

Math 456: Introduction to Financial Mathematics

Homework Set 4

1. Implement a compound Put option, the option to purchase a European-style Put option, with expiry T and strike price K , for price L at time T_1 satisfying $0 < T_1 < T$. Use your code to price such an option with parameters $(T, T_1, S_0, K, L, r, v, N) = (1, 0.4, 97, 99, 1.50, 0.04, 0.23, 20)$. (Hint: modify `CRRcc.m`.)
2. Let $\mathcal{M}(n)$ be the set of path numbers in a recombining binomial tree to depth n , as defined in Eq.4.6. Prove that

$$\mathcal{M}(n+1) = [2\mathcal{M}(n)] \cup [2\mathcal{M}(n) + 1],$$

where $aX + b \stackrel{\text{def}}{=} \{ax + b : x \in X\}$ for sets X of numbers.

3. Implement floating strike option pricing in the CRR model using geometric means instead of arithmetic means, as in `CRRgro` versus `CRRaro`. Compare the results on the inputs $(T, S_0, r, v, N) = (1, 97, 0.04, 0.25, 20)$.
4. Use `CRRaro` to compute the average-rate Call and Put premiums in the CRR model for $S_0 = K = 420$, $T = 2$, $r = 0.035$, $v = 0.33$, and different values of N such as $N = 5, 10, 15, 20$. Compare $\overline{C}(0) - \overline{P}(0)$ with the limit value in Eq.4.20.
5. Implement floating strike option pricing in the CRR model using path-dependent Arrow-Debreu securities. Check that the results agree with `CRRflt` on that function's suggested example inputs.
6. Write an Octave program to compute the maximums along all paths in a non-recombining binary tree of depth N . (Hint: modify `NRTmin()`.) Run both programs on the inputs `N=3` and

$$\text{Sbar} = [2 \ 1 \ 3 \ 4 \ 1 \ 0 \ 4 \ 5 \ 3 \ 1 \ 2 \ 4 \ 6 \ 1 \ 2]$$

Compare the results to confirm that path maxima are at least as large as path minima.

7. Implement lookback option pricing in the CRR model using path-dependent Arrow-Debreu securities. Check that the results agree with `CRRlb` on the inputs $(T, S_0, r, v, N) = (1.5, 213, 0.04, 0.21, 10)$.

8. Implement ladder Put option pricing in the CRR model by modifying `CRRladC` appropriately. Check that the premium is at least as great as that for the vanilla European-style Put with the same parameters, using the example inputs suggested within `CRRladC`.