**Generalization of Finite-Difference Schemes**

Take a grid node which corresponds to . Assume we are pricing call options. The option value in the node will be denoted by .

The partial derivative of with respect to time in node can be approximated as follows:

For other derivatives, such as , we will take a linear combination of finite-difference approximations at two neighboring nodes to :

Also, for the values of we will use this approximation: .

The Finite Difference approximation of the Black-Scholes PDE,

will result in the following numerical scheme:

By combining/grouping similar terms, we can rewrite it as follows:

Denote the coefficients of the 6 terms above as follows:

**Note**: The coefficients are all -dependent.

Then, we can derive the following numerical scheme:

The latter scheme can be rewritten as

Solving this equation backwards in time will result in option prices at time , given that the option prices are known at time .

Denote by the right side of the above equation. That is.

Then,

For and .

Using option values in 3 nodes at time will help us find option values at 3 nodes at time .

By combining the above equations in a matrix equation, we can solve for the vector of option prices, as was done in the three cases discussed earlier.

where

The goal here is to find .

The above matrix equation can easily be solved to estimate .

**Remark:** We did not include the extreme values of *C* in the calculations above. That is, the values of and that correspond to option values for “very large” and “very small” stock prices, respectively, were not included in the matrix equation above.

* For call options, we can use and for .
* For put options, we can use for .

Special Cases:

1. . This will be the **explicit** F.D. method.
2. . This is an **implicit** F.D. method.
3. This is the **fully implicit** F.D. method.
4. This is the **Crank-Nicolson** F.D. method.