

Lattice methods for no-arbitrage pricing of interest rate securities

Toby Daglish

November 30, 2009

Abstract

In this paper, we explore the calibration of single factor no-arbitrage short rate models to yield and volatility information. We note that the calculation of Arrow-Debreu prices for interest rate securities is analogous to solving the Kolmogorov Forward Equation and show that the Hull-White trinomial tree approach can be seen to be using an explicit approach to solving this. We demonstrate how this insight allows us to implement implicit methods, which exhibit more rapid convergence than explicit methods. We use this insight to develop a general algorithm for calibrating the drift terms of a model to match both the yield curve and the term structure of volatilities. This approach is general across single factor short rate models, and also across finite difference techniques. We present numerical examples confirming that our approach vastly improves on computation times for interest rate derivative pricing.

The use of short-rate no-arbitrage models for pricing interest rate securities has a strong popularity amongst academics and practitioners. Being able to calibrate a model to zero coupon bond prices

ensures that pricing of derivative securities is consistent with observed interest rates, rather than being based on some best fit, as would be the case for equilibrium interest rate models.

A number of such no-arbitrage models exist. The best known of these models are Ho and Lee (1986), Black, Derman, and Toy (1990), Hull and White (1990) and Black and Karasinski (1991). While closed form solutions are available in some cases, for most implementations, the models are set up as binomial or trinomial trees, which are calibrated in order to match information on the yield curve. Hull and White (1993) show that their trinomial tree-building technique can in fact be applied to the other models in this family, and so presents a general numerical technique for no-arbitrage bond pricing.

Uhrig and Walter (1996) and Vetzal (1998) make a case that using implicit methods could improve upon the performance of trinomial models. Uhrig and Walter (1996) calibrate a time varying market price of risk to match the yield curve. Their approach entails full valuation of bonds of progressively longer maturities leading to excessive computational costs as the number of time steps increases. Vetzal (1998) avoids this shortfall by instead building in the current yield curve as a deterministic addition to his stochastic interest rate process. While this is tractable, it results in a new model for interest rates, rather than an implementation of existing models. It is also unclear how Vetzal's model could be calibrated to match volatility term structures.

Our contribution is to provide an extension of the Hull and White (1993) approach which allows the use of implicit methods. We demonstrate how the Kolmogorov Forward Equation can be solved to obtain Arrow-Debreu prices for an interest rate model. These can be used to price European derivatives and also bonds. We then demonstrate how we can use this result to derive an algorithm, similar to that used in binomial and trinomial models to choose drift terms to match the existing term structure through forward induction. Next we extend our methodology to having two parameters in

the drift term (a mean reversion rate as well as a level to which interest rates mean revert) and show how we can calibrate these to a term structure of volatilities. Our resulting algorithm is considerably faster than existing methodologies (having second order convergence in time rather than first order). This is very important for many fixed income pricing problems where securities can have lives of up to thirty years, and therefore computational burdens are sizable.

The layout of the remainder of the paper is as follows: section 1 describes the Arrow-Debreu pricing approach. Section 2 describes calibration of one parameter to match the yield curve. Section 3 generalises this to the two parameter case. Section 4 shows some numerical examples of our method, compared to a conventional tree. Lastly section 5 concludes.

1 Finding Arrow-Debreu prices

An Arrow-Debreu security is conventionally defined as a security which pays off \$1 in a particular state at a particular time. In continuous time, this can be replaced by a state price density, specifying a continuous function, which can be used as an integrand with security payoffs to determine a security's price. This density incorporates the risk-neutral probabilities of reaching a particular point, but also the discounting which is necessary to reach this point. This state price density is sometimes also referred to as a Green's function since this is the term given to a kernel which can be used to find solutions to a differential equation for arbitrary boundary conditions.

Suppose that we have a short rate process under the risk-neutral probability measure given by:

$$dr = \mu(r, t)dt + \sigma(r, t)dW, \tag{1}$$

where r is the short rate and μ and σ are arbitrary functions. t measures time and dW is the innovation of a standard Brownian motion. It can be shown that the price of a contingent claim (f) will solve the partial differential equation:

$$\frac{\partial f}{\partial t} + \mu \frac{\partial f}{\partial r} + \frac{1}{2} \sigma^2 \frac{\partial^2 f}{\partial r^2} = r f. \quad (2)$$

Pricing of a particular claim is then generally a case of solving (2) subject to appropriate boundary conditions. Equation (2) is an example of a Backward Kolmogorov Equation, coupled with discounting (the term on the right hand side). The Backward Kolmogorov Equation allows us to evaluate expectations.

Clearly one way in which the state price density could be formed would be to solve 2 for a range of different boundary conditions. However, if our objective is to simply find all state price densities, an easier approach is to use the Kolmogorov *Forward* Equation, as described in Jamshidian (1991), to find the state price density, working forward from time 0. The Kolmogorov Forward Equation (including discounting) for the process (1) is given by:

$$\frac{\partial f}{\partial t} + \left(\mu - \frac{\partial \sigma^2}{\partial r} \right) \frac{\partial f}{\partial r} + \left(\frac{\partial \mu}{\partial r} - \frac{1}{2} \frac{\partial^2 \sigma^2}{\partial r^2} \right) f - \frac{1}{2} \sigma^2 \frac{\partial^2 f}{\partial r^2} = -r f. \quad (3)$$

Note that this equation differs slightly from (2) in that the diffusion term now has a negative coefficient, and that an additional f term appears on the left hand side. Given an initial state price density ($f(r, 0)$), we could solve (3) forward to find the state price density at any future date.

1.1 Trinomial approach

Hull and White (1993) tackle the evaluation of Arrow-Debreu prices by building an explicit approximation of (2). For brevity, we focus on interior points of the tree. For these points, the transition probabilities take the form:

$$\begin{aligned} p_1(r, t) &= \frac{\sigma^2 \Delta t}{2\Delta r^2} + \frac{\mu \Delta t^2}{2\Delta r^2} + \frac{\mu \Delta t}{2\Delta r} \\ p_2(r, t) &= 1 - \frac{\sigma^2 \Delta t}{2\Delta r^2} - \frac{\mu \Delta t^2}{\Delta r^2} \\ p_3(r, t) &= \frac{\sigma^2 \Delta t}{2\Delta r^2} + \frac{\mu \Delta t^2}{2\Delta r^2} - \frac{\mu \Delta t}{2\Delta r} \end{aligned}$$

where p_1 is the probability of moving from r to $r + \Delta r$, p_2 is the probability of r not changing and p_3 is the probability of r moving to $r - \Delta r$. Δt is the timestep of the tree. Applying a Taylor expansion to $f(r, t)$ we can show that:

$$\begin{aligned} f(r, t - \Delta t) &= e^{-r\Delta t} [p_1(r, t - \Delta t)f(r + \Delta r, t) + p_2(r, t - \Delta t)f(r, t) + p_3(r, t - \Delta t)f(r - \Delta r, t)] \\ \Rightarrow 0 &= \Delta t \left[rf - \mu \frac{\partial f}{\partial r} - \frac{\sigma^2}{2} \frac{\partial^2 f}{\partial r^2} - \frac{\partial f}{\partial t} \right] + O(\Delta r^2 \Delta t, \Delta t^2) \end{aligned}$$

so that the relationship implies:

$$\frac{\partial f}{\partial t} + \mu \frac{\partial f}{\partial r} + \frac{\sigma^2}{2} \frac{\partial^2 f}{\partial r^2} = rf + O(\Delta t, \Delta r^2)$$

and the trinomial tree approximates (2) with second order accuracy in Δr and first order accuracy

in Δt .

Denoting Arrow-Debreu prices for the tree as $Q(r, t)$, the Hull-White algorithm initialises $Q(r_0, 0) = 1$ (where r_0 is the initial short rate) and then calculates the subsequent nodes in the tree as:

$$Q(r, t + \Delta t) = e^{-r\Delta t} [p_3(r + \Delta r, t)Q(r + \Delta r, t) + p_2(r, t)Q(r, t) + p_1(r - \Delta r, t)Q(r - \Delta r, t)].$$

Taking a two dimensional Taylor series around $Q(r, t)$ and exploiting the fact that:

$$\begin{aligned} \frac{\sigma^2(r + \Delta r, t) - \sigma^2(r - \Delta r, t)}{2\Delta r} &= \frac{\partial \sigma^2}{\partial r}(r, t) + \frac{1}{6} \frac{\partial^3 \sigma^2}{\partial r^3}(r, t)(\Delta r)^2 + O(\Delta r^3) \\ \frac{\sigma^2(r + \Delta r, t) - \sigma^2(r, t) + \sigma^2(r - \Delta r, t)}{\Delta r^2} &= \frac{\partial^2 \sigma^2}{\partial r^2}(r, t) + \frac{1}{12} \frac{\partial^4 \sigma^2}{\partial r^4}(r, t)(\Delta r)^2 + O(\Delta r^3) \\ \frac{\sigma^2(r + \Delta r, t) + \sigma^2(r - \Delta r, t)}{2} &= \sigma^2(r, t) + \frac{1}{2} \frac{\partial^2 \sigma^2}{\partial r^2}(r, t)(\Delta r)^2 + O(\Delta r^3) \end{aligned}$$

with similar relationships holding for μ and μ^2 , we obtain

$$\begin{aligned} Q(r, t + \Delta t) &= e^{-r\Delta t} [p_3(r + \Delta r, t)Q(r + \Delta r, t) + p_2(r, t)Q(r, t) \\ &\quad + p_1(r - \Delta r, t)Q(r - \Delta r, t)] \\ \Rightarrow \frac{\partial Q}{\partial t} \Delta t &= \Delta t \left[Q \left(\frac{1}{2} \frac{\partial^2 \sigma^2}{\partial r^2} - \frac{\partial \mu}{\partial r} - r \right) + \frac{\partial Q}{\partial r} \left(\frac{\partial \sigma^2}{\partial r} - \mu \right) + \frac{\partial^2 Q}{\partial r^2} \frac{\sigma^2}{2} \right] + O(\Delta t^2, \Delta t \Delta r^2). \end{aligned}$$

so that

$$\frac{\partial Q}{\partial t} + Q \left(\frac{\partial \mu}{\partial r} - \frac{1}{2} \frac{\partial^2 \sigma^2}{\partial r^2} \right) + \frac{\partial Q}{\partial r} \left(\mu - \frac{\partial \sigma^2}{\partial r} \right) - \frac{\partial^2 Q}{\partial r^2} \frac{\sigma^2}{2} = -rQ + O(\Delta t, \Delta r^2).$$

and so the Hull-White approach to evaluating the state price density approximates (3) with similar accuracy to its approximation of (2).

1.2 Alternative solutions

The explicit method is not the only method that could be used to solve either (2) or (3). We can construct alternative finite difference approximations. For the backward equation, we define our finite difference method as:

$$f(r, t) + w\Delta t\mathcal{L}_{\mathcal{B}}f(r, t) = f(r, t + \Delta t) - (1 - w)\Delta t\mathcal{L}_{\mathcal{B}}f(r, t + \Delta t)$$

where $\mathcal{L}_{\mathcal{B}}$ is the operator:

$$\mathcal{L}_{\mathcal{B}}f(r, t) = \mu(r, t)\frac{f(r + \Delta r, t) - f(r - \Delta r, t)}{2\Delta r} + \frac{\sigma(r, t)^2}{2}\frac{f(r + \Delta r, t) - 2f(r, t) + f(r - \Delta r, t)}{\Delta r^2} - rf(r, t).$$

Our goal is to evaluate $f(r, t)$ given information regarding $f(r, t + \Delta t)$. For $w = 0$, we obtain an explicit relationship, where a single point at time $t + \Delta t$ is calculated using three points at time t , as is the case in a trinomial tree. If $w = 1$, we obtain an implicit relationship, where three points at time $t + \Delta t$ are related to a single point at time t . For $w = 0.5$, we obtain the Crank-Nicholson approach, which relates three points at time t to three points at time $t + \Delta t$. These latter two approaches result in a system of equations with one equation for each point at the new time step, which are solved to find $f(r, t)$. To deal with points on the boundary, following Vetzal (1998), we can use the operators:

$$\begin{aligned}\mathcal{L}_{\mathcal{BL}}f(r, t) &= \mu(r, t)\frac{-f(r + 2\Delta r, t) + 4f(r + \Delta r, t) - 3f(r, t)}{2\Delta r} \\ &\quad + \frac{\sigma(r, t)^2}{2}\frac{f(r + 2\Delta r, t) - 2f(r + \Delta r, t) + f(r, t)}{\Delta r^2} - rf(r, t) \\ \mathcal{L}_{\mathcal{BU}}f(r, t) &= \mu(r, t)\frac{3f(r, t) - 4f(r - \Delta r, t) + f(r - 2\Delta r, t)}{2\Delta r} \\ &\quad + \frac{\sigma(r, t)^2}{2}\frac{f(r, t) - 2f(r - \Delta r, t) + f(r - 2\Delta r, t)}{\Delta r^2} - rf(r, t).\end{aligned}$$

where \mathcal{L}_{BL} is applied at the lower boundary and \mathcal{L}_{BU} is applied at the upper boundary. The resulting system of equations in $f(r, t)$ is tridiagonal, except for the first and last equations. It is a simple matter to eliminate the non-tridiagonal elements prior to solving the system.¹ We can develop a similar approximation for solution of (3):

$$Q(r, t + \Delta t) + \mathcal{L}_{\mathcal{F}w}Q(r, t + \Delta t)\Delta t = Q(r, t) - \Delta t\mathcal{L}_{\mathcal{F}}(1 - w)Q(r, t)$$

where $\mathcal{L}_{\mathcal{F}}$ is the operator:

$$\begin{aligned} \mathcal{L}_{\mathcal{F}}Q(r, t) = & \left(-\mu(r, t) + \frac{\partial\sigma^2}{\partial r} \right) \frac{Q(r + \Delta r, t) - Q(r - \Delta r, t)}{2\Delta r} \\ & - \frac{\sigma(r, t)^2}{2} \frac{Q(r + \Delta r, t) - 2Q(r, t) + Q(r - \Delta r, t)}{\Delta r^2} + \left(-\frac{\partial\mu}{\partial r} + \frac{\sigma(r, t)^2}{2} + r \right) Q(r, t). \end{aligned}$$

with boundary conditions:

$$\begin{aligned} \mathcal{L}_{\mathcal{F}L}Q(r, t) = & \left(-\mu(r, t) + \frac{\partial\sigma^2}{\partial r} \right) \frac{-Q(r + 2\Delta r, t) + Q(r + \Delta r, t) - 3Q(r, t)}{2\Delta r} \\ & - \frac{\sigma(r, t)^2}{2} \frac{Q(r + 2\Delta r, t) - 2Q(r + \Delta r, t) + Q(r, t)}{\Delta r^2} + \left(-\frac{\partial\mu}{\partial r} + \frac{\sigma(r, t)^2}{2} + r \right) Q(r, t) \\ \mathcal{L}_{\mathcal{F}U}Q(r, t) = & \left(-\mu(r, t) + \frac{\partial\sigma^2}{\partial r} \right) \frac{3Q(r, t) - 4Q(r - \Delta r, t) + Q(r - 2\Delta r, t)}{2\Delta r} \\ & - \frac{\sigma(r, t)^2}{2} \frac{Q(r, t) - 2Q(r - \Delta r, t) + Q(r - 2\Delta r, t)}{\Delta r^2} + \left(-\frac{\partial\mu}{\partial r} + \frac{\sigma(r, t)^2}{2} + r \right) Q(r, t). \end{aligned}$$

Note that in this situation, we wish to solve for $Q(r, t + \Delta t)$ given $Q(r, t)$ (forward induction). As is the case for the Hull-White implementation, our initial condition is that $Q(r_0, 0) = 1$ and $Q(r, 0) = 0$ for $r \neq r_0$. Also similarly to the Hull-White implementation, we can find the price of a zero coupon

¹Further discussion of finite difference techniques can be found in many textbooks. A good reference for this is Tavella and Randall (2000). Reduction to tridiagonality is discussed in Vetzal (1998) and Cairns (2004).

bond maturing at time T as $P(T) = \sum_j Q(r_j, T)$. Again $w = 0$ will result in the explicit method, $w = 1$ will result in the implicit method, and $w = 0.5$ will give the Crank-Nicholson method.

Of the three approaches, the Crank-Nicholson approach is generally taken to be the superior, since has the advantage of being unconditionally stable and having second order accuracy in the time dimension as well as the space dimension.² Our applications in section 4 will make use of the Crank-Nicholson method.

2 Calibrating to the yield curve

Hull and White (1994) finesse the approach of Hull and White (1993) for the specific case:

$$dr = (\theta(t) - \lambda r) dt + \sigma dW. \quad (4)$$

where $\theta(t)$ is an arbitrary function, chosen to match zero coupon bond prices from the model with observed yields. They suggest that an appropriate tree-building technique is to set up a tree for

$$dr = -\lambda r dt + \sigma dW. \quad (5)$$

They then solve for the Arrow-Debreu prices, at each step augmenting the interest rate by an amount $\alpha(t)$ so that the price of that time's zero coupon bond is consistent with the yield curve.³ Mathe-

²Stability, unfortunately, does not guarantee non-oscillatory behaviour. For some problems, where advection is particularly strong relative to diffusion, the Crank-Nicholson method may generate solutions with negative prices/probabilities, requiring upwind differencing in the spatial dimension. This reduces the method to second order in time, first order in space.

³Their paper shows that the relationship between $\alpha(t)$ and $\theta(t)$ is $(\theta(t) - \lambda\alpha(t))\Delta t = \alpha(t) - \alpha(t - \Delta t)$.

matically,

$$Q(r, t + \Delta t) = e^{-(r+\alpha)\Delta t} [p_3(r + \Delta r, t)Q(r + \Delta r, t) + p_2(r, t)Q(r, t) + p_1(r - \Delta r, t)Q(r - \Delta r, t)]. \quad (6)$$

If we define the price of a zero coupon bond maturing at time $t + \Delta t$, conditional on $\alpha(t)$ as being $P(t + \Delta t, \alpha(t))$, then:

$$P(t + \Delta t, \alpha(t)) = \sum_j Q(r_j, t) e^{-(\alpha(t)+r_j)\Delta t} = e^{-\alpha(t)\Delta t} \sum_j Q(r_j, t) e^{-r_j\Delta t}.$$

Hence knowing the level of $Q(r, t)$ for the previous time step and knowing the price of a zero coupon bond maturing at time $t + \Delta t$ is sufficient information to back out $\alpha(t)$. Once $\alpha(t)$ is known, the Arrow-Debreu prices can be updated using (6).

For a more general finite difference technique, we could write, in matrix notation, that

$$BQ_{t+\Delta t} = AQ_t$$

where Q_t is a vector containing the level of Q for each level of r at time t . The matrices A and B describe the system of equations that must be solved to find $Q_{t+\Delta t}$ given Q_t . We will assume that these have (as was the case of the Hull-White model) been set up to be consistent with (5). Increasing the discount rate will simply affect the right hand side of (3), so we can adjust our updating to incorporate α as:

$$(B + w\alpha(t)I)Q_{t+\Delta t} = (A - (1 - w)\alpha(t)I)Q_t \quad (7)$$

where w takes on value 1 for a fully implicit method, 0.5 for the Crank Nicholson method (discussed

previously) or 0 for an explicit method. Note that for the explicit method, the matrix B will be an identity matrix, and so there is a straightforward relationship between choice of $\alpha(t)$ and the level of $Q_{t+\Delta t}$. Hence it is easy to solve for $\alpha(t)$.

For an implicit method, matters are complicated by the appearance of $\alpha(t)$ on the left hand side of equation (7), creating an implicit relationship between choice of $\alpha(t)$ and the level of $Q_{t+\Delta t}$. This requires us to solve the problem iteratively.

Consider that we guess that the level of $\alpha(t)$ is $\hat{\alpha}(t)$, which has corresponding level of $Q_{t+\Delta t}$ of $\hat{Q}_{t+\Delta t}$. And consider perturbing $\hat{\alpha}(t)$ by $\Delta\alpha$. Then the left hand side of (7) will change by

$$w\Delta\alpha\hat{Q}_{t+\Delta t} + (B + w\hat{\alpha}(t)I)\Delta Q_{t+\Delta t}$$

where $\Delta Q_{t+\Delta t}$ is the change in time $t + \Delta t$ Arrow-Debreu prices caused by the change in $\alpha(t)$. Meanwhile, the right hand side of the equation will change by:

$$-(1 - w)\Delta\alpha Q_t.$$

Equating the changes, and solving for $\Delta Q_{t+\Delta t}$, we obtain:

$$\Delta Q_{t+\Delta t} = -\Delta\alpha(B + w\hat{\alpha}(t)I)^{-1}((1 - w)Q_t + w\hat{Q}_{t+\Delta t}). \quad (8)$$

Furthermore, we know that the price of a time $t + \Delta t$ bond ($P(t + \Delta t)$) should equal the sum of the Arrow-Debreu prices at time $t + \Delta t$. Hence we would like:

$$\mathbb{1}^T \Delta Q_{t+\Delta t} = P(t + \Delta t) - \mathbb{1}^T \hat{Q}_{t+\Delta t}, \quad (9)$$

where $\mathbb{1}$ is a vector of ones of the same length as Q_t . Substituting (8) into (9) and solving for $\Delta\alpha$, we obtain:

$$\Delta\alpha = \frac{-(P(t + \Delta t) - \mathbb{1}^T \hat{Q}_{t+\Delta t})}{\mathbb{1}^T (B + w\hat{\alpha}(t)I)^{-1}((1-w)Q_t + w\hat{Q}_{t+\Delta t})}.$$

Solving for $\alpha(t)$ thus follows the following algorithm:

1. Solve the equation (7) to obtain a preliminary level for $\hat{Q}_{t+\Delta t}$ given a guess of $\hat{\alpha}(t)$.
2. Update $\hat{\alpha}(t)$ to:

$$\hat{\alpha}(t) + \frac{\mathbb{1}^T \hat{Q}_{t+\Delta t} - P(t + \Delta t)}{\mathbb{1}^T R}$$

where the vector R solves

$$(B + w\hat{\alpha}(t)I)R = (1-w)Q_t + w\hat{Q}_{t+\Delta t}.$$

3. Repeat the iterations until $\hat{\alpha}(t)$ converges.

In practice, given a reasonable guess of $\hat{\alpha}(t)$ (such as $\alpha(t - \Delta t)$) the algorithm converges within one or two iterations. Note that this algorithm is not peculiar to the particular form of volatility term used for the short rate – it could be implemented for a geometric or square root style process.

We are thus able to update an implicit method in a similar fashion to the Hull-White tree. Section 4 contains some examples of relative performance of the Hull-White tree compared to the implicit Crank-Nicholson method.

3 Calibrating to the volatility curve

To fit not only the yield curve, but also the volatility curve, it becomes necessary to introduce two functions of time into the equation. Following Hull and White (1993), we consider:

$$dr = (\theta(t) - \phi(t)r)dt + \sigma r^\beta dW.$$

Here $\theta(t)$ and $\phi(t)$ are arbitrary functions of time, chosen to now match the yield curve *and* observed yield volatilities. Intuitively, if the mean-reversion speed ($\phi(t)$) increases in t , it will damp later volatilities, while if it declines, later volatilities will be higher.

As noted in Hull and White (1993) this technique generalises to transformations of r so that (for example) the Black-Karasinski model can be implemented in a similar fashion. We refer readers to Hull and White (1993) for complete treatment of their approach. Their algorithm begins by using a binomial step where the entire yield curve is shifted up or down by a sufficient amount to match the volatility curve specified. The algorithm then chooses $\phi(t)$ and $\theta(t)$ at each step so that the value of a zero coupon bond maturing at that time, conditional on the initial binomial step having been up, or having been down, are consistent with the zero coupon bond price observed at time zero.

Our algorithm is similar in spirit to our approach for matching the yield curve. Firstly, we note that the volatility of an interest rate can be written in terms of the volatility of a bond's price:

$$V(t) = \frac{\frac{\partial P(t)}{\partial r} \sigma}{P(t)t}.$$

Note that we define volatility in absolute terms (the diffusion term of the process for the interest rate in question). Adjusting to be a proportional volatility simply requires dividing $V(t)$ by $R(t) \equiv$

$-\log(P(t))/t$.

We know that $P(t)$ is equal to the sum of $Q(r, t)$ across all values of r at time t , so we may write:

$$V(t) \simeq \frac{\sigma}{2\Delta r t} \frac{\mathbb{1}^T(Q_t^u - Q_t^d)}{\mathbb{1}^T Q_t} \quad (10)$$

where Q_t^u is the vector of Arrow-Debreu prices consistent with the time zero interest rate being equal to $r_0 + \Delta r$ and Q_t^d is the vector of Arrow-Debreu prices at time t consistent on the time zero interest rate being equal to $r_0 - \Delta r$. Note that Q^u and Q^d both satisfy (3) but with the boundary conditions such that $Q^u(r, 0)$ is zero except when $r = r_0 + \Delta r$ (where it equals one) and $Q^d(r, 0)$ is zero except when $r = r_0 - \Delta r$ (where it similarly equals one).

Our conditions we require to hold are thus (10) and the familiar

$$\mathbb{1}^T Q_t = P(t)$$

with both equations required to hold for each time step.

Our algorithm proceeds similarly to that in section 2. At each time step, we will iteratively find $\theta(t)$ and $\phi(t)$, given some initial guess $\hat{\theta}(t)$ and $\hat{\phi}(t)$. We will assume that the updating equations, including $\phi(t)$ and $\theta(t)$ are given by:

$$(B + w\hat{\theta}(t)D_1 + w\hat{\phi}(t)D_2)\hat{Q}_{t+\Delta t} = (A - (1-w)\hat{\theta}(t)D_1 - (1-w)\hat{\phi}(t)D_2)Q_t \quad (11)$$

where D_1 and D_2 are matrices representing the terms in the finite difference equations which have θ

and ϕ as a coefficient (see section 1.2). We will also define:

$$Z(\theta(t), \phi(t)) \equiv B + w\theta(t)D_1 + w\phi(t)D_2.$$

Now consider the effect of changing $\hat{\theta}$ by $\Delta\theta$ and $\hat{\phi}$ by $\Delta\phi$. The effect on the left hand side of (11) is:

$$\Delta\theta w D_1 \hat{Q}_{t+\Delta t} + \Delta\phi w D_2 \hat{Q}_{t+\Delta t} + Z(\hat{\theta}, \hat{\phi}) \Delta Q_{t+\Delta t},$$

while the effect on the right hand side of (11) is:

$$(-\Delta\theta(1-w)D_1 - \Delta\phi(1-w)D_2)Q_t.$$

We can equate these and solve for $\Delta Q_{t+\Delta t}$ as:

$$\Delta Q_{t+\Delta t} = -Z^{-1}(\Delta\phi D_2 + \Delta\theta D_1) \left(w\hat{Q}_{t+\Delta t} + (1-w)Q_t \right). \quad (12)$$

Defining

$$\begin{aligned} R_1 &= Z^{-1}D_1 \left(w\hat{Q}_{t+\Delta t} + (1-w)Q_t \right) \\ R_2 &= Z^{-1}D_2 \left(w\hat{Q}_{t+\Delta t} + (1-w)Q_t \right), \end{aligned}$$

and setting $\Delta Q_{t+\Delta t} = P(t + \Delta t) - \mathbb{1}^T \hat{Q}_{t+\Delta t}$, we obtain the condition:

$$P(t + \Delta t) - \mathbb{1}^T \hat{Q}_{t+\Delta t} = -\Delta\phi \mathbb{1}^T R_2 - \Delta\theta \mathbb{1}^T R_1. \quad (13)$$

We also need the volatility condition to hold for time $t + \Delta t$. We can use the quotient rule to determine the effect of perturbing $\hat{\theta}(t + \Delta t)$ and $\hat{\phi}(t + \Delta t)$ on the level of volatility:

$$\Delta V = \frac{-\sigma}{2\Delta r(t + \Delta t)} \left[\frac{\mathbb{1}^T (\Delta Q_{t+\Delta t}^u - \Delta Q_{t+\Delta t}^d)}{\mathbb{1}^T \hat{Q}_{t+\Delta t}} - \frac{\mathbb{1}^T (\hat{Q}_{t+\Delta t}^u - Q_{t+\Delta t}^d)}{(\mathbb{1}^T \hat{Q}_{t+\Delta t})^2} \mathbb{1}^T \Delta Q_{t+\Delta t} \right]. \quad (14)$$

Using (12) applied to Q^u and Q^d , we can rewrite (14) as:

$$\begin{aligned} \Delta V &= \frac{\sigma}{2\Delta r(t + \Delta t)} \left[\frac{\mathbb{1}^T (Z^{-1}(\Delta\phi D_2 + \Delta\theta D_1) (w\hat{Q}_{t+\Delta t}^u + (1-w)Q_t^u - w\hat{Q}_{t+\Delta t}^d - (1-w)Q_t^d))}{\mathbb{1}^T \hat{Q}_{t+\Delta t}} \right. \\ &\quad \left. - \frac{\mathbb{1}^T (\hat{Q}_{t+\Delta t}^u - Q_{t+\Delta t}^d)}{(\mathbb{1}^T \hat{Q}_{t+\Delta t})^2} \mathbb{1}^T Z^{-1}(\Delta\phi D_2 + \Delta\theta D_1)(w\hat{Q}_{t+\Delta t} + (1-w)Q_t) \right] \\ &= \Delta\phi \frac{\sigma}{2\Delta r(t + \Delta t)} \left[\frac{\mathbb{1}^T R_4}{\mathbb{1}^T \hat{Q}_{t+\Delta t}} - \frac{\mathbb{1}^T (\hat{Q}_{t+\Delta t}^u - \hat{Q}_{t+\Delta t}^d)}{(\mathbb{1}^T \hat{Q}_{t+\Delta t})^2} \mathbb{1}^T R_2 \right] \\ &\quad + \Delta\theta \frac{\sigma}{2\Delta r(t + \Delta t)} \left[\frac{\mathbb{1}^T R_3}{\mathbb{1}^T \hat{Q}_{t+\Delta t}} - \frac{\mathbb{1}^T (\hat{Q}_{t+\Delta t}^u - \hat{Q}_{t+\Delta t}^d)}{(\mathbb{1}^T \hat{Q}_{t+\Delta t})^2} \mathbb{1}^T R_1 \right] \end{aligned} \quad (16)$$

where R_1 and R_2 are as given above and:

$$\begin{aligned} R_3 &= Z^{-1} D_1 (w\hat{Q}_{t+\Delta t}^u + (1-w)Q_t^u - wQ_{t+\Delta t}^d - (1-w)Q_t^d) \\ R_4 &= Z^{-1} D_2 (w\hat{Q}_{t+\Delta t}^u + (1-w)Q_t^u - wQ_{t+\Delta t}^d - (1-w)Q_t^d). \end{aligned}$$

Equating (16) to $V(t) - \hat{V}(t)$, where $\hat{V}(t)$ is given by

$$\hat{V}(t + \Delta t) \simeq \frac{\sigma}{2\Delta r(t + \Delta t)} \frac{\mathbb{1}^T (\hat{Q}_{t+\Delta t}^u - \hat{Q}_{t+\Delta t}^d)}{\mathbb{1}^T \hat{Q}_{t+\Delta t}},$$

we obtain a second condition for $\Delta\phi$ and $\Delta\theta$. We can summarise the two conditions as:

$$\begin{bmatrix} \kappa_1 & \kappa_2 \\ \kappa_3 & \kappa_4 \end{bmatrix} \begin{bmatrix} \Delta\theta \\ \Delta\phi \end{bmatrix} = \begin{bmatrix} P(t + \Delta t) - \mathbb{1}^T Q(t + \Delta t) \\ V(t + \Delta t) - \hat{V}(t + \Delta t) \end{bmatrix} \quad (17)$$

where

$$\begin{aligned} \kappa_1 &= -\mathbb{1}^T R_1 \\ \kappa_2 &= -\mathbb{1}^T R_2 \\ \kappa_3 &= \frac{\sigma}{2\Delta r(t + \Delta t)} \left[\frac{\mathbb{1}^T R_3}{\mathbb{1}^T \hat{Q}_{t+\Delta t}} - \frac{\mathbb{1}^T (\hat{Q}_{t+\Delta t}^u - \hat{Q}_{t+\Delta t}^d)}{(\mathbb{1}^T \hat{Q}_{t+\Delta t})^2} \mathbb{1}^T R_1 \right] \\ \kappa_4 &= \frac{\sigma}{2\Delta r(t + \Delta t)} \left[\frac{\mathbb{1}^T R_4}{\mathbb{1}^T \hat{Q}_{t+\Delta t}} - \frac{\mathbb{1}^T (\hat{Q}_{t+\Delta t}^u - \hat{Q}_{t+\Delta t}^d)}{(\mathbb{1}^T \hat{Q}_{t+\Delta t})^2} \mathbb{1}^T R_2 \right]. \end{aligned}$$

Stepping forward in time thus consists of alternating between updating $(\hat{Q}_{t+\Delta t}, \hat{Q}_{t+\Delta t}^u, \hat{Q}_{t+\Delta t}^d)$ using (11) and solving (17) to update $\hat{\phi}(t + \Delta t)$ and $\hat{\theta}(t + \Delta t)$. Again, using $\phi(t)$ and $\theta(t)$ as starting conditions for $\hat{\phi}(t + \Delta t)$ and $\hat{\theta}(t + \Delta t)$, this process only takes one or two iterations.

Again, we note that this algorithm is general across all finite difference techniques.

4 Numerical examples

We consider two test problems, to demonstrate our new technique. In both cases, we make use of the Crank-Nicholson algorithm, described in section 1.2.

Maturity	Yield
0.5	0.03430
1.0	0.03824
1.5	0.04183
2.0	0.04512
2.5	0.04812
3.0	0.05086

Table 1: Yield curve for examples in section 4.1. All rates expressed with continuous compounding.

4.1 Calibration to yield curve

For this example, we consider the standard Hull-White model given by (4) and implement the procedures described in section 2. We implement the traditional Hull-White tree, and also the Crank-Nicholson algorithm. The yield curve is given in table 1, and is assumed to be piecewise linear for points between those given. We consider the pricing of a two year call option, written on a three year zero coupon bond with face value \$1, where the option has strike price \$0.943. We assume a mean reversion rate of $\lambda = 0.1$. A closed form solution exists for options on zero coupon bonds in this model (see Hull and White (1990)). The correct price for this security is 0.0028. In table 2 we present relative pricing error, as time step is decreased. Since the Crank-Nicholson method is second order, we use a timestep chosen as the square-root of the time step used by the Trinomial tree. Given its second order accuracy in time, this gives comparable accuracy performance.

Table 2 gives a clear picture of the advantages of using the new algorithm. For large time steps (low accuracy) the advantage of not needing to either solve the implicit equations nor iterate to find α renders the trinomial method more efficient, however, as more accuracy is required, the second order convergence of the Crank-Nicholson method more than outweighs its more involved calculations.⁴

⁴It should be noted that while the Crank-Nicholson method does require more work than the Trinomial tree, the effort required to calculate one step is still $O(N)$ where N is the number of spatial steps in the lattice.

Timestep (Tri)	Timestep (CN)	Error (Tri)	Error (CN)	Time (Tri)	Time (CN)
0.5000	0.5000	0.0363	0.0141	0.0037	0.0105
0.2500	0.5000	0.0292	0.0141	0.0134	0.0105
0.1000	0.2500	0.0044	-0.0024	0.0679	0.0327
0.0500	0.2000	-0.0071	0.0024	0.1307	0.0631
0.0250	0.1429	0.0030	-0.0025	0.5135	0.1687
0.0100	0.1000	0.0009	0.0001	3.1802	0.4969

Table 2: Performance of algorithms for exercise in section 4.1. Tri refers to the Hull-White trinomial tree, while CN refers to the Crank-Nicholson algorithm. For the Hull-White model, we assume that $\Delta r = \sigma\sqrt{3\Delta t}$, with boundaries set at $\pm\frac{0.184}{\lambda\Delta t}$ as suggested in Hull and White (1994). For the Crank-Nicholson method, $\Delta r = \sigma\sqrt{2\Delta t}$, with boundaries set at $\pm\frac{5\sigma}{\sqrt{2\lambda}\Delta r}$, as suggested by Cairns (2004). Time taken is in seconds running MATLAB on an AMD Opteron 148.

Maturity	Yield	Volatility
1	0.0525	0.006825
2	0.055	0.0066
3	0.0575	0.006325
4	0.06	0.006
5	0.0625	0.005625
6	0.065	0.0052

Table 3: Yield curve and volatility curve for examples in section 4.2. All rates expressed with continuous compounding. Volatility is absolute.

4.2 Calibration to yield curve and volatility

Our second numerical example follows the methodology of section 3, using a yield curve and volatility curve as given in table 3. We assume $\beta = 0$ so that we are calibrating an extended-Vasicek style model. We consider pricing a 4 year call option written on a 5 year zero coupon bond, with strike price 0.933. The analytical price for this security is 0.0019. We then compare performance of our Crank-Nicholson algorithm to a Hull-White trinomial tree. Again, we consider using a larger time step for the Crank-Nicholson method, since it has a higher order convergence rate. Results are given in table 4. Here some additional numerical costs are borne by the Crank-Nicholson approach. The algorithm

Timestep (Tri)	Timestep (CN)	Error (Tri)	Error (CN)	Time (Tri)	Time (CN)
0.5000	0.5000	0.0649	-0.0150	0.0042	0.0310
0.2500	0.5000	0.0646	-0.0150	0.0057	0.0310
0.1000	0.2500	0.0270	-0.0038	0.0284	0.0470
0.0500	0.2000	0.0068	-0.0028	0.1036	0.0798
0.0250	0.1429	0.0074	-0.0012	0.4189	0.2149
0.0100	0.1000	0.0030	-0.0006	3.2740	0.7482

Table 4: Performance of algorithms for exercise in section 4.2. Tri refers to the Hull-White trinomial tree, while CN refers to the Crank-Nicholson algorithm. For the Hull-White model, we assume that $\Delta r = \sigma\sqrt{3\Delta t}$, with boundaries set at $\pm\frac{0.184}{\lambda\Delta t}$ as suggested in Hull and White (1994). For the Crank-Nicholson method, we use $\Delta r = \sqrt{2}\sigma\Delta t$, with boundaries set at $\pm\frac{5\sigma}{\sqrt{0.1}\Delta r}$, consistent with Cairns' analysis with a level of mean reversion of 0.05. Time taken is in seconds running MATLAB on an AMD Opteron 148.

in section 3 requires 7 matrix inversions for each iteration on ϕ and θ . Each inversion requires solution of a different system of equations. In contrast, for section 3, the same set of equations are solved at each time. It is thus possible to adjust these equations to render them tridiagonal once, and then simply solve this system for each calculation. The added burden of re-tridiagonalising, increases the time taken for the Crank-Nicholson algorithm for this case.

This added burden notwithstanding, we find that the Crank-Nicholson method substantially outperforms the Trinomial tree. Using a time step of 0.25, we obtain a solution which has 0.3% error. To achieve a comparable error with the Trinomial tree, a time step of 0.01 would be required, taking approximately 40 times as much processor time as the Crank-Nicholson algorithm requires.

5 Conclusion

We have presented a methodology for extending the trinomial tree approach to calibrating short rate models to more general classes of finite difference techniques. Our methodology is applicable across

a wide range of different models, covering most of the widely used no-arbitrage short rate models.

Being able to use higher order methods that are frequently used in conventional options pricing results in a far more efficient pricing of securities. Given that many fixed income securities have very long lives (often up to 30 years) computational speed is highly important. For researchers in this area, the ability to evaluate bond prices quickly is also enormously useful, when dealing with large fixed income data sets, or where the algorithm must be implemented many times, such as when other model parameters must be estimated using time-series data.

References

- Black, Fischer, Emanuel Derman, and William Toy, 1990, A one-factor model of interest rates and its application to treasury bond options, *Financial Analysts Journal* 46, 33–39.
- Black, Fischer, and Piotr Karasinski, 1991, Bond and option pricing when short rates are lognormal, *Financial Analysts Journal* 47, 52–59.
- Cairns, Andrew J. G., 2004, *Interest Rate Models: An Introduction* (Princeton University Press: Princeton, New Jersey).
- Ho, Thomas S.Y., and Sang Bin Lee, 1986, Term structure movements and the pricing of interest rate claims, *Journal of Finance* 41, 1011–1029.
- Hull, John, and Alan White, 1990, Pricing interest rate derivative securities, *Review of Financial Studies* 3, 573–592.
- , 1993, One-factor interest-rate models and the valuation of interest-rate derivative securities, *Journal of Financial and Quantitative Analysis* 28, 235–254.

- , 1994, Numerical procedures for implementing term structure models i: Single-factor models, *Journal of Derivatives* 2, 7–16.
- Jamshidian, Farshid, 1991, Forward induction and construction of yield curve diffusion models, *Journal of Fixed Income* 1, 62–74.
- Tavella, Domingo, and Curt Randall, 2000, *Pricing Financial Instruments: The Finite Difference Approach* (John Wiley and Sons: New York).
- Uhrig, Marliese, and Ulrich Walter, 1996, A new numerical approach for fitting the initial yield curve, *Journal of Fixed Income* 5, 82–90.
- Vetzal, Kenneth R., 1998, An improved finite difference approach to fitting the initial term structure, *Journal of Fixed Income* 7, 62–81.