Practical Optimization in Finance, Fall 2001

"Aftermath" of Lectures on Wednesday October 24, 2001

The course: The course will not be canceled. Seems not to be able to lecture for the next month or so, therefore I'll be giving a talk on this period (whose length is not exactly known). Referring to the course name, 1 will focus considerably more on the terms "finance" and "practical" than on "optimization". Two key-phrases are:

- Computational techniques for option pricing.
- Calibration of financial models.

The projects: Project 1 will be handed back on October 31. Project 2 has now been posted on the homepage. It deals with pricing in binomial models, and is not intended to be particularly hard. The "hand-in" deadline is November 15. (The original deadline for Project 2 was November 8, and I still think you should be able to make that, too. Anyway: November 15 is.) The final project is supposed to be "something you choose yourself & a bit longer than the others", and is to be presented on December 5. More info to follow on that. We will try to stick to this time-plan in order not to come in conflict with other exams. (I wish I could schedule exams in December here at KU.)

We will make a list of suggested projects soon. The list will include projects from both Soren's and my topics.

The literature: My lectures will be based on notes (some of which may not exist yet) and working papers that, if it is at all possible, I'll post on the course homepage. In other words you don't have to go out and buy a new book. If you have a stringy urge to do so anyway, I can recommend John Hull's book "Options, Futures, and Other Derivatives". You should be able to find its 4th edition (the newest I know of) for around £35 or so on the www.

The software: Soren is a GAMS buff. I know nothing about GAMS. To do my project(s) you will have some kind of program that can perform basic calculations & allows "black programming" (such as looping). Besides that, it can be anything: Excel, Maple, Matlab, Mathematica, Pascal, C, SAS, Splus (a statistics program you may not have heard of, but one of my favorites)...

GAMS probably works too, I just have no idea how.

(And now finally) The Lectures October 24: I talked about (call) option pricing in binomial models; something along the lines of Section 2 in RMP and Chapter 4 in LP (even abbreviations of notes posted on the homepage).

The bottom line was: A call option has only one possible price; any other price would create an arbitrage possibility, another word for a money machine or "a free lunch". And from an economic/financial point of view it is a very mild equilibrium requirement that prices are such that arbitrage is not possible. We also saw that the arbitrage-free price could be explicitly calculated by working backwards through the stock-price tree or lattice. At each node the call-price was the (risk-free rate) discounted expected value of the call-price at the two successive nodes, where the expectation was calculated with an adjusted probability \( q = \frac{1}{2} \) (where it has been naturally suppressed that \( u, d \) and even \( r \) are in fact allowed to vary across nodes). We also noted that the original probability \( p \) played no role in the pricing formula. At each node we could also easily find the number of stocks in the replicating portfolio - the existence of a replicating portfolio was the whole reason we could be sure the call option could only trade at one possible price - as

\[
q_{\text{node}} = \frac{\text{call-price (successive up-node)} - \text{call-price (successive down-node)}}{\text{stock-price (successive up-node)} - \text{stock-price (successive down-node)}}
\]

where the last equality is mostly a definition of notation, but quite a sensible one. Finally, we looked at a numerical 2-period example, namely the one given in Figure 1.1 in RMP. This model has \( T = 1/2 \) (the 3 time-points involved are then \( 0, T/4 \) and \( T/2 \)) and a 1-period risk-free rate of 0.0125, i.e. 1.25% (something that was not entirely clear to me at the lectures; that's what happens when you try 'ad-lib' numerics). Further, the replicating strategy was found:

- At time 0 (call this node C)
  \( a_C = \frac{(6.24 - 0)}{(107.79 - 92.77)} = 0.4154 \). The replicating strategy then has \( b_C = 3.48 - 100 \times 0.4154 = 836.96 \) in the bank (i.e. borrowed money; assume that units of money are called 8).

- At time \( T/4 \), up-node (say, A)
  \( a_A = 0.6910 \) and \( b_A = 6.24 - 107.79 \times 0.6910 = 98.24 \).

- At time \( T/4 \), down-node (say, B):
  \( a_B = 0 \) and \( b_B = 92.77 \times 0 = 0 \).

1
But what do these numbers mean? They tell us what the no-arbitrage price of the call is ($3.48) and how to make arbitrage if somebody is willing to trade (buy & sell) at a different price. Let’s walk through an example: Suppose someone is willing to buy the call-option for $84. We then do the following: Sell a call-option to him (thus receiving $84), borrow $38.06 in the bank, and buy \( a_c = 0.4154 \) units of the stock. All in all you have received \( 4 + 38.06 - 100 = 0.4154 - 80.52. \) Put that in your pocket & wait. At time \( T/4, \) your action depends on what has happened. At node \( A \) (i.e., if the stock went up) you buy \( 0.6910 - 0.4154 \) more units of stock: thus taking you total position in stock to \( 0.6910 = a_A. \) This costs \( 0.6910 - 0.4154 \times 107.79 = 829.71. \) That money you borrow at the bank, meaning that you now owe the bank 38.06 * (1.0125) + 29.71 = 38.24 (\{0.4\}). At node \( B \) you sell the stock you have, thus receiving 92.77 * 0.4154 = 38.54; with this money you can exactly pay off the 38.06 = 1.0125 = 38.54 8 that you owe the bank. So you do that, if the stock goes up from node \( A, \) you have to pay $11.18 to the person you sold the call-option to. Further you now owe 68.24 * (1.0125) = 69.09 the bank liabilities in all of $80.27. But you have 0.6910 units of stock. These are worth 0.6910 * 136.18 = 86.28, which (up to rounding errors) exactly covers what you have to pay. If the stock goes down from node \( A, \) the option finishes worthless (so you don’t have to pay money to the holder). You still owe 69.09, but that you can exactly cover by selling you units of stock, because this rises 0.6910 * 100 = 69.09. From node \( B, \) it doesn’t matter what happens. The option finishes worthless, you don’t own any stock, and your bank-account stands at 0. So once the smoke clears, the net result for you is the $80.52 that you pocketed at time 0; all other transactions net out. You have made “free money”.

**Wednesday October 31:** Hand-back of Project 1. More on (binomial) option pricing. If you weren’t at the lectures on October 24 (and judging from the number of people present, there is a good chance that you weren’t), you should look closely at the notes on the homepage. I’ll talk about how to build a reasonable lattice, i.e., one that goes well with observed market behaviour. And I’ll illustrate a connection to optimization & portfolio choice questions. In the “computer-lab”-time, you should start working on Project 2.

Kindly,

Rolf