td>3</td>
<td>0.97</td>
<td>1.03</td>
<td>0.90</td>
<td>1.00</td>
<td>0.98</td>
</tr>
<tr>
<td>4</td>
<td>0.97</td>
<td>1.03</td>
<td>0.92</td>
<td>1.00</td>
<td>0.98</td>
</tr>
<tr>
<td>5</td>
<td>0.97</td>
<td>1.03</td>
<td>0.94</td>
<td>0.99</td>
<td>0.97</td>
</tr>
</tbody>
</table>

Mean reversion in log-normal models

- Black and Karasinski (1991) relax the BDT restriction on the mean reversion coefficient,
  \[
  d \log r_t = \{ b(t) - \phi(t) \log r_t \} dt + \sigma(t) dW_t^Q. \tag{9}
  \]

- The model (9) cannot be implemented in a recombining binomial tree with constant time steps and probabilities \( \theta(n,s) = 0.5 \).
- There are three possible modifications of the tree which allow for (arbitrary) mean reversion:
  1. Non-constant time steps, \( \Delta_n \), with \( \theta(n,s) = 0.5 \). This is suggested by Black and Karasinski (1991). The main disadvantage is that the spacing declines over time, and we generally want the opposite (if anything).
  2. Non-constant probabilities of an up-move, but with constant time steps. We match the expected change in \( r \) (mean reversion) at node \( (n,s) \) by adjusting \( \theta(n,s) \). The disadvantage is slower convergence.
  3. Trinomial trees (Hull-White) which have three branches at each node.
Brownian-path independence (BPI)

- Why is it possible to construct a simple binomial tree — with constant time steps and $\theta(n,s) = 0.5$ — in the BDT case?
- The solution to the BDT SDE (3) can be written as
  \[
  \log r_t = \frac{\sigma(t)}{\sigma(0)} \log r_0 + \sigma(t) \int_0^t \frac{b(s)}{\sigma(s)} ds + \sigma(t) W_t^Q
  \equiv B(t) + \sigma(t) W_t^Q
  \]

  \[ (10) \]
- The BDT model modifies the Brownian motion tree only by scaling [through $\log(\delta_n) = c \cdot \sigma(t)$] and the bottom node, $r(n,0)$.
- In most BPI models, the short rate has the general form:
  \[ r_t = F \left( B(t) + \sigma(t) W_t^Q \right) , \]
  \[ (11) \]
  for some function $F(x)$. Note that BDT has $F(x) = \exp(x)$.
- Simple binomial trees requires a BPI model [Jamshidian (1991)].

Trinomial lattices – 1

- Introduced by Hull and White (1993, 1994).
- Extended Vasicek model with $\kappa(t) = \kappa$ and $\sigma(t) = \sigma$,
  \[
  dr_t = \kappa \{ \theta(t) - r_t \} dt + \sigma dW_t^Q.
  \]
  \[ (12) \]
- We rewrite (12) as
  \[
  r_t = \alpha(t) + x_t
  \]
  \[
  \alpha(t) = e^{-\kappa t} r_0 + \int_0^t e^{-\kappa(t-s)} \kappa \theta(s) ds
  \]
  \[
  dx_t = -\kappa x_t dt + \sigma dW_t^Q, \quad \text{with } x_0 = 0.
  \]
  \[ (13) \]
- First step: build a trinomial tree for $x_t$. This tree is symmetric around $x = 0$, and the geometry depends only on $\kappa$ and $\sigma$.
- Second step: calibrate the time-dependent parameters, $\alpha_i = \alpha(t_i)$, to match the initial term structure.
Trinomial lattices – 2

• Three-period trinomial tree:

```
x(0,0) ← x(1,0) ← x(1,1) ← x(2,0) ← x(2,1) ← x(2,2) ← x(3,0) ← x(3,1) ← x(3,2)
```

• **Comment 1:** the numbering scheme, \(x(i,j)\), is different from binomial case. The center node has \(j = 0\) for all times \(i\).

• **Comment 2:** there is a different branching scheme for low and high values of \(j\) (node number). The purpose is accommodating mean reversion, while retaining positive probabilities.

Trinomial lattices – 3

• First and second moments of the SDE for \(x_t\):

\[
E_t \left[ x_{t+\Delta} - x_t \right] = Mx_t = \left( e^{-\kappa \Delta} - 1 \right) x_t \approx -\kappa \Delta x_t \tag{16}
\]

\[
\text{Var}_t \left[ x_{t+\Delta} - x_t \right] \equiv V = \frac{\sigma^2}{2\kappa} \left( 1 - e^{-2\kappa \Delta} \right) \approx \sigma^2 \Delta \tag{17}
\]

• The two approximations follow from \(\exp(z) \approx 1 + z\).

• The trinomial model has a constant time step, denoted \(\Delta\).

• The spacing on the \(x\)-axis is specified as \(\Delta x = \sqrt{3V}\).

• This means that \(x(i,j) = j \Delta x\), for \(-n_i \leq j \leq n_i\).

• From each node, \((i,j)\), there are branches to three nodes with probabilities: \(p_u\) (top node), \(p_m\) (mid mode), and \(p_d\) (low node).
Trinomial lattices - 4

- We look at a “normal” node \((i, j)\) — not special branching.
- The probabilities \(p_u, p_m\) and \(p_d\) are chosen in order to satisfy
  \[
  p_u \Delta x - p_d \Delta = Mj \Delta x \\
  p_u(\Delta x)^2 + p_d(\Delta x)^2 = V + M^2 j^2(\Delta x)^2 \\
  p_u + p_m + p_d = 1
  \]
  (18)  (19)  (20)
- That is, we match the moments of \((x_{t+\Delta} - x_t)\), see (16) and (17).
- Since \(V = (\Delta x)^2/3\), the solution is easily found as
  \[
  p_u = \frac{1}{6} + \frac{j^2 M^2 + j M}{2} \\
  p_m = \frac{2}{3} - \frac{j^2 M^2}{2} \\
  p_d = \frac{1}{6} + \frac{j^2 M^2 - j M}{2}
  \]
  (21)  (22)  (23)

Trinomial lattices - 5

- Note that the probabilities are independent of the initial term structure. Apart from \(j\), they only depend on \(\kappa\) and \(\Delta\).
- When branching out from the special top and bottom nodes, similar formulas apply — see Hull (1997) or Hull & White (1994).
- This completes the first step, setting up the nodes of the trinomial lattice.
- Second step: let \(r(i, j) = \alpha_i + x(i, j)\), and calibrate \(\alpha_i\) recursively so that the \((i + 1)\)-period bond price is matched exactly.
- As in the BDT model, the calibration is done with forward induction and Arrow-Debreu prices.
- We start with \(\alpha_0 = r(0, 0) = -\log P(1)/\Delta\).
- Define \(p(i, j) = \exp[-r(i, j)\Delta] = \exp[-\alpha_i \Delta] \cdot \exp[-j(\Delta x)\Delta]\).
Trinomial lattices – 6

- Assume that we have computed $\alpha_{m-1}$, where $m \geq 1$.
- First, we use the forward equation to compute $G(m, j)$,

$$G(m, j) = \sum_{k=-(n_m-1)}^{n_m-1} q(k, j)p(m-1, k)G(m-1, k), \quad (24)$$

where $q(k, j)$ is the probability of moving from the node $(m-1, k)$ to $(m, j)$. **Note:** $q(k, j)$ is only non-zero for at most three $k$.

- Second, with $G(m, j)$ at hand, the $(m + 1)$-period bond price is given by

$$P(m + 1) = \sum_{j=-n_m}^{n_m} G(m, j)p(m, j)$$

$$= e^{-\alpha_m \Delta} \sum_{j=-n_m}^{n_m} G(m, j)e^{-j(\Delta x)\Delta} \quad (25)$$


Trinomial lattices – 7

- The solution to (25) is readily available in closed form:

$$\alpha_m = \frac{\log \left( \sum_{j=-n_m}^{n_m} G(m, j)e^{-j(\Delta x)\Delta} \right) - \log P(m + 1)}{\Delta} \quad (26)$$

- Having found $\alpha_m$, we proceed to $m + 1$ (next period) using the same recursions — forward equation (24) followed by (26).

- This completes the construction of the Hull-White trinomial tree for the extended Vasicek model.

- The parameters $\kappa$ and $\sigma$ can be calibrated to, e.g., cap prices by minimizing the squared pricing errors

$$S(\kappa, \sigma) = \sum_i \left(V^\text{actual}_i - V^\text{model}_i\right)^2. \quad (27)$$