Using Monte Carlo to Value Derivatives With Early Exercise

Craig Pirrong
Bauer College of Business
University of Houston

April 8, 2020

It is possible, but cumbersome, to implement Monte Carlo methods to take into account early exercise. This involves using the same logic as in finite difference methods: start at expiration, work backwards in time towards the present (i.e., when you are valuing the option), and at exercise dates evaluate whether early exercise is optimal. This is essentially stochastic dynamic programming.

It is obviously not feasible to do this exactly for an American option–that would require evaluating early exercise at an infinite number of dates. You can do so by evaluating at a large (but finite) number of dates. This tends to be computationally costly using the method I will describe. Thus, this method is best suited for Bermudan options which allow exercise at a finite number of dates prior to expiration.

The basic idea behind the algorithm is that the early exercise decision involves comparing the proceeds from exercise (e.g., $\max[K-S,0]$ for a put), to the value of holding onto the option. This last is sometimes referred to as the "continuation value." It is basically the expected value of the option at the next exercise period. So how do you estimate that holding value?

The first step of the process is to draw random price paths, where you simulate the price at each possible exercise date. Assume there are N_E such dates τ_i, where τ_{N_E} is the expiration date. Further, assume you have N_S simulated paths, and denote the price on path j at time i S_{ij}.

At τ_{N_E} you know the payoff to the option on every possible path. Denote the payoff on path j as Π_j. Now proceed to τ_{N_E-1}. You want to determine the present value of the expected payoff for each path, $E(\Pi_j | S_{\tau_{N_E-1},j})$. There are a variety of ways of estimating that expectation, but the most common is to run a regression:

$$\Pi_j = \beta_0 + \sum_{k=1}^{P} \beta_k S^k_{\tau_{N_E-1},j}$$

That is, you regress the payoff on each path against a polynomial of the price at τ_{N_E-1}, where P is the order of the polynomial. This is usually something like $P = 5$ or $P = 6$. (Note: In Matlab, you will usually get an error like `Warning: Matrix is close to singular or badly scaled. Results may be inaccurate.` You can ignore.)

Call the fitted value at stock price path j at τ_{N_E-1} $\hat{C}_j^{\tau_{N_E-1}}(S_{\tau_{N_E-1},j})$: this fitted value is the estimate of the expectation. The continuation (holding value) at τ_{N_E-1} is therefore:

$$C_j^{\tau_{N_E-1}}(S_{\tau_{N_E-1},j}) = max[K - S_{\tau_{N_E-1},j}, 0, e^{-r(\tau_{N_E}-\tau_{N_E-1})}\hat{C}_j^{\tau_{N_E-1}}(S_{\tau_{N_E-1},j})]$$

That is, the continuation value is the maximum of the proceeds from exercise, or the present value of the expected value of the option next period.

Proceed backwards through the exercise dates, and perform a similar analysis. Consider exercise date m. First, run the following polynomial regression:

$$C_j^{\tau^{m+1}} = \beta_0^m + \sum_{k=1}^{P} \beta_k^m S^k_{\tau_m,j}$$

Here I've superscripted the β coefficients to emphasize that you will get different coefficients for each exercise date. Then call the fitted value $\hat{C}_j^{\tau^{m+1}}(S_{\tau_m,j})$. Now, the continuation value is:

$$C_j^{\tau_m} = \max[K - S_{\tau_m,j}, 0, e^{-r(\tau_{m+1}-\tau_m)}\hat{C}_j^{\tau^{m+1}}(S_{\tau_m,j})]$$

Rinse, wash, repeat. That is, perform this at every exercise time.

To determine the value of the option, take the average of the continuation values at the first exercise date, and discount them back one period:

$$V = e^{-r\tau_1}\frac{\sum_{j=1}^{N_S} C_j^1}{N_S}$$