**Numerical Methods in Option Pricing**  
*(Part III)*

E. **Explicit Finite Differences**

1. **Use of the Forward, Central, and Symmetric Central**

   a. In order to obtain an explicit solution for the price of the derivative, we need to use the following combination of approximations: forward difference for $\frac{\partial f}{\partial t}$, central difference for $\frac{\partial f}{\partial S}$, and symmetric central for $\frac{\partial^2 f}{\partial S^2}$

   The approximation to the PDE
   \[
   -\frac{\partial f}{\partial t} + a \frac{\partial f}{\partial S} + b \frac{\partial^2 f}{\partial S^2} + h - rf = 0
   \]
   
   thus becomes:
   \[
   -\frac{f(S,t+\Delta t) - f(S,t)}{\Delta t} + a \frac{f(S+\Delta S,t) - f(S-\Delta S,t)}{2\Delta S} + b \frac{f(S+\Delta S,t) - 2f(S,t) + f(S-\Delta S,t)}{(\Delta S)^2} + h - rf = 0
   \]

   b. We then evaluate the approximated PDE for the different values on the grid, starting from $S=n\Delta S$ and $t=m\Delta t$: 

\[-\frac{f_{n+1}^m - f_n^m}{\Delta t} + a_n^m \frac{f_{n+1}^m - f_{n-1}^m}{2\Delta S} + b_n^m \frac{f_{n+1}^m - 2f_n^m + f_{n-1}^m}{(\Delta S)^2} + h_n^m - r_{n+1}^m f_{n+1}^m = 0\]

- The resulting equation is called a “difference equation”.

- The difference equation approximates the PDE as $\Delta t$ and $\Delta S$ go to 0.

c. The difference equation gives us EXPLICITLY the value of $f$ for stock level $n$ and time $m+1$, thus it gives us $f_{n+1}^m$ once we have computed the values at time $m$:

\[
(1 + r_{n+1}^m \Delta t)f_n^m = f_n^m + \left(\frac{\Delta t}{(\Delta S)^2}\right) b_n^m \left(f_{n+1}^m - 2f_n^m + f_{n-1}^m\right) + \left(\frac{\Delta t}{\Delta S}\right) a_n^m \left(f_{n+1}^m - f_{n-1}^m\right) + (\Delta t) h_n^m
\]

d. Notice that time $m$ is actually temporally AFTER time $m+1$, since the time represents time left to maturity. The initial condition (hence the value of the derivative at maturity, when $m=0$) is: $f_n^0 = f_0(n\Delta S,0)$

e. For stock levels that reach the boundaries, use the following boundary conditions:

- At $n+1=N$: $f_{N}^m = \overline{f}(N\Delta S,m\Delta t)$
- At $n-1=N$: $f_{N}^m = \underline{f}(N\Delta S,m\Delta t)$

(What are these conditions for puts and calls?)
2. For the simple basic diffusion equation \( \frac{\partial f}{\partial t} - \frac{\partial^2 f}{\partial S^2} = 0 \), the solution obtained \( f^n_m \) is stable only if \( 0 < \Delta t/(\Delta S)^2 \leq 1/2 \).

a. The issue is that the use of difference equations as approximations to the PDE introduces rounding errors.

b. We obtain a stable solution only if these rounding errors are not getting magnified at each iteration.

c. For the simple diffusion case, all we need to make sure of is that we choose the increments for \( S \) and \( t \) so that \( 0 < \Delta t/(\Delta S)^2 \leq 1/2 \).

d. This is a major problem for more complicated cases, however, as the stability condition that needs to be satisfied is not known in advance.

e. Keeping \( \Delta S \) proportional to \( \sqrt{\Delta t} \) as they converge is a good general rule, however. Ex: let \( \Delta t/(\Delta S)^2 = \alpha \)

<table>
<thead>
<tr>
<th>( S )</th>
<th>( \alpha = 0.25 )</th>
<th>( \alpha = 0.50 )</th>
<th>( \alpha = 0.52 )</th>
<th>exact</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.00</td>
<td>9.7531</td>
<td>9.7531</td>
<td>9.7531</td>
<td>9.7531</td>
</tr>
<tr>
<td>2.00</td>
<td>7.7531</td>
<td>7.7531</td>
<td>7.7531</td>
<td>7.7531</td>
</tr>
<tr>
<td>4.00</td>
<td>5.7531</td>
<td>5.7531</td>
<td>5.7531</td>
<td>5.7531</td>
</tr>
<tr>
<td>6.00</td>
<td>3.7532</td>
<td>3.7532</td>
<td>2.9498</td>
<td>3.7532</td>
</tr>
<tr>
<td>7.00</td>
<td>2.7567</td>
<td>2.7567</td>
<td>1.4192</td>
<td>2.7568</td>
</tr>
<tr>
<td>8.00</td>
<td>1.7986</td>
<td>1.7985</td>
<td>1.7985</td>
<td>1.7987</td>
</tr>
<tr>
<td>9.00</td>
<td>0.9879</td>
<td>0.9879</td>
<td>330.5603</td>
<td>0.9880</td>
</tr>
<tr>
<td>10.00</td>
<td>0.4418</td>
<td>0.4419</td>
<td>625.0347</td>
<td>0.4420</td>
</tr>
<tr>
<td>11.00</td>
<td>0.1605</td>
<td>0.1607</td>
<td>457.3122</td>
<td>0.1606</td>
</tr>
<tr>
<td>12.00</td>
<td>0.0483</td>
<td>0.0483</td>
<td>208.9135</td>
<td>0.0483</td>
</tr>
<tr>
<td>13.00</td>
<td>0.0124</td>
<td>0.0123</td>
<td>40.5813</td>
<td>0.0124</td>
</tr>
<tr>
<td>14.00</td>
<td>0.0028</td>
<td>0.0027</td>
<td>15.2150</td>
<td>0.0028</td>
</tr>
<tr>
<td>15.00</td>
<td>0.0006</td>
<td>0.0005</td>
<td>3.1682</td>
<td>0.0006</td>
</tr>
<tr>
<td>16.00</td>
<td>0.0001</td>
<td>0.0001</td>
<td>0.7365</td>
<td>0.0001</td>
</tr>
</tbody>
</table>

Comparison of exact Black-Scholes solution and explicit finite-difference solutions for a European put with \( E = 10, r = 0.05, \sigma = 0.20 \) and with six months to expiry. Note the effect of taking \( \alpha > \frac{1}{2} \).

(WDH 1993)
3. Probabilistic Interpretation

a. Simplifying by setting $h=0$, the equation can be rearranged as:

$$f_n^{m+1} = \left[ \frac{1}{1 + r^{m+1} \Delta t} \right] \left[ b_n^m + \left( \frac{\Delta t}{(\Delta S)^2} \right) b_n^m + \left( \frac{\Delta t}{2\Delta S} \right) a_n^m \right] f_{n+1}^m$$

$$+ \left[ 1 - \left( \frac{2\Delta t}{(\Delta S)^2} \right) b_n^m \right] f_n^m + \left[ \left( \frac{\Delta t}{(\Delta S)^2} \right) b_n^m - \left( \frac{\Delta t}{2\Delta S} \right) a_n^m \right] f_{n-1}^m$$

b. The terms in front of $f_{n+1}^m$, $f_n^m$ and $f_{n-1}^m$ can be viewed as the risk-neutral probabilities at point $(n,m)$ on the grid:

- We have three possible moves (Trinomial lattice)
- $f_n^{m+1}$ is given by the discounted next period expected value. Remember that time $m+1$ is actually an instant before time $m$, since we transformed $t$ into $T-t$. 

4. Black-Scholes Example

a. In order to use the explicit finite-difference method on the Black-Scholes PDE, we need to reduce the PDE to the simpler form mentioned earlier (through changes of variables) so that we know what the stability condition is. Otherwise, applying the methodology to the “raw” Black-Scholes PDE directly will most likely lead to an unstable system (numbers blowing up) since we do not know the stability condition for that case.

• Let \( x \) and \( \tau \) be defined as: \( x = \ln(S/K) \) and \( \tau = (T-t)(\sigma^2/2) \)

• Let \( C(S,t) \) be the price of a European Call option. Now let \( U(x,\tau) = C(S,t)/K. \)

\( C \) satisfies

\[
\frac{\partial C}{\partial t} + \frac{\partial C}{\partial S}(r-\delta)S + \frac{1}{2} \frac{\partial^2 C}{\partial S^2} \sigma^2 S^2 - rC = 0
\]

with Terminal condition \( C(S,T) = \max(0,S-K) \). Note that we also know that \( C(0,t) = 0 \) and that \( C(S,t) \) approaches \( S-K \) as \( S \) goes to \( \infty \).

• Substitute in the new variables and obtain:

\[
\frac{\partial U}{\partial \tau} = \frac{\partial U}{\partial x} \{ \kappa_2 - 1 \} + \frac{\partial^2 U}{\partial x^2} - \kappa_1 U
\]

where \( \kappa_1 = r/(\sigma^2/2) \) and \( \kappa_2 = (r-\delta)/(\sigma^2/2) \), and

\( U(x,0) = \max(0, e^x - 1) \).
• Note that this equation only contains two parameters \((k_1\) and \(k_2\)) instead of four \((K, T, r, \sigma, \delta)\)

• Perform one additional change of variable:
  let \(U(x, \tau) \equiv e^{\alpha x + \beta \tau} V(x, \tau)\)

• The equation for \(V\) thus is:

\[
\frac{\partial V}{\partial \tau} + \beta V = \alpha^2 V + 2\alpha \frac{\partial V}{\partial x} + \frac{\partial^2 V}{\partial x^2} + \left\{ \alpha \frac{\partial V}{\partial x} \right\} \{k_2 - 1\} - \kappa_1 V
\]

• Choose \(\alpha\) and \(\beta\) so that the equation is simplified as much as possible, thus choose:

\[
\begin{align*}
\beta &= \alpha^2 + (k_2 - 1) \alpha - \kappa_1 \\
2\alpha + (k_2 - 1) &= 0
\end{align*}
\]

that is:

\[
\begin{align*}
\beta &= -\kappa_1 - (k_2 - 1)^2 / 4 \\
\alpha &= - (k_2 - 1) / 2
\end{align*}
\]

The equation for \(V\) now becomes:

\[
\frac{\partial V}{\partial \tau} = \frac{\partial^2 V}{\partial x^2} \text{ for } -\infty < x < +\infty \text{ and } \tau > 0
\]

With initial condition (i.e., payoff):

\[
V(x, 0) = \max\left(0, e^{(k_2 + 1)x/2} - e^{(k_2 - 1)x/2}\right) \equiv V_0(x)
\]

We therefore have transformed the problem into a simpler form that has a well-known solution:

The solution of the heat equation (diffusion equation)

\[
\frac{\partial V}{\partial \tau} = \frac{\partial^2 V}{\partial x^2} \text{ for } -\infty < x < +\infty \text{ and } \tau > 0 \text{ with initial condition } V(x, 0) = V_0(x) \text{ is:}
\]

\[
V(x, \tau) = \frac{1}{2\sqrt{\pi \tau}} \int_{-\infty}^{+\infty} V_0(y) e^{-\frac{(x-y)^2}{4\tau}} \, dy.
\]
b. Programming the Finite-Difference Solution (Tips)

- First, redefine \( \tau \)
- Select the number of divisions of the time-to-maturity interval, and infer \( \Delta \tau \)
- Select the stock price range (min and max)
- Apply the change of variables (compute \( x, x_{\text{min}}, x_{\text{max}}, \kappa_1, \kappa_2, \alpha, \beta, \Delta x, \) etc…)
- Once you have defined your horizontal and vertical axis, you are ready to build a grid that has “\( x \)” vertically and the “new \( \tau \)” horizontally.
- Start building the grid by filling in values in the first column (i.e., the terminal payoff values), the first row (boundary value when \( x \) is very large) and the bottom row (boundary value when \( x \) is very small). Make sure to use the transformed boundary/initial/terminal condition given in the previous section.
- You can now use the formula found in the explicit finite difference technique (beginning of the notes) to fill the rest of the grid. Get your coefficients \( a, b, \) and \( h \) from the PDE for \( V(x,\tau) \). This means that the solution you obtain will be the value for \( V(x,\tau) \), not \( C(S,t) \).
- Finally convert your solution for \( V(x,\tau) \) into the value for \( C(S,t) \) by reversing the change of variable.
F. Implicit Finite Difference Method

1. The advantage of using the implicit finite differences method is that the method does not suffer from the stability issue \(0 < \Delta t/(\Delta S)^2 \leq 1/2\).

2. The approximations needed to implement the implicit method are: backward difference for \(\partial f/\partial t\), central difference for \(\partial f/\partial S\), and symmetric central difference for \(\partial^2 f/\partial S^2\).

   a. The PDE

   \[-\frac{f_{n+1}^m - f_{n-1}^m}{\Delta t} + a_n \frac{f_{n+1}^m - f_{n-1}^m}{2\Delta S} + b_n \frac{f_{n+1}^m - 2f_n^m + f_{n-1}^m}{(\Delta S)^2} + h_n - cf = 0\]

   therefore has the following approximation:

   \[-\frac{f_n^m - f_{n-1}^m}{\Delta t} + a_n \frac{f_{n+1}^m - f_{n-1}^m}{2\Delta S} + b_n \frac{f_{n+1}^m - 2f_n^m + f_{n-1}^m}{(\Delta S)^2} + h_n - cf = 0\]

   b. This equation does not give a solution for \(f\) at time \(m\) given values at time \(m-1\) directly, but gives an implicit solution for it.

   c. This equation uses boundary conditions in the same manner that the explicit method did.
3. The solution obtained for $f^m_n$ is stable regardless of how the increments for $S$ and $t$ are chosen. This is an important advantage of the methodology.

Example:

<table>
<thead>
<tr>
<th>$S$</th>
<th>$\alpha = 0.50$</th>
<th>$\alpha = 1.00$</th>
<th>$\alpha = 5.00$</th>
<th>exact</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.00</td>
<td>9.7531</td>
<td>9.7531</td>
<td>9.7531</td>
<td>9.7531</td>
</tr>
<tr>
<td>2.00</td>
<td>7.7531</td>
<td>7.7531</td>
<td>7.7531</td>
<td>7.7531</td>
</tr>
<tr>
<td>4.00</td>
<td>5.7531</td>
<td>5.7531</td>
<td>5.7530</td>
<td>5.7531</td>
</tr>
<tr>
<td>6.00</td>
<td>3.7569</td>
<td>3.7570</td>
<td>3.7573</td>
<td>3.7569</td>
</tr>
<tr>
<td>8.00</td>
<td>1.9025</td>
<td>1.9025</td>
<td>1.9020</td>
<td>1.9024</td>
</tr>
<tr>
<td>10.00</td>
<td>0.6690</td>
<td>0.6689</td>
<td>0.6675</td>
<td>0.6694</td>
</tr>
<tr>
<td>12.00</td>
<td>0.1674</td>
<td>0.1674</td>
<td>0.1670</td>
<td>0.1675</td>
</tr>
<tr>
<td>14.00</td>
<td>0.0327</td>
<td>0.0328</td>
<td>0.0332</td>
<td>0.0326</td>
</tr>
<tr>
<td>16.00</td>
<td>0.0054</td>
<td>0.0055</td>
<td>0.0058</td>
<td>0.0054</td>
</tr>
</tbody>
</table>

Comparison of exact Black-Scholes and fully implicit finite-difference solutions for a European put with $E = 10$, $r = 0.1$, $\sigma = 0.4$ and three months to expiry. Even with $\alpha = 5.0$ the results are accurate to 2 decimal places.

(WDH 1993)

4. The drawback of the technique is that the difference equation only gives an implicit solution.

a. Note however that only three different unknown values appear in the equation ($f^m_{n+1}$, $f^m_n$ and $f^m_{n-1}$):

$$
\left\{ \left( \frac{\Delta t}{\Delta S} \right)^2 b^m_n + \left( \frac{\Delta t}{\Delta S} \right) \frac{a^m_n}{2} \right\} f^m_{n+1} + \left\{ 1 - \left( \frac{\Delta t}{\Delta S} \right)^2 \right\} 2b^m_n - (\Delta t) r^m \right\} f^m_n
$$

$$
+ \left\{ \left( \frac{\Delta t}{\Delta S} \right)^2 b^m_n - \left( \frac{\Delta t}{\Delta S} \right) \frac{a^m_n}{2} \right\} f^m_{n-1} = f^m_{n-1} - h^m_n
$$
b. Also note that we do know the price of the option at the boundaries. Therefore when \( n-1 = N \) or \( n+1 = \bar{N} \), there are only two unknown prices in the difference equation, as shown on the grid above.

c. Note that even though the solution is an implicit one, we can still retrieve the derivative prices:

- At a given date \( m \), the system of equations for \( n = \bar{N} + 1 \) to \( n = \bar{N} - 1 \) is said to be tridiagonal.

- A tridiagonal system has the following form:
\[
\begin{align*}
&b_1x_1 + c_1x_2 = d_1 \\
&a_2x_1 + b_2x_2 + c_2x_3 = d_2 \\
&a_3x_2 + b_3x_3 + c_3x_4 = d_3 \\
&\vdots \\
&a_{n-1}x_{n-2} + b_{n-1}x_{n-1} + c_{n-1}x_n = d_{n-1} \\
&a_nx_{n-1} + b_nx_n = d_n
\end{align*}
\]

where the unknowns are represented by the \(x\)'s and where the \(a, b, c,\) and \(d\)'s represent known values.

d. Simple algorithms exist that solve such systems quickly (by substitution) and produce a solution for \(f_n^m\) for \(n=N\) to \(\bar{N}\).

5. The Crank-Nicholson Algorithm

a. It lies between the explicit and implicit methods.

Explicit at time \(m\):

\[
\frac{-f_n^{m+1} - f_n^m}{\Delta t} + a_n \frac{f_{n+1}^m - f_{n-1}^m}{2\Delta S} \\
+ b_n \frac{f_n^{m+1} - 2f_n^m + f_{n-1}^m}{(\Delta S)^2} + h_n^{m} - r_n^{m}f_n^{m+1} = 0
\]
Implicit at time \( m+1 \):

\[
-\frac{f_{n}^{m+1} - f_{n}^{m}}{\Delta t} + a_{n}^{m+1} \frac{f_{n+1}^{m+1} - f_{n-1}^{m+1}}{2\Delta S} + b_{n}^{m+1} \frac{f_{n+1}^{m+1} - 2f_{n}^{m+1} + f_{n-1}^{m+1}}{(\Delta S)^2} + h_{n}^{m+1} - r_{n}^{m+1} f_{n}^{m+1} = 0
\]

- The Crank-Nicholson method sums the two equations, yielding:

\[
-2\frac{f_{n}^{m+1} - f_{n}^{m}}{\Delta t} + \left\{a_{n}^{m} \frac{f_{n+1}^{m} - f_{n-1}^{m}}{2\Delta S} + a_{n}^{m+1} \frac{f_{n+1}^{m+1} - f_{n-1}^{m+1}}{2\Delta S}\right\} + \left\{b_{n}^{m} \frac{f_{n+1}^{m} - 2f_{n}^{m} + f_{n-1}^{m}}{(\Delta S)^2} + b_{n}^{m+1} \frac{f_{n+1}^{m+1} - 2f_{n}^{m+1} + f_{n-1}^{m+1}}{(\Delta S)^2}\right\}

+ \left\{h_{n}^{m} - r_{n}^{m} f_{n}^{m} + h_{n}^{m+1} - r_{n}^{m+1} f_{n}^{m+1}\right\} = 0
\]

b. Overall, the methodology is still implicit, but it has a higher degree of precision than the regular implicit method.

c. Just like the implicit method, it is also guaranteed to converge, regardless of the value for \( \alpha = \Delta t/(\Delta S)^2 \).
IV. Comparison of Numerical Methods: Finite Differences, Monte Carlo, Trees, Numerical Integration

A. Criteria

1. Convergence
   a. Convergence is attained with every method.
   b. Comparable (constant scale as they converge to zero) $\Delta t$ and $\Delta S$ are needed for each method.

2. Stability
   a. Major concern for explicit finite difference method.
   b. Guaranteed for the implicit methods.

3. Computational Time and Algorithm Complexity
   a. Explicit is more straightforward than the implicit or Crank-Nicholson method.
   b. Both trees and numerical methods for PDEs grow linearly in the grid size.
   c. Monte Carlo simulation can take a long time.

4. Ease of Implementation
   a. Monte Carlo and Numerical Integrations are easy.
   b. Explicit finite differences and trees are relatively easy.
B. Applicability Potential

1. European vs. American Options

   a. With American options, the investor must decide at every point/node up to the expiration date whether it is optimal to exercise.

   b. Due to the large number of paths, it is thus very difficult to use Monte Carlo for American options.

   c. Trees are much better suited for American options.

   d. Finite differences can also be used for American options, although slightly more complicated.

2. Higher Number of Underlying Assets or Factors

   a. Monte Carlo adapts well to many factors, as one can simulate the risk-neutral path of each asset or factor, possibly taking into account the correlation(s) between the various Brownian motions.

   b. Numerical integrations are faster, however.

   c. More difficult for trees and finite differences, because going to higher dimensions (the tree becoming a “cube”, for instance) implies quickly rising computational costs.
3. Path Dependency

a. Path dependency means that the derivative’s payoff depends not only on the terminal value of the asset but also on the path taken between t and T. Example: mean, min, or max value of the asset during that path.

b. Very easy for Monte Carlo.

c. More difficult for trees and finite differences.

4. Vanilla Options vs. Exotic Options

a. Very easy to implement if it is only a matter of changing the payoff function at expiration. Binary options are a good example of that.

b. If the option payoff is also path-dependent, see section 3.