

# Hull and White Two-factor model

Ismail Laachir

February 18, 2016

## Contents

<b>1 Hull and White two-factor model</b>	<b>1</b>
<b>2 Trinomial Tree method</b>	<b>2</b>
2.1 Tree for a one-factor process : . . . . .	2
2.2 Tree for a two-factor process : . . . . .	3
2.3 Calibration of the tree to the market yield curve : . . . . .	5
<b>3 Pricing of a security using the tree :</b>	<b>5</b>

## Premia 18

### 1 Hull and White two-factor model

Hull and White model is a short-rate model. One of its main characteristics is its ability to match the initial yield curve by using time-varying parameter. A one factor version of this model was first proposed in [1] (already implemented in Premia). In this project we consider the two-factor version proposed in [2].

Hull&White two-factor model is defined by an EDS which describes the evolution of the spot rate  $r(t)$ :

$$\begin{cases} dr(t) = [\theta(t) + u(t) - a r(t)] dt + \sigma_1 dW_1(t) \\ du(t) = -b u(t) dt + \sigma_2 dW_2(t), \quad u(0) = 0 \end{cases}$$

The two processes  $W_1$  and  $W_2$  are brownian motions with instantaneous correlation  $\rho$ , and  $\theta$  is a deterministic function totally given by the market value of the zero coupon bonds.

Let us denote by  $P_M(0, T)$  the market zero coupon bond value maturing at time  $T$  and  $f_M(t) = -\frac{\partial \log(P_M(0, t))}{\partial t}$  the market present instantaneous forward rate, then with an appropriate choice for the function  $\theta$  (see Hull&White 1994 for details), the model exactly fits the market bonds curve and we have several analytical formulas :

Zero coupon bond at time  $t$  knowing that  $r(t) = r_t$  and  $u(t) = u_t$  :

$$P(t, T) = A(t, T)e^{-B(t, T)r_t - C(t, T)u_t}.$$

Explicite formulations for  $A$ ,  $B$  and  $C$  can be found in [2].

The price at time  $t$  for a European Call on a ZC bond :

$$C_t = \mathbb{E}_t \left[ e^{-\int_t^T r(s)ds} (P(T, S) - K)_+ \right] = P(t, S)\mathcal{N}(h) - KP(t, T)\mathcal{N}(h - \sigma_p).$$

Where  $\mathcal{N}$  is the cumulative function of the normal law,

$$h = \frac{1}{\sigma_p} \log \left( \frac{P(t, S)}{P(t, T)K} \right) + \frac{\sigma_p}{2}$$

and  $\sigma_p$  is given in [2].

This closed formula for european option on bond also leads to closed formula for cap and floor.

## 2 Trinomial Tree method

### 2.1 Tree for a one-factor process :

We first recall the procedure for the construction of a trinomial tree that approximates a process  $x$  of the form :

$$dx(t) = -ax(t)dt + \sigma dW(t), \quad x(0) = 0$$

Let  $0 = t_0 < t_1 < \dots < t_n = T$  be a time scale for our tree in  $[0, T]$ ,  $\Delta t_i = t_{i+1} - t_i$ , and  $x_{i,j}$  the  $x$  node value at time  $t_i$  for the  $j^{th}$  space step of the tree (starting from the down). We need then :

$$\begin{cases} \mathbb{E}[x(t_{i+1})|x(t_i) = x_{i,j}] = M_{i,j} \\ \mathbb{V}ar[x(t_{i+1})|x(t_i) = x_{i,j}] = V_{i,j}^2 = V_i^2 \end{cases}$$

Knowing that  $x$  has is a gaussian process,  $M_{i,j}$  and  $V_i$  can be computed :

$$\begin{cases} M_{i,j} = x_{i,j} e^{-a \Delta t_i} \\ V_i^2 = \frac{\sigma^2}{2a} [1 - e^{-a \Delta t_i}] \end{cases}$$

At time  $t_i$ , the nodes are equally spaced, so :  $x_{i,j} = j \Delta x_i$ , with  $\Delta x_i = V_{i-1} \sqrt{3}$ .

Starting at time  $t_i$  from node  $x_{i,j}$ , the process can move to three node at time  $t_{i+1}$  :

$$\begin{cases} \text{Up with probability } p_u(i, j) \text{ to the node } x_{i+1,k+1} \\ \text{Middle with probability } p_m(i, j) \text{ to the node } x_{i+1,k} \\ \text{Down with probability } p_d(i, j) \text{ to the node } x_{i+1,k-1} \end{cases}$$

The index  $k$  is chosen so that  $x_{i+1,k}$  is as close as possible to the mean  $M_{i,j}$ , ie :

$$k = \text{round} \left( \frac{M_{i,j}}{\Delta x_{i+1}} \right) = \text{round} (j \beta_i)$$

$$, \text{ with } \beta_i = \frac{\Delta x_i}{\Delta x_{i+1}} e^{-a \Delta t_i}.$$

The probabilities  $p_u(i, j)$ ,  $p_m(i, j)$  and  $p_d(i, j)$  are chosen so that that conditional mean and variance of the discrete process in the tree match those of the continuous process ( $M_{i,j}$  and  $V_i$ ). See [3] for the obtained formulas.

## 2.2 Tree for a two-factor process :

Now that we know how to construct a trinomial tree for a process of the kind  $dx(t) = -a x(t)dt + \sigma(t) dW(t)$ , we will use this technique for the two-factor process  $r$ .

First, we consider the process  $x$  verifying the same equation as  $r$ , with  $\theta = 0$  :

$$\begin{cases} dx(t) = [u(t) - a x(t)] dt + \sigma_1 dW_1(t), & x(0) = 0 \\ du(t) = -b u(t) dt + \sigma_2 dW_2(t), & u(0) = 0 \end{cases}$$

If we suppose that  $a \neq b$ , then the dependance of  $x$  on  $u$  can be eliminated by defining

$$y = x + \frac{u}{b-a}$$

so that

$$\begin{cases} dy(t) = -a y(t) dt + \sigma_3 dW_3(t), & y(0) = 0 \\ du(t) = -b u(t) dt + \sigma_2 dW_2(t), & u(0) = 0 \end{cases}$$

where

$$\sigma_3^2 = \sigma_3^2 + \frac{\sigma_2^2}{(b-a)^2} + \frac{2\rho\sigma_1\sigma_2}{b-a}$$

and  $W_3$  is a brownian motion. The correlation between  $W_2$  and  $W_3$  is

$$\rho_{uy} = \frac{\rho\sigma_1 + \sigma_2/(b-a)}{\sigma_3}$$

The first step to construct a tree for  $x$  is to construct two trinomial trees, with the technique explained above, for the two processes  $u$  and  $y$ , then use the formula

$$x = y - \frac{u}{b-a}$$

The tree obtained for  $x$  will be a two-dimensional trinomial tree, where every node will have nine branches, result of the combination of the tree branches of  $u$  and  $y$ .

At time  $t_i$ , we define the nodes  $y(i, h)$  and  $u(i, l)$  so the node for  $x$  is  $x(i, h, l)$ . We define  $j$  the index of middle branche (in the tree of  $y$ ) emanating from  $y(i, h)$  and the corresponding probabilities  $pu$ ,  $pm$ ,  $pd$ , and define  $k$  the index of middle branche (in the tree of  $u$ ) emanating from  $u(i, l)$  and the corresponding probabilities  $qu$ ,  $qm$ ,  $qd$ . Then, starting from  $x(i, h, l)$ , the process move to nine branches  $x(i, j + \epsilon_1, k + \epsilon_2)$ , where  $\epsilon_1$  and  $\epsilon_2$  take the values 0, 1 or -1.

Finally we have to decide the probabilities associated with every node of the nine :

In case of zero correlation between  $u$  and  $y$  (ie  $\rho_{uy} = 0$ ), the matrix of probabilities for the nine branches of  $x$  is simply :

		u-move		
		Down	Middle	Up
y-move	Down	$p_d q_d$	$p_d q_m$	$p_d q_u$
	Middle	$p_m q_d$	$p_m q_m$	$p_m q_u$
	Up	$p_u q_d$	$p_u q_m$	$p_u q_u$

In the case of correlated processes, the elements of the matrix above are shifted in such a way that the sum of shifts in each row and column is zero (to preserve the law of  $u$  and  $y$ ) and to have the good correlation  $\rho_{uy}$  between  $u$  and  $y$ . See [2] for further details.

### 2.3 Calibration of the tree to the market yield curve :

After the construction of the tree for the process  $x$ , the process  $r$  can be defined as  $r(t) = x(t) + \alpha(t)$  where  $\alpha$  is a deterministic function. It is calculated using the Arrow-Debreu node prices and the market price of Zero Coupon Bonds.

We denote by  $Q_{i+1,j,k}$  the present value of a security that pays 1 if the node  $(i+1, j, k)$  is attained and zero otherwise. These quantities are calculated recursively, knowing  $\alpha_i$  and  $Q_{i,h,l}$  for all  $(h, l)$ , by :

$$Q_{i+1,j,k} = \sum_{h,l} Q_{i,h,l} q_i(h, l, j, k) \exp\{-(\alpha_i + x_{i,h,l}) \Delta t_i\}$$

where  $q_i(h, l, j, k)$  is the probability of moving from  $(i, h, l)$  to  $(i+1, j, k)$ . Then,  $\alpha_{i+1}$  is calculated by solving :

$$P_M(0, t_{i+2}) = \sum_{i,j} Q_{i+1,j,k} \exp\{-(\alpha_{i+1} + x_{i+1,j,k}) \Delta t_{i+1}\}$$

ie :

$$\alpha_{i+1} = \frac{1}{\Delta t_{i+1}} \ln \frac{\sum_{i,j} Q_{i+1,j,k} \exp\{-x_{i+1,j,k} \Delta t_{i+1}\}}{P_M(0, t_{i+2})}$$

The initial value for  $\alpha$  and  $Q$  are :  $Q_{0,0,0} = 1$  and  $\alpha_0 = -\ln(P_M(0, t_1))/t_1$ .

## 3 Pricing of a security using the tree :

Now that we have a trinomial tree of the spot rate  $r_{i,j,k}$  with their transition probabilities we can compute the price  $h(t, r(t), u(t))$  of any european option

with payoff  $H(T, r(T), u(T))$  thanks to a backward induction, starting with  $h(T, r(T), u(T)) = H(T, r(T), u(T))$  :

$$h_{i,j,k} = e^{-r_{i,j,k}\Delta t_i} \sum_{\epsilon_1, \epsilon_2 \in \{-1, 0, 1\}} h_{i+1, j^* + \epsilon_2, k^* + \epsilon_2} q_i(j, k, j^* + \epsilon_1, k^* + \epsilon_2)$$

Where  $(i+1, j^*, k^*)$  is the index of middle branche emanating from  $(i, j, k)$  and  $q_i(j, k, j^* + \epsilon_1, k^* + \epsilon_2)$  is the probability of moving from  $(i, j, k)$  to  $(i+1, j^* + \epsilon_1, k^* + \epsilon_2)$ .

In the case of an american payoff, we compare the result of the backward induction with the payoff  $H(t_i, r_{i,j,k}, u_{i,k})$  :

$$h_{i,j,k} = \max \left( H(t_i, r_{i,j,k}, u_{i,k}), e^{-r_{i,j,k}\Delta t_i} \sum_{\epsilon_1, \epsilon_2 \in \{-1, 0, 1\}} h_{i+1, j^* + \epsilon_2, k^* + \epsilon_2} q_i(j, k, j^* + \epsilon_1, k^* + \epsilon_2) \right)$$

## References

- [1] J.Hull, A.White, *Numerical procedures for implementing term structure models I*, *Journal of Derivatives*, Fall 1994 **1**
- [2] J.Hull, A.White, *Numerical procedures for implementing term structure models II*, *Journal of Derivatives*, Winter 1994 **1, 2, 5**
- [3] D. Brigo, F. Mercurio , *Interest Rate Models - Theory and Practice: With Smile, Inflation and Credit* (Springer Finance)

**3**

## References