ORF 522: Lecture 21

Pricing American Options

Robert J. Vanderbei

December 10, 2013

Slides last edited at 7:40am on Thursday 12th December, 2013

Operations Research and Financial Engineering, Princeton University
http://www.princeton.edu/∼rvdb

American Options

A *Perpetual American Option* is a legal contract giving the holder of the option the right to buy a particular stock at a particular price, say K, at any time in the future.

Let X_n denote the share price of the stock at time n (i.e., n days in the future).

Obviously, X_n only becomes known at time n. Prior to time n it is a *random variable*.

To keep things as simple as possible, we assume that at the end of each day the share price can do only one of two things:

- Go up a little bit with probability p, or
- Go down by the same amount with probability $q = 1 - p$.

So, if the price today is X_0, the price tomorrow will be either $X_0 + \Delta x$ or $X_0 - \Delta x$, where Δx is a small positive number, say, for example, $\Delta x = 0.1$ (dollars per share of stock).

State Space

Because our simple model allows only two choices for how the share price changes from one day to the next, it follows that after n days there are only a fixed (finite) collection of possible share prices.

To see it, suppose that up to day n there has been j up days and k down days. Then, clearly, the share price on day n is

$$X_n = X_0 + (j-k)\Delta x.$$

We assume that the initial stock price X_0 is itself a multiple of Δx. We also assume that if the stock price hits zero, then it stays there forever.

Hence, the set of *states* that the share price can be "in" is simply

$$0,\ \Delta x,\ 2\Delta x,\ \ldots,\ X_0 - 2\Delta x,\ X_0 - \Delta x,\ X_0,\ X_0 + \Delta x,\ X_0 + 2\Delta x,\ \ldots$$

Payoff

Suppose that, on day n, the holder of the option decides to dispose of the option.

If the share price X_n is greater than the strike price K, then the holder of the option can buy the stock for K dollars and immediately sell it for X_n dollars and realize a gain of $X_n - K$ dollars.

If, on the other hand, the share price is less than the strike price, the option holder would lose money if he/she were to exercise the option.

Hence, the value of the option, if disposed of on day n, is

$$f(X_n) = \begin{cases} X_n - K & \text{if } X_n \geq K \\ 0 & \text{if } X_n \leq K \end{cases}$$

Expected Payoff

We suppose the option holder chooses to dispose of the option at some future time τ, the choice of which can depend on the evolution of the share price between now and then. That is, on the first day $(n = 0)$, τ must be modeled as a random variable.

At $n = 0$, we want to know the *expected value* of the payoff:

$$\mathbf{E}\alpha^{\tau} f(X_{\tau}).$$

Here, since the option is perpetual, it is important to introduce a discount rate α. It is a number slightly less than one. It represents today's value of tomorrow's dollar.

The optimal strategy is determined by maximizing over all (non-clairvoyant) random times τ:

$$v(x) = \max_{\tau} \mathbf{E}\left(\alpha^{\tau} f(X_{\tau}) \mid X_0 = x\right).$$

Principle of Dynamic Programming

Suppose the market has been evolving for awhile and we still have the option. Suppose the current share price is x.

Let's consider our choices. We can either decide to exercise the option or hold it until tomorrow at which time the same question will be asked again.

If we exercise the option, then we get $f(x)$ dollars.

If we hold until tomorrow (or beyond) and assume that we will behave optimally from then on, then the share price either goes up by Δx or down by Δx with equal probability. Hence, the value of this choice is $pv(x + \Delta x) + qv(x - \Delta x)$. But, that's in tomorrow's dollars. To convert to today's dollars, we have to multiply by the discount factor α.

Obviously we should pick the better of the two options:

$$v(x) = \max\Big(f(x),\ \alpha\left(pv(x + \Delta x) + qv(x - \Delta x)\right)\Big).$$

Converting to an LP Problem

One can show that $v(x)$ is the smallest function that satisfies:

$$\begin{aligned} v(x) &\geq f(x), \\ v(x) &\geq \alpha \left(pv(x+\Delta x) + qv(x-\Delta x) \right). \end{aligned}$$

Since the set of possible x's (the so-called *state space*) is discrete, it is convenient to introduce the following notation:

$$\begin{aligned} x_j &= j\Delta x, & j &= 0, 1, 2, \ldots, \\ v_j &= v(x_j), & j &= 0, 1, 2, \ldots, \\ f_j &= f(x_j) & j &= 0, 1, 2, \ldots. \end{aligned}$$

If we arbitrarily truncate that set, then we get the following linear programming problem:

$$\begin{array}{lll} \text{minimize} & \displaystyle\sum_{j=0}^{\infty} v_j & \\ \text{subject to} & v_j \geq f_j & j = 0, 1, 2, \ldots \\ & v_j \geq \alpha \left(pv_{j+1} + qv_{j-1} \right) & j = \quad 1, 2, \ldots. \end{array}$$

AMPL Model

```
param K := 9;
param dx := 0.1;
param p := 0.51;
param q := 1-p;
param alpha := 0.999;

param N := 150;

set X  := setof {j in 0..N} j*dx ordered;

param f{x in X} := max(x-K,0);

var v{X} >= 0;

minimize sumvx: sum {x in X} v[x];

subject to exerciseLater {x in X: x != first(X) && x != last(X)}:
        v[x] >= alpha*(p*v[next(x)] + q*v[prev(x)]);

subject to exerciseNow {x in X}: v[x] >= f[x];

solve;

printf {x in X}: "%8.3f %8.3f %8.3f \n", x, f[x], v[x] > "xfv";
```

Output