Trinomial tree

By Xingjian Yin

Overview

The trinomial tree is a lattice-based computational model used in financial mathematics to price options. It was developed by Phelim Boyle in 1986. It is an extension of the binomial options pricing model, and is conceptually similar. It can also be shown that the approach is equivalent to the explicit finite difference method for option pricing.[1] For fixed income and interest rate derivatives see Lattice model (finance)#Interest rate derivatives. -----Wikipedia

Formula

Recombining Tree Structure

To keep the tree “recombining” (so that different sequences of moves may lead to the same price), we choose the up and down factors such that

$$d = \frac{1}{u}$$

A common choice (especially in one popular formulation) is:

$$u = e^{\sigma\sqrt{2\,\Delta\,t}}, \quad d = e^{-\sigma\sqrt{2\,\Delta\,t}}, \quad m = 1$$

where σ is the volatility of the asset.

Risk-Neutral Probabilities

The model assigns risk-neutral probabilities to the three outcomes—p_u for an up move, p_d for a down move, andp_m for the middle (unchanged) move—with:

$$p_u + p_m + p_d = 1$$

These probabilities are chosen so that the expected price of the asset (under the risk-neutral measure) grows at the risk-free rate rrr and the variance matches that of the continuous process. One popular set of formulas is:

$$p_u = \left(\frac{e^{(r-q)\,\Delta\, t/2} - e^{-\sigma\sqrt{\Delta\, t/2}}}{e^{\sigma\sqrt{\Delta\, t/2}} - e^{-\sigma\sqrt{\Delta\, t/2}}}\right)^2$$

$$p_d = \left(\frac{e^{\sigma\sqrt{\Delta\, t/2}} - e^{(r-q)\,\Delta\, t/2}}{e^{\sigma\sqrt{\Delta\, t/2}} - e^{-\sigma\sqrt{\Delta\, t/2}}}\right)^2$$

$$p_m = 1 - (p_u + p_d)$$

where q is the dividend yield (which may be set to zero for a non-dividend–paying asset).

Option Valuation

After constructing the asset-price lattice over NNN time steps, you evaluate the option at expiration (the terminal nodes) using the payoff function (for example, for a European call: $max(S - K, 0)$. Then you work backward—using the risk-neutral probabilities and discounting at the risk-free rate—to obtain the option's current value.

A Simple Example: A One-Step Trinomial Tree

Let's consider pricing a European call option with these parameters:

- **Initial stock price:** $S_0 = \$100$
- **Strike price:** $K = \$100$
- **Risk-free rate:** $r = 5\%$ per year
- **Volatility:** σ=20% per year
- **Time to maturity:** T=1 year
- **Number of time steps:** N=1 (for simplicity, so Δt=1 year)

Step 1. Compute the Up, Down, and Middle Factors

Using a common specification:

For Δt=1 year:

- Calculate $\sqrt{2 \Delta t} = \sqrt{2} \approx 1.414$
- Then:

$$\boldsymbol{u = e^{0.20 \times 1.414} = e^{0.2828} \approx 1.327}, \ d = e^{-0.2828} \approx 0.754,$$

m=1.

Step 2. Determine the Risk-Neutral Probabilities

For simplicity, let's assume zero dividend (q=0) and use the following formulas (with Δt=1 year):

$$p_u \approx 0.3066$$

$$p_d \approx 0.1996$$

$$p_m \approx 0.4938$$

Step 3. Construct the Tree and Compute Option Payoffs

At time t=1 year, the possible stock prices are:

Up move:$S_u = S_0 \times u = 100 \times 1.327 \approx 132.70.$

Middle move: $S_m = S_0 \times m = 100 \times 1 = 100.$

Down move: $S_d = S_0 \times d = 100 \times 0.754 \approx 75.40$

Step 4. Discount the Expected Payoff

Now, the option's present value is the discounted expected payoff:

$$\text{Option Price} = e^{-r \Delta t} \left(p_u \times 32.70 + p_m \times 0 + p_d \times 0\right)$$

With $r\Delta t = 0.05$ and $e^{-0.05} \approx 0.9512$:

$$\text{Option Price} \approx 0.9512 \times (0.3066 \times 32.70) \approx 0.9512 \times 10.03 \approx 9.54.$$

So, in this one-step example the European call option is valued at approximately $9.54

Convergence of Trinomial Tree

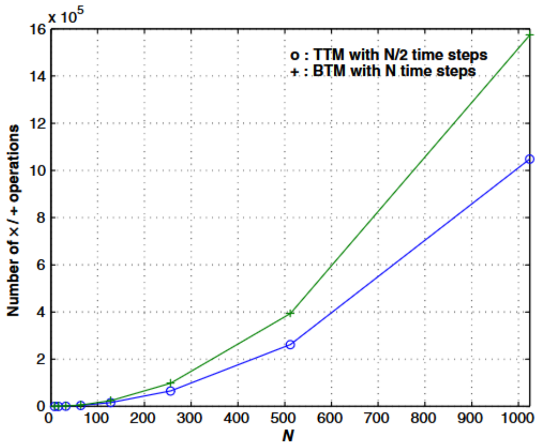
Compared with Binomial Tree, the trinomial tree converges to the BSM pricing much faster under same steps and the difference from BSM is rather small. I have written a program and here is a table of their differences:

Suppose $S_0 = 100, \ K = 90, \ r = 0.05, \ \sigma = 0.4, \ T = 0.25$

Then:

Step	BTP	TTP	BT Diff	TT Diff
1	10.5260	6.2823	-18.747	36.1380
2	7.6502	7.3600	11.7958	16.2038
3	9.2196	7.7908	-7.2346	9.7782
4	8.0766	8.0036	5.8936	6.8594
5	8.9506	8.1252	-4.4466	5.2602
6	8.2304	8.2027	3.9148	4.2657
7	8.8358	8.2562	-3.2051	3.5900
8	8.3092	8.2954	2.9293	3.1005
9	8.7723	8.3253	-2.5045	2.7302
10	8.3571	8.3489	2.3393	2.4398

And here is a graph of the cost to run the code:



Summary

A **trinomial tree** is a numerical method used to model the possible future prices of financial assets, especially in option pricing. Unlike the binomial tree, which allows only two outcomes at each step (up or down), the trinomial tree introduces a third possibility—no change—at each time step. This three-branch structure offers greater accuracy and stability, especially with fewer time steps. The asset price can move up, down, or stay the same, with probabilities adjusted to match the expected return and volatility under the risk-neutral measure. Each step reflects small changes in price over time, and by working backward from the terminal nodes—where option payoffs are known—the model calculates the present value of the option. Trinomial trees are widely used in pricing European, American, and exotic options. Their flexibility and improved precision make them particularly useful when modeling more complex financial instruments or capturing features like early exercise.

References:

[1] J. Ahn, M. Song, *Convergence of the trinomial tree method for pricing European/American options*， Applied Mathematics and Computation 189 (2007) 575–582

[2] J.C. Cox, S.A. Ross, M. Rubinstein, *Option pricing: a simple approach*, J. Financial Econ. 7 (1979) 229–263.