Trinomial Tree Algorithm for Barrier Option

A trinomial tree can be used for pricing particular types of barrier options. We consider particular types of single barrier and double barrier options. The single barrier options include certain types of:

Down and In (D_IN) calls and puts,

Down and Out (D_OUT) calls and puts,

Up and In (UP_IN) calls and puts, and

Up and Out (UP_OUT) calls and puts.

We specify these types next.

A D_IN call option specification, for example, includes the exercise type (i.e., either American or European), an exercise time, T, a strike level, X, a rebate value, R (where $R \geq 0$), and a barrier level, $H_d(t)$, which depends continuously on time over the interval $[0, T]$.

Here the underlying security is any security whose price, $S(t)$, can be modeled as a piecewise geometric Brownian motion over the life of the option. In addition we require that the initial spot level for the underlying security, $S(0)$, lie above the initial barrier level, $H_d(0)$.

The payoff for a long, European D_IN call is that of a standard European call, provided that the underlying falls below the barrier level at some time during the option's life (i.e., $S(t) \leq H(t)$, for some $t \in [0,T]$); otherwise, the payoff at expiry is equal to the fixed rebate, R. Formally a long, European D_IN call option payoff is defined, at exercise time T, as

$$\begin{cases} \max(S(T)-X,0), & \text{if } S(t) \leq H_d(t), \text{ for some } t \in [0,T], \\ R, & \text{otherwise.} \end{cases}$$

In contrast a long, American D_IN call can be exercised immediately after it has knocked in (i.e., after the underlying security has crossed the lower barrier level); if the option does not knock in, then the fixed rebate, R, is received at maturity.

As mentioned above, we also consider certain types of double barrier options. These options include particular types of:

Down and Out or *Up and Out* (D_OUT_OR_UP_OUT) calls and puts,

Down and In and *Up and Out* (D_IN_AND_UP_OUT) calls and puts,

Down and Out and *Up and In* (D_OUT_AND_UP_IN) calls and puts,

Down and In or *Up and In* (D_IN_OR_UP_IN) calls and puts,

Down and Out and *Up and Out* (D_OUT_AND_UP_OUT) calls and puts, and

Down and In and *Up and In* (D_IN_AND_UP_IN) calls and puts.

The double barrier options above allow only for European exercise. Below we provide specifications for certain of these options.

A D_OUT_OR_UP_OUT European call option specification, for instance, includes an exercise time, a strike level, and a rebate value (we denote these parameters as above). In addition two barrier levels, $H_u(t)$ and $H_d(t)$, which depend continuously on time over the interval $[0,T]$, must be specified (here $H_u(t) > H_d(t)$, for all $t \in [0,T]$). Note that the initial price of the underlying security, $S(0)$, must lie between the initial upper and lower barrier values, $H_u(0)$ and $H_d(0)$.

The payoff from a long European D_OUT_OR_UP_OUT call option is equal to that of a long, standard European call option, if the price of the underlying security does not cross either the upper or lower barrier during the time interval $[0,T]$ (i.e., if $H_d(t) < S(t) < H_u(t)$, for all $t \in [0,T]$); otherwise, as *soon as* one of these barriers is touched, a rebate value of R is received. Note that, of the double barrier options above, the D_OUT_OR_UP_OUT option is the only option that allows for a rebate. Specifications for the remaining double barrier options above are provided in [Myint, 1996a]

We also consider two types of knockout annuities, *Down and Out* and *Up and Out.* If we are long such a knockout annuity, we receive a fixed coupon annuity until the price of the underlying security crosses a preset barrier level; we then receive the accrued annuity since the last pay date. Note that only European exercise is permitted for the knockout annuities above, and no rebates are allowed.

Analytic formulas for pricing barrier options do not exist for the case where the barrier is an arbitrary, continuous function of time or where the exercise type is American. Tree methods (e.g., trinomial or binomial) can, however, be used to approximate the price of barrier options.

Unfortunately standard tree methods, when applied to price barrier options, suffer from several drawbacks, that is, these methods may converge very slowly and/or display a persistent bias in the price. The disadvantages above are due to the inability of standard tree methods to ensure, for example, for a single barrier option, that a layer of tree nodes always coincides with the barrier.

In such a case, then, the tree method effectively prices a different option (i.e., with a new barrier). An interesting, new trinomial tree method is presented for overcoming the above specification error in the barrier. The idea of the method is to construct a tree lattice, for example, for a single barrier option, by ensuring that certain nodes near the barrier always branch onto the barrier.

Next we present the methods for pricing the types of barrier options described in Section 2. Each method is based on a combination of techniques, that is, a tree generation technique and a backward induction pricing technique. Below we describe the tree generation techniques for both single barrier and double barrier. We then describe backward induction techniques for the types of options considered.

Let $0 = t_0 < \cdots < t_N = T$ be a partition of the time interval $[0, T]$. Furthermore suppose that the underlying security follows piecewise geometric Brownian motion, in the sense described below, over the interval $[0, T]$. Specifically, assume that the underlying security can be modeled as a process, $\left\{ S(t) | t \in [0, T] \right\}$, which, under the risk neutral probability measure, satisfies a stochastic differential equation (SDE) of the form

$$dS(t) = \mu(t) S(t) dt + \sigma(t) S(t) dW(t), \quad t \in [0, T], \tag{1}$$

where $\{W(t) | t \in [0,T]\}$ is standard Brownian motion. Here $\mu(t)$ and $\sigma(t)$ are deterministic functions of the piecewise constant form

$$\mu(t) = \begin{cases} \mu_1, & t \in [0, t_1), \\ \vdots & \\ \mu_N, & t \in [t_{N-1}, t_N], \end{cases} \quad \text{and} \quad \sigma(t) = \begin{cases} \sigma_1, & t \in [0, t_1), \\ \vdots & \\ \sigma_N, & t \in [t_{N-1}, t_N], \end{cases} .$$

Each method includes a technique for constructing, based on the SDE (1), an appropriate tree of discrete prices of the underlying security. Each such technique uses a mathematical result, described below, for ensuring that branching probabilities from each tree node are appropriate (i.e., probabilities, for each node, must be non-negative and sum to one).

Consider a tree node, ω, at a time slice, t_i, where $0 \le i < N$; furthermore, assume that the logarithm of the price of the underlying security at this node is equal to $\log Sold$. We assume that node ω branches into three nodes, at time slice t_{i+1}, with respective logarithm of the price of the underlying security of the form $(\lambda+1)\Delta \log Snew$, $(\lambda)\Delta \log Snew$, and $(\lambda-1)\Delta \log Snew$ where $\lambda \in \Re$ and $\Delta \log Snew > 0$.

Here $(\lambda)\Delta \log Snew$ is the value that, among all tree nodes at time t_{i+1}, is closest to $\log Sold + \hat{\mu}_{i+1}\Delta t_{i+1}$ where $\hat{\mu}_{i+1} = \mu_{i+1} - \dfrac{\sigma_{i+1}^2}{2}$ and $\Delta t_{i+1} = t_{i+1} - t_i$; furthermore, $(\lambda+1)\Delta \log Snew$ and $(\lambda-1)\Delta \log Snew$ are values for the two nodes closest to the node with value $(\lambda)\Delta \log Snew$. Next we associate with node ω a discrete random variable, Y, which takes the values

$$Y = \begin{cases} (\lambda+1)\Delta \log Snew, & \text{with probability } p_u, \\ \lambda(\Delta \log Snew), & \text{with probability } p_m, \\ (\lambda-1)\Delta \log Snew, & \text{with probability } p_{d.} \end{cases}$$

We seek to determine p_u, p_m and p_d, above, so that the mean and variance of the discrete random variable Y match those of the continuous random variable $\log Sold + \hat{\mu}_{i+1}\Delta t_{i+1} + \sigma_{i+1} W_{\Delta_{t_{i+1}}}$ (obtained by solving the SDE (1), with initial condition $\log S(t_i) = \log Sold$, for the time interval $[t_i, t_{i+1}]$).

By matching mean and variances as described above, and by ensuring that the probabilities sum to one, we obtain the following system of linear equations

$$\begin{cases} p_u(\lambda+1)\Delta \log Snew + p_m \lambda(\Delta \log Snew) + p_d(\lambda-1)\Delta \log Snew = \log Sold \\ \qquad + \hat{\mu}_{i+1}\Delta t_{i+1}, \\ p_u(\lambda+1)^2(\Delta \log Snew)^2 + p_m(\lambda)^2(\Delta \log Snew)^2 + \\ \qquad p_d(\lambda-1)^2(\Delta \log Snew)^2 = {\sigma_{i+1}}^2 \Delta t_{i+1} + (\log Sold + \hat{\mu}_{i+1}\Delta t_{i+1})^2, \\ p_u + p_m + p_d = 1, \end{cases} \tag{2}$$

for the unknowns p_u, p_d and p_m. By algebraic manipulation, the linear system of equations above is equivalent to

$$\begin{cases} p_u - p_d = B - \lambda, \\ p_u(1+2\lambda) + p_d(1-2\lambda) = A^2 + B^2 - \lambda^2, \\ p_u + p_m + p_d = 1, \end{cases} \tag{3}$$

Where

$$A^2 = \frac{\sigma_{i+1}{}^2 \Delta t_{i+1}}{(\Delta \log Snew)^2} \qquad (4a)$$

and

$$B = \frac{\log Sold + \hat{\mu}_{i+1} \Delta t_{i+1}}{\Delta \log Snew}. \qquad (4b)$$

Notice that while the system of equations above has a unique solution, we have no guarantee that p_u, p_m and p_d will be non-negative. Next we determine a condition on A to ensure that $p_u, p_m, p_d \geq 0$.

Recall that the branching rule from node ω implies that

$$(\lambda + \frac{1}{2}) \Delta \log Snew \geq \log Sold + \hat{\mu}_{i+1} \Delta t_{i+1} \geq (\lambda - \frac{1}{2}) \Delta \log Snew. \qquad (5)$$

Dividing (5) by $\Delta \log Snew$ and substituting B for the right hand side of (4b), we have $\lambda + \frac{1}{2} \geq B \geq \lambda - \frac{1}{2}$, which we can rewrite as

$$B = \lambda - \frac{1}{2} + x \tag{6}$$

for some $x \in [0,1]$. Solving (3) and substituting the right hand side of (4b) for B, we obtain

$$\begin{cases} p_u = \dfrac{A^2}{2} + \dfrac{x^2}{2} - \dfrac{1}{8}, \\ p_m = \dfrac{3}{4} + x - x^2 - A^2, \\ p_d = \dfrac{A^2}{2} + \dfrac{x^2}{2} - x + \dfrac{3}{8}, \end{cases} \tag{7}$$

where $x \in [0,1]$. Notice that the right hand side of (7) has no dependency on λ.

Analyzing the right hand side of (7) over the range $x \in [0,1]$, we obtain the condition

$$\frac{3}{4} \geq A^2 \geq \frac{1}{4} \tag{8}$$

on A, which ensures that $p_u, p_m, p_d \geq 0$. Notice that (8) yields an equivalent condition on $\Delta \log Snew$, that is,

$$\frac{2\sigma_{i+1}\sqrt{\Delta t_{i+1}}}{\sqrt{3}} \leq \Delta \log Snew \leq 2\sigma_{i+1}\sqrt{\Delta t_{i+1}} \; . \tag{9}$$

To summarize, for an arbitrary tree node on an arbitrary time slice, appropriate branching probabilities are given as the solution of (2) provided that condition (9) holds.

Reference:

https://finpricing.com/lib/IrBasisCurve.html