Lab Assignments Computational Finance

Lecturer: Dr Drona Kandhai
Office: F 2.02
Address: Section Computational Science, Kruislaan 403, 1098 SJ, Amsterdam
Email: B.D.Kandhai@uva.nl

These assignments can be done in groups of two students. Reports with a *clear description of the assignment, the methods, the results and discussion* should be submitted *before the deadlines.* You are free to choose the programming language/environment in which you would like to write your computer programs. If you have questions about the assignments do not hesitate to contact the teaching assistant or the lecturer.

Assignment 3: Finite-Difference Methods for Option Pricing

A. Background

The Black-Scholes Partial Differential Equation (BS-PDE) derived for a plain vanilla option has the following form

$$\frac{\partial V}{\partial t} + rS\frac{\partial V}{\partial S} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} = rV \quad (1)$$

The equation can be transformed into a PDE with constant coefficients, by introducing $X = \ln S$. Since the equation is commonly solved backward in time, it is for further numerical treatment convenient to introduce the following transformation

$$\frac{\partial V}{\partial t} = -\frac{\partial V}{\partial \tau}$$

The transformed BS-PDE has the following form:

$$\frac{\partial V}{\partial \tau} = \left(r - \frac{1}{2}\sigma^2\right)\frac{\partial V}{\partial X} + \frac{1}{2}\sigma^2 \frac{\partial^2 V}{\partial X^2} - rV \quad (2)$$

The explicit FTCS (Forward Time Centred Scheme) for the transformed BS-PDE (2) can be derived by means of Taylor Expansion Techniques. Recalling that

$$V(X + \Delta X, \tau) = V(X, \tau) + \Delta X \frac{\partial V(X,\tau)}{\partial X} + \frac{1}{2!}\Delta X^2 \frac{\partial^2 V(X,\tau)}{\partial X^2} + \cdots$$

it can be shown hat the Finite-Difference (FD) scheme becomes

$$V_i^{n+1} = V_i^n + \left(r - \frac{1}{2}\sigma^2\right)\frac{\Delta\tau}{2\Delta X}\left(V_{i+1}^n - V_{i-1}^n\right) + \frac{1}{2}\sigma^2 \frac{\Delta\tau}{\Delta X^2}\left(V_{i+1}^n - 2V_i^n + V_{i-1}^n\right) - r\Delta\tau V_i^n \quad (3)$$

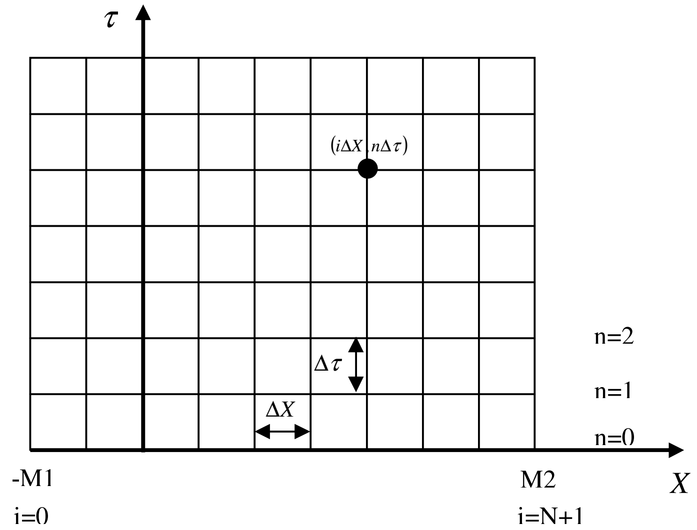
where the superscript n denotes the time level and subscript the spatial index of the underlying log-
value X

Similarly, it can be shown that the Crank-Nicolson scheme for equation (2) is given by

$$\begin{aligned} V_i^{n+1} = V_i^n &+ \left(r - \frac{1}{2}\sigma^2\right)\frac{\Delta\tau}{4\Delta X}\left(V_{i+1}^{n+1} - V_{i-1}^{n+1} + V_{i+1}^n - V_{i-1}^n\right) + \\ &\frac{1}{4}\sigma^2 \frac{\Delta\tau}{\Delta X^2}\left(V_{i+1}^{n+1} - 2V_i^{n+1} + V_{i-1}^{n+1} + V_{i+1}^n - 2V_i^n + V_{i-1}^n\right) - r\Delta\tau\left(V_i^{n+1} + V_i^n\right) \end{aligned} \quad (4)$$

B. FD-Schemes for European call

The FD-mesh for the derived schemes is drawn below. It should be noted that the cell-width is uniform over the whole computational domain. The infinite extent of $X = \ln S$ in the continuous problem is approximated by the truncated interval $[-M1, M2]$, where $M1$ and $M2$ are sufficiently large chosen positive constants so that the boundary conditions at the two ends of the infinite interval can be applied with sufficient accuracy. Approximated option values are computed at grid points $(i\Delta X, n\Delta\tau)$, i=1,2,…,N and n=0,1,… The option values along the boundaries i=0 and i=N+1 are prescribed by boundary conditions of the option model, while initial values along n=0 are given by the terminal payoff function.



1. Rewrite the FTCS-scheme (3) and Crank-Nicolson scheme (4) in the form

$$a_1 V_{i+1}^{n+1} + a_0 V_i^{n+1} + a_{-1} V_{i-1}^{n+1} = b_1 V_{i+1}^{n} + b_0 V_i^{n} + b_{-1} V_{i-1}^{n} \qquad \forall i = 1,2,\cdots,N \quad n = 0,1,\cdots \tag{5}$$

   above form (5) describes the general class of two-level (in time), six-point scheme (in space). This form allows to write above equation (5) in matrix form.

$$\begin{pmatrix} a_0 & a_1 & 0 & \cdots & \cdots & 0 \\ a_{-1} & a_0 & a_1 & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \ddots & & \\ & & \ddots & \ddots & \ddots & \\ 0 & \cdots & \cdots & 0 & a_{-1} & a_0 \end{pmatrix} \begin{pmatrix} V_1^{n+1} \\ V_2^{n+1} \\ \vdots \\ \vdots \\ V_N^{n+1} \end{pmatrix} = \begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ \vdots \\ c_N \end{pmatrix} \tag{6}$$

   where,

$$
\begin{aligned}
c_1 &= b_1 V_2^n + b_0 V_1^n + b_{-1} V_0^n - a_{-1} V_0^{n+1} \\
c_N &= b_1 V_{N+1}^n + b_0 V_N^n + b_{-1} V_{N-1}^n - a_1 V_{N+1}^{n+1} \\
c_i &= b_1 V_{i+1}^n + b_0 V_i^n + b_{-1} V_{i-1}^n, \qquad \forall i = 2,\cdots,N-1
\end{aligned}
\tag{7}
$$

2. Solve the above tri-diagonal system by writing a computer program in VBA/C/C++/Fortran or even MatLab. Do this for both the FTCS-scheme and the Crank-Nicolson scheme. What are the boundary conditions? Use the following market parameters for a European call

   r=4%, vol = 30%, S0 = 100, K=110, T=1 year (in the money)
   r=4%, vol = 30%, S0 = 110, K=110, T=1 year (at the money)
   r=4%, vol = 30%, S0 = 120, K=110, T=1 year (out of the money)

   Check whether your computer program is inline with your expectations from the theoretical analysis of exercise A.

   Determine your own optimal mesh size $(\Delta\tau, \Delta X)$. Plot for all the test cases the delta as function of the underlying S, i.e.

   $$\Delta = \frac{\partial V}{\partial S} \quad (8)$$

   Note (!) that the Greeks are in the original variable S and not in X. Explain the graphs.

Recommended Literature:

- Seydel, R. U. (2004). “Tools for Computational Finance” (Chapter 4), Springer-Verlag Softcover, ISBN: 3-540-40604-2.
- Brennan, M. J. and Schwartz, E. S. (1978). “Finite Difference Methods and Jump Processes Arising in the Pricing of Contingent Claims: A Synthesis.” *Journal of Financial and Quantitative Analysis*, pp. **461- 474**
- Domingo Tavella and Curt Randall (2000), “Pricing Financial Instruments: The Finite Difference Method (Chapter 5)”, John-Wiley and sons.