FE545 - Final Exam

Author: Sid Bhatia

Date: May 3th, 2024

Pledge: I pledge my honor that I have abided by the Stevens Honor System.

Professor: Steve Yang

Background

Random Tree Methods (RTMs)

Tree-based methods can be used for obtaining option prices, which are especially popular for pricing American options. Binomial (BAPM/BOPM), trinomial (TAPM/TOPM), and random tree methods (RTM) can be used to price many options, including plain vanilla options, and also exotic options such as barrier options, digital options, Asian options, and others.

We will be pricing American options using TAPM/RT methods via design principles in C++.

Pricing a derivative security entails calculating the **expected value of its payoff**. This reduces, in principle, to a problem of *numerical integration*; but in practice this calculation is often difficult for high-dimensional pricing problems.

The binomial options pricing model (BOPM) approach has been widely used since it is able to handle a variety of conditions for which other models cannot easily be applied. This mainly since the BOPM is based on the description of an underlying instrument over a period of time rather than a single point. Therefore, it can be used to price Americans that are exercisable $\forall\, t \in [0, T]$.

The TOPM, proposed by **Phelim Boyle** in '86, is (considered to be) more accurate than BOPM, producing the same results w/fewer steps.

Broadie and **Glasserman** ('97) proposed the simulated random tree to price Americans, deriving the upper/lower bounds; this combo makes it easier to measure & control error as computational effort increases $(m \to \infty)$. The main drawback of RTM is *computational requirements* grow exponentially with m, # of exercise dates, applicable only when $m < \text{big number}$. For $m = \text{small number}$, RTM works well & shows theme of **mananging scores** of **high/low bias**.

The typical simulation to Euro pricing is use sims to $\approx$ the expectation:

$$P = e^{-rT}\mathbb{E}^Q[f(S_T)] \tag{1}$$

where $f(S_T)$ is the payoff function at maturity T. For a call, $f(S_T) = (S_T - K)_+$. r denotes the risk-free rate, K is the strike, and S_τ is the **terminal stock price**.

For Americans, the dilemma is as follows:

$$P = \max_{\tau}\{e^{-r\tau}(S_\tau - K)_+\},\ \forall \tau \leq T \tag{2}$$

We discretize this problem where $\tau \in \mathcal{P}$, with $\mathcal{P} = \{t_0, t_1, \ldots, t_d\}$ such that $t_0 < t_1 < \cdots < t_d = T$. For an American option, we simulate a path of asset prices $(S_0, S_1, \ldots, S_T)$. Let $i \in \{1, \ldots, d\}$ correspond to intermediate times t_i in $\mathcal{P}$.

To calculate the **discounted value** of an option for a given simulation path, you first determine the payoff for each path at maturity and then average these results over many simulations. This process helps estimate the expected payoff under stochastic conditions. How do we compute the value along each path?

Broadie and **Glasserman** (1997) developed a stochastic method to estimate lower and upper bounds for American options. Let $\tilde{h}_i$ denote the **payoff function** for exercise at t_i, depending on i. Let $\tilde{V}_i(x)$ denote the value of the option at t_i given $X_i = x_i$ (assuming the option has **not been exercised**).

Why do we care? We have an interest in $\tilde{V}_0(X_0)$, which is recursively defined as follows:

$$\tilde{V}_m(x) = \tilde{h}_m(x) \tag{3}$$

$$\tilde{V}_{i-1}(x) = \max\{\tilde{h}_{i-1}(x),\, \mathbb{E}[D_{i-1,i}\tilde{V}_i(X_i) \mid X_{i-1} = x]\} \tag{4}$$

For each i from 1 to $m-1$, we introduce the notation $D_{i-1,i}$ for the discount factor from time t_{i-1} to t_i. This ensures that at the $(i-1)$-th exercise date, the option value (OV) is the maximum of the **immediate exercise value** and the **expected present value** of continuing. At expiration, the option value is given by the payoff function $\tilde{h}_m$.

As the name indicates, the random tree method (define RTM here) is based on simulating a *tree of paths* of the underlying **Markov chain** $\{X_0, X_1, \ldots, X_m\}$. Let's assume we fix a *branching parameter b* ≥ 2, where $b \in \mathbb{Z}^+$ (for simplicity).

From the initial state X_0, simulate b independent successor states $\{X_1^1, \ldots, X_1^b\}$, each following the same probability distribution (**law**) as X_1. From each X_1^i, simulate b independent successors $X_2^{i1}, \ldots, X_2^{ib}$, each following the **conditional law** of X_2 given $X_1 = X_1^i$.

From each $X_2^{i_1 i_2}$, generate b successors $X_3^{i_1 i_2 1}, \ldots, X_3^{i_1 i_2 b}$, and so on up to X_m.

We denote a **generic node** in the tree at time step i by $X_i^{j_1 j_2 \cdots j_i}$. This notation indicates that this node is reached by following the j_1-th branch out of X_0, the j_2-th branch out of the subsequent node, and so forth, reflecting the path taken through the branching tree.

At all terminal nodes of the tree, the estimator is set equal to the payoff at that node:

$$\hat{v}_m^{j_1 \cdots j_m} = h_m(X_m^{j_1 \cdots j_m}) \tag{5}$$

This means that at the final time step, the value of the option is **exactly equal** to its **payoff**, as there are no subsequent decisions to make.

At each node and at each branching step within the tree, the decision is calculated based on the **potential future values** versus the **immediate payoff**:

$$\hat{v}_{ik}^{j_1 j_2 \cdots j_i} = \begin{cases} h_i(X_i^{j_1 j_2 \cdots j_i}) & \text{if } \frac{1}{b}\sum_{j=1}^{b} \hat{v}_{i+1}^{j_1 j_2 \cdots j_i j} \leq h_i(X_i^{j_1 j_2 \cdots j_i}) \\ \hat{v}_{i+1}^{j_1 j_2 \cdots j_i k} & \text{otherwise} \end{cases} \tag{6}$$

This evaluates whether to exercise the option now or *continue* to the next step, based on expected values calculated from future possible states.

Next, the estimator for each node is computed as the average of the estimators from each possible path branching from this node:

$$\hat{v}_i^{j_1 \cdots j_i} = \frac{1}{b}\sum_{k=1}^{b} \hat{v}_{i,k}^{j_1 \cdots j_i} \tag{7}$$

This calculation effectively averages the outcomes from all *potential future states* accessible from the current node, **weighting each equally**. This average is crucial for determining the expected value of the option at each node, considering all possible future paths.

Working backward through the tree, the value at each node is determined by comparing the **immediate payoff** if the option were exercised at that node, versus the *expected value* if the option were not exercised but instead **continued**:

$$\hat{V}_i^{j_1 \cdots j_i} = \max\left(h_i(X_i^{j_1 \cdots j_i}), \frac{1}{b}\sum_{k=1}^{b} \hat{V}_{i,k}^{j_1 \cdots j_i}\right) \tag{8}$$

This equation takes the maximum of two values:

1. $h_i(X_i^{j_1 \cdots j_i})$: the payoff if the option is **exercised** at this node.
2. The *average* of the estimated values of continuing to the next time step (**first raw moment/expectation**), summed over all b branches leading out of the current node.

This decision rule ensures that the option is exercised at the time step and state that provides the maximum expected value, capturing the essence of the American option's flexibility to choose the optimal exercise time.

The **highest estimator** of the option price at the current time and state is represented by $\hat{v}_0$. This value is the result of the recursive calculation process applied from the terminal nodes back to the root of the decision tree, considering all possible decisions at each step.

The **low estimator** is also set equal to the payoff at the terminal nodes:

$$\hat{v}_m^{j_1 \cdots j_m} = h_m(X_m^{j_1 \cdots j_m}) \tag{9}$$

This ensures that at the final time step, the value of the option is directly its intrinsic value, given there are no subsequent choices.

At each non-terminal node, the estimator is determined by comparing the **immediate payoff** to a *modified average* of the outcomes from the other paths:

$$\hat{v}_{ik}^{j_1 j_2 \cdots j_i} = \begin{cases} h_i(X_i^{j_1 j_2 \cdots j_i}) & \text{if } \frac{1}{b-1}\sum_{j=1; j \neq k}^{b} \hat{v}_{i+1}^{j_1 j_2 \cdots j_i j} \leq h_i(X_i^{j_1 j_2 \cdots j_i}) \\ \hat{v}_{i+1}^{j_1 j_2 \cdots j_i k} & \text{otherwise} \end{cases} \tag{10}$$

Here, we exclude the **current branch's continuation value** from the averaging process to ensure a *conservative estimation*, minimizing the potential overestimation from positive outliers in the continuation path.

Finally, the low estimator for each node is calculated as the *average* (first raw moment/discrete expectation) of all branch estimators at that node:

$$\hat{v}_i^{j_1 \cdots j_i} = \frac{1}{b}\sum_{k=1}^{b} \hat{v}_{ik}^{j_1 \cdots j_i} \tag{11}$$

This averages the adjusted estimators, incorporating a balanced view from all potential paths stemming from the node, using the conservative estimates derived from the modified decision rule.

Consider the random tree depicted in *Figure (1)*, which is used for pricing an **American put option**. Utilizing this tree, please compute both the **high** and **low** estimates for the price of an **American call option**. Be sure to outline your calculation steps for each estimator.

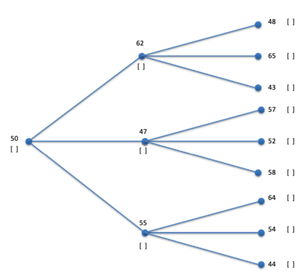

Figure 1: Trinomial Tree

Pricing

Now, we begin by pricing a standard American call option on a **single asset**, which pays *continuous dividends* and whose price is governed by a **geometric Brownian motion** (**GBM**) process. We assume that the risk-neutralized price of the underlying asset S_t satisfies the following **stochastic differential equation** (**SDE**):

$$\frac{dS_t}{S_t} = (r - \delta)dt + \sigma dW_t \tag{12}$$

where $W_t \sim \mathcal{N}(0, t)$.