# Finite Difference Methods

In this chapter we will see how to estimate derivative values by numerically solving the partial differential equation (pde) that the derivative value satisfies, using finite difference methods. More advanced discussions of this topic can be found in Wilmott, DeWynne and Howison [@WDH], Wilmott [@Wilmott], and Tavella [@Tavella], among other places. We will only consider derivatives written on a single underlying asset, but the ideas generalize to derivatives written on multiple underlying assets (e.g., basket and spread options) in much the same way that binomial models can be applied to derivatives on multiple underlying assets. The curse of dimensionality is the same for finite difference methods as for binomial models—the computation time increases exponentially with the number of underlying assets.

## Fundamental PDE

Consider an asset with price $S$ and constant dividend yield $q$.
Set $X = \log S$. Then we have

$$\mathrm{d}X = \nu\,\mathrm{d}t + \sigma\,\mathrm{d}B\ ,$$

where $\nu = r - q - \sigma^2/2$ and $B$ is a Brownian motion under the risk-neutral measure.

Let $T$ denote the maturity date of a derivative security. At time $t$ (when the remaining time to maturity is $T-t$), assume the price of the derivative can be represented as $C(t, X(t))$.[1] Since $C$ is a function of $t$ and $X$, Ito's formula implies

$$\mathrm{d}C = \frac{\partial C}{\partial t}\,\mathrm{d}t + \frac{\partial C}{\partial X}\,\mathrm{d}X + \frac{1}{2}\frac{\partial^2 C}{\partial X^2}(\mathrm{d}X)^2 \tag{1}$$

$$= \frac{\partial C}{\partial t}\,\mathrm{d}t + \frac{\partial C}{\partial X}(\nu\,\mathrm{d}t + \sigma\,\mathrm{d}B) + \frac{1}{2}\frac{\partial^2 C}{\partial X^2}\sigma^2\,\mathrm{d}t\ .$$

[1] If the price of the derivative is a function of the asset price $S$ and time, then we can always write it in this form as a function of the natural logarithm of $S$ and time.

On the other hand, under the risk-neutral measure, the instantaneous expected rate of return on the derivative is the risk-free rate, so

$$\frac{\mathrm{d}C}{C} = r\,\mathrm{d}t + \text{something}\ \mathrm{d}B\ .$$

where the something is the volatility of the derivative value. We can of course rearrange this as

$$\mathrm{d}C = rC\,\mathrm{d}t + \text{something}\ C\,\mathrm{d}B\ . \tag{2}$$

In order for both Equation **??** and Equation **??** to hold, the drifts on both right-hand sides must be equal.[2] This implies

$$rC = \frac{\partial C}{\partial t} + \nu\frac{\partial C}{\partial X} + \frac{1}{2}\sigma^2\frac{\partial^2 C}{\partial X^2}\ . \tag{3}$$

This equation is the fundamental pde. It is an equation that we want to solve for the function $C$. Every derivative written on $S$ satisfies this same equation. Different derivatives have different values because of boundary conditions. The boundary conditions are the intrinsic value at maturity, optimality conditions for early exercise, barriers and the like.

To translate the terms in Equation **??** into more familiar ones, notice that, because $S = \mathrm{e}^X$, we have

$$\frac{\partial S}{\partial X} = \mathrm{e}^X = S\ .$$

Therefore, by the chain rule of calculus,

$$\frac{\partial C}{\partial X} = \frac{\partial C}{\partial S}\frac{\partial S}{\partial X} = S\frac{\partial C}{\partial S}\ .$$

Thus the term $\partial C/\partial X$ is the delta of the derivative multiplied by the price of the underlying. Similarly, by ordinary calculus, the term $\partial^2 C/\partial X^2$ can be written in terms of the delta and the gamma of the derivative.

Sometimes one writes the derivative value as a function of time to maturity ($\tau = T - t$) instead of $t$. The partial derivative of $C$ with respect to $\tau$ is the negative of the partial derivative with respect to $t$, so the fundamental pde is the same except for a different sign on the first term of the right-hand side of Equation **??**. Rearranging a little, we have

$$\frac{\partial C}{\partial \tau} = -rC + \nu\frac{\partial C}{\partial X} + \frac{1}{2}\sigma^2\frac{\partial^2 C}{\partial X^2}\ . \tag{4}$$

[2] Suppose a process $X$ satisfies $\mathrm{d}X = \alpha_1\,\mathrm{d}t + \sigma_1\,\mathrm{d}B = \alpha_2\,\mathrm{d}t + \sigma_2\,\mathrm{d}B$ for coefficients $\alpha_i$ and $\sigma_i$. This implies $(\alpha_1 - \alpha_2)\,\mathrm{d}t = (\sigma_2 - \sigma_2)\,\mathrm{d}B$. The right-hand side defines a (local) martingale and the left-hand side defines a continuous finite-variation process. As discussed in **?@sec-s_quadraticvariation**, the only continuous finite-variation martingales are constants, so the changes must be zero; i.e., $\alpha_1 = \alpha_2$ and $\sigma_1 = \sigma_2$.

In this form, the pde is similar to important equations in physics, in particular the equation for how heat propagates through a rod over time. In fact, it can be transformed exactly into the heat equation, which is how Black and Scholes originally solved the option valuation problem. The terminal condition for a call option, $C = \max(S-K, 0)$, can be viewed as defining $C$ over the $X$ dimension at $\tau = 0$, just as the temperature along the length of the rod might be specified at an initial date, and as $\tau$ increases $C$ changes at each point $X$ according to Equation **??**, which is similar, as noted, to the equation for the change in temperature at a point on the rod as time passes.

## Discretizing the PDE

To numerically solve the fundamental pde, we consider a discrete grid on the $(t, x)$ space. We label the time points as $t_0, t_1, t_2, \ldots, t_N$, and the $x$ points as $x_{-M}, x_{-M+1}, \ldots, x_0, x_1, \ldots, x_M$, with $t_0 = 0$, $t_N = T$, and $x_0 = \log S(0)$. The equation should hold for $-\infty < x < \infty$, but obviously we will have to bound this space, and we have denoted the upper and lower bounds by $x_M$ and $x_{-M}$ here. We take the points to be evenly spaced and set $\Delta t = t_i - t_{i-1}$ and $\Delta x = x_j - x_{j-1}$ for any $i$ and $j$.

For specificity, we will consider a call option, though the discussion in this section applies to any derivative. We will compute a value for the call at each of the points on the grid. Then we return the value of the call at the point $(t_0, x_0)$.

Consider a point $(t_i, x_j)$. We could denote the estimated value of the call at this point by $C_{ij}$ but for now we will just use the symbol $C$. Think of $t$ being on the horizontal axis and $x$ on the vertical axis. There are four points that can be reached from $(t_i, x_j)$ by one step (an increase or decrease) in either $t$ or $x$. Let's denote the estimated call value at $(t_i, x_j + \Delta x)$ as $C_{\text{up}}$, the value at $(t_i, x_j - \Delta x)$ as $C_{\text{down}}$, the value at $(t_i + \Delta t, x_j)$ as $C_{\text{right}}$ and the value at $(t_i - \Delta t, x_j)$ as $C_{\text{left}}$.

We want to force Equation **??** to hold on the grid. To estimate $\partial C/\partial X$ and $\partial^2 C/\partial X^2$, we make exactly the same calculations we made to estimate deltas and gammas in a binomial model. At the point $(t_i, x_j)$, we estimate

$$\frac{\partial C}{\partial X} \approx \frac{C_{\text{up}} - C_{\text{down}}}{2\Delta x} \; . \tag{5}$$

There are two other obvious estimates of this derivative:

$$\frac{C_{\text{up}} - C}{\Delta x} \qquad \text{and} \qquad \frac{C - C_{\text{down}}}{\Delta x} \; .$$

The first of these should be understood as an estimate at the midpoint of $x_j$ and $x_j + \Delta x$ and the second as an estimate at the midpoint of $x_j$ and $x_j - \Delta x$. The distance between these two midpoints is $\Delta x$, so the difference in these two estimates of $\partial C/\partial X$ divided by $\Delta x$ is an estimate of the second derivative:

$$\frac{\partial^2 C}{\partial X^2} \approx \frac{C_{\text{up}} - 2C + C_{\text{down}}}{(\Delta x)^2} \; . \tag{6}$$

The obvious estimate of $\partial C/\partial t$, which is analogous to the estimate of $\partial C/\partial X$, is

$$\frac{C_{\text{right}} - C_{\text{left}}}{2\Delta t} \; .$$

This is *not* the estimate we are going to use. The reason is that we want to solve for the call values on the grid in much the same way that we solved the binomial model—starting at the end and working backwards. If we use the above estimate of the time derivative, then at each point $(t_i, x_j)$, Equation **??** will link the call values at times $t_{i-1}$, $t_i$ and $t_{i+1}$. This would substantially complicate the backing up process. However, in a sense, it is the right estimate, and the Crank-Nicolson method to be discussed below uses a similar idea.

The other two choices for estimating $\partial C/\partial t$ are analogous to the other two choices for estimating $\partial C/\partial X$. We can use either

$$\frac{\partial C}{\partial t} \approx \frac{C - C_{\text{left}}}{\Delta t} \; , \tag{7}$$

or

$$\frac{\partial C}{\partial t} \approx \frac{C_{\text{right}} - C}{\Delta t} \; . \tag{8}$$

Using the first is called the explicit method of solving the pde, and using the second is called the implicit method. The reason for these names should become clear below.

## Explicit and Implicit Methods

We first consider the explicit method. We set the value of the call at the final date $t_N$ and each point $x_j$ to be its intrinsic value, $\max\left(\mathrm{e}^{x_j} - K, 0\right)$. Now consider calculating the value at date $t_{N-1}$ and any point $x_j$. We do this by forcing the approximation to Equation **??** based on Equation **??**–Equation **??** to hold at the point $(t_N, x_j)$. Using the same notation as before, for $(t_i, x_j) = (t_N, x_j)$, implies

$$rC = \frac{C - C_{\text{left}}}{\Delta t} + \nu\left(\frac{C_{\text{up}} - C_{\text{down}}}{2\Delta x}\right) + \frac{1}{2}\sigma^2\left(\frac{C_{\text{up}} + C_{\text{down}} - 2C}{(\Delta x)^2}\right) \; . \tag{9}$$

Given that $t_i$ is the final date $t_N$, the values $C$, $C_{\text{up}}$ and $C_{\text{down}}$ have already been calculated as the intrinsic value of the call at maturity. The only unknown is $C_{\text{left}}$, which is the value of the call at $(t_{N-1}, x_j)$. We can solve this *explicitly* for $C_{\text{left}}$, whence the name of the algorithm. We do this at each point $x_j$ at date $t_{N-1}$ (except for the top and bottom points, which we will discuss below) and then we follow the same procedure to back up sequentially to the initial date, as in the binomial model.

Equation Equation **??** cannot be used to find $C_{\text{left}}$ at the bottom point $x_{-M}$, because at this point there is no $C_{\text{down}}$ at date $t_N$. Similarly, we cannot use it to find $C_{\text{left}}$ at the top point $x_M$, because at that point there is no $C_{\text{up}}$. We have to define the values along the top and bottom of the grid in some other fashion. We do this using conditions the derivative is known to satisfy as the stock price approaches $+\infty$ or 0. For example, for a European call option, we use the conditions that $\partial C/\partial S \to 1$ as $S \to \infty$ and $\partial C/\partial S \to 0$ as $S \to 0$. We will explain this in more detail in the following section.

The solution of Equation **??** for $C_{\text{left}}$ can be written as

$$C_{\text{left}} = (1 - r\Delta t)(p_u C_{\text{up}} + pC + p_d C_{\text{down}}) \; , \tag{10}$$

where

$$\begin{aligned} p_u &= \frac{\sigma^2 \Delta t + \nu \Delta t \Delta x}{2(1 - r\Delta t)(\Delta x)^2} \; , \\ p_d &= \frac{\sigma^2 \Delta t - \nu \Delta t \Delta x}{2(1 - r\Delta t)(\Delta x)^2} \; , \\ p &= 1 - p_u - p_d \; . \end{aligned}$$

This can be interpreted as discounting the probability-weighted values of the call at the next date, where we consider that starting at the grid point $(t_i, x_j)$, the logarithm of the stock price takes three possible values ($x_j - \Delta x$, $x_j$, and $x_j + \Delta x$) at the next date $t_{i+1}$, and where we use $1 - r\Delta t$ as the discount factor. Thus, it is essentially a trinomial model. This relationship was first noted by Brennan and Schwartz [@BrennanSchwartz].

Actually, for this to be a sensible trinomial model, the probabilities $p_u$, $p$ and $p_d$ should be nonnegative. Assuming $1 - r\Delta t > 0$, this will be the case if and only if

$$\Delta x \leq \frac{\sigma^2}{|\nu|} \qquad \text{and} \qquad \Delta t \leq \frac{(\Delta x)^2}{\sigma^2 + r(\Delta x)^2} \; .$$

The first of these conditions characterizes $p_u$ and $p_d$ being nonnegative. The second is derived from $p_u + p_d \leq 1$. It is interesting to examine these conditions in terms of the number $N$ of time periods and the number of steps in the $x$ dimension, which is $2M$. To simplify the notation in the following somewhat, denote the distance of the upper $x$ boundary from $x_0$ by

$D$ (i.e., $D = x_M - x_0$). Then $\Delta t = T/N$ and $\Delta x = D/M$. The probabilities are nonnegative if and only if

$$M \geq \frac{|\nu| D}{\sigma^2} \qquad \text{and} \qquad N \geq rT + \left(\frac{\sigma^2 T}{D^2}\right) M^2 \,.$$

Consider fixing $D$ and increasing the number of time periods and space steps (i.e., steps along the $x$ dimension). To maintain positive probabilities, the above shows that the number of time periods must increase as the square of the number of space steps: increasing $M$ by a factor of 10 requires increasing $N$ by a factor of 100. The upshot is it can be computationally expensive to use a large number of space steps, if we want to maintain nonnegative probabilities.

One can reasonably ask whether this is important, because we can certainly solve Equation **??** to estimate the call values even when the probabilities are negative. The answer is that it is important, but for a reason we have not yet discussed. In a numerical algorithm for solving a partial differential equation (or for solving many other types of problems) there are two types of errors: discretization error and roundoff error. If we increase $N$ and $M$ sufficiently, we should reduce the discretization error. However, each calculation on the computer introduces roundoff error. An algorithm is said to be stable if the roundoff errors stay small and bounded as the discretization error is reduced. An unfortunate fact about the explicit method is that it is stable only if the number of time steps increases with the square of the number of space steps. In the absence of this condition, the roundoff errors can accumulate and prevent one from reaching a solution of the desired accuracy.

The implicit method is known to be fully stable, so it is to be preferred to the explicit method. We will discuss briefly how to implement this method, before moving in the next section to the Crank-Nicolson method, which is also fully stable and known to be more efficient than the implicit method.

The implicit method uses the approximation Equation **??** for $\partial C/\partial t$. As before, the call values are defined at the final date as the intrinsic value. Backing up a period, consider a grid point $(t_{N-1}, x_j)$. We will try to estimate the call value at this date by forcing Equation **??** to hold at this point. This means

$$rC = \frac{C_{\text{right}} - C}{\Delta t} + \nu \left(\frac{C_{\text{up}} - C_{\text{down}}}{2\Delta x}\right) + \frac{1}{2}\sigma^2 \left(\frac{C_{\text{up}} + C_{\text{down}} - 2C}{(\Delta x)^2}\right) \,. \tag{11}$$

We know $C_{\text{right}}$, because it is the intrinsic value at $(t_N, x_j)$. This equation links three unknowns ($C$, $C_{\text{up}}$, and $C_{\text{down}}$) to the known value $C_{\text{right}}$. We cannot solve it explicitly for these three unknowns. Instead, we need to solve a system of linear equations to simultaneously solve for all the call values at date $t_{N-1}$. There are $2M - 1$ equations of the form Equation **??** plus conditions that we will impose at the upper and lower boundaries, and we need to solve these for the $2M + 1$ call values. This system of equations has the same form, and is solved in the same way, as the system of equations in the Crank-Nicolson method.

## Crank-Nicolson

The estimate Equation **??** of $\partial C/\partial t$ used in the implicit method is best understood as an estimate of $\partial C/\partial t$ at the midpoint of $(t_i, x_j)$ and $(t_{i+1}, x_j)$, i.e., at $(t_i + \Delta t/2, x_j)$. This is the basic idea of the Crank-Nicolson method. With this method, we continue to estimate the call values at the grid points, but we do so by forcing Equation **??** to hold at midpoints of this type. To do this, we also need estimates of $C$, $\partial C/\partial X$ and $\partial^2 C/\partial X^2$ at the midpoints, but these are easy to obtain.

Let's modify the previous notation somewhat, writing $C'$ for $C_{\text{right}}$ and $C'_{\text{up}}$ and $C'_{\text{down}}$ for the values to the right and one step up and down, i.e., at the grid points $(t_i + \Delta t, x_i + \Delta x)$ and $(t_i + \Delta t, x_i - \Delta x)$ respectively. The obvious estimate of the call value at the midpoint $(t_i + \Delta t/2, x_j)$ is the average of $C$ and $C'$, so set

$$C^{\text{mid}} = \frac{C + C'}{2} \ .$$

Analogously, define

$$C^{\text{mid}}_{\text{up}} = \frac{C_{\text{up}} + C'_{\text{up}}}{2} \ , \qquad \text{and} \qquad C^{\text{mid}}_{\text{down}} = \frac{C_{\text{down}} + C'_{\text{down}}}{2} \ . \tag{12}$$

The formulas Equation **??** give us estimates of the call value at the midpoints one space step up and one space step down from $x_j$—i.e., at $(t_i + \Delta t/2, x_{j+1})$ and $(t_i + \Delta t/2, x_{j-1})$. We can now estimate $\partial C/\partial X$ and $\partial^2 C/\partial X^2$ at the midpoint $(t_i + \Delta t/2, x_j)$ exactly as before:

$$\frac{\partial C}{\partial X} \approx \frac{C^{\text{mid}}_{\text{up}} - C^{\text{mid}}_{\text{down}}}{2\Delta x} \ ,$$

and

$$\frac{\partial^2 C}{\partial X^2} \approx \frac{C^{\text{mid}}_{\text{up}} + C^{\text{mid}}_{\text{down}} - 2C^{\text{mid}}}{(\Delta x)^2} \ .$$

Now, Equation **??** becomes

$$rC^{\text{mid}} = \frac{C' - C}{\Delta t} + \nu \left( \frac{C^{\text{mid}}_{\text{up}} - C^{\text{mid}}_{\text{down}}}{2\Delta x} \right) + \frac{1}{2}\sigma^2 \left( \frac{C^{\text{mid}}_{\text{up}} + C^{\text{mid}}_{\text{down}} - 2C^{\text{mid}}}{(\Delta x)^2} \right) \ . \tag{13}$$

Substituting from the formulas for $C^{\text{mid}}$, $C^{\text{mid}}_{\text{up}}$, and $C^{\text{mid}}_{\text{down}}$, we can re-write Equation **??** as

$$\begin{aligned}
&\left( \frac{r}{2} + \frac{1}{\Delta t} + \frac{\sigma^2}{2(\Delta x)^2} \right) C - \left( \frac{\sigma^2}{4(\Delta x)^2} + \frac{\nu}{4\Delta x} \right) C_{\text{up}} \\
&\qquad - \left( \frac{\sigma^2}{4(\Delta x)^2} - \frac{\nu}{4\Delta x} \right) C_{\text{down}} \quad = \quad \left( \frac{1}{\Delta t} - \frac{r}{2} - \frac{\sigma^2}{2(\Delta x)^2} \right) C' \\
&\qquad\qquad + \left( \frac{\sigma^2}{4(\Delta x)^2} + \frac{\nu}{4\Delta x} \right) C'_{\text{up}} + \left( \frac{\sigma^2}{4(\Delta x)^2} - \frac{\nu}{4\Delta x} \right) C'_{\text{down}}
\end{aligned} \tag{14}$$

{#eq-crank2}

We can also write this as

$$a_1 C - a_2 C_{\text{up}} - a_3 C_{\text{down}} = a_4 C' + a_2 C'_{\text{up}} + a_3 C'_{\text{down}} \ , \tag{15}$$

where the constants $a_i$ are the factors in parentheses in **?@eq-crank2**.

As before, we start at the final date $t_N$ and define the call value at that date by its intrinsic value. Consider a grid point $(t_{N-1}, x_j)$. Forcing Equation **??** to hold at the midpoint $(t_{N-1}+\Delta t/2, x_j)$ leads us to Equation **??**. In this equation, $C'$, $C'_{\text{up}}$ and $C'_{\text{down}}$ are known from the intrinsic value at maturity, and we need to solve for $C$, $C_{\text{up}}$ and $C_{\text{down}}$. There are $2M-1$ linear equations of this type and we will add linear equations at the upper and lower boundaries of the grid and solve the resulting system of $2M+1$ linear equations for the $2M+1$ call values. After finding the call values at date $t_{N-1}$, we then repeat the calculation at $t_{N-2}$ and continue backing up in this way until we reach the initial date.

Notice that the Crank-Nicolson equations **?@eq-crank2** are similar to the equations Equation **??** in the implicit method, but more information is used in each step of the Crank-Nicolson method than is used in each step of the implicit method. Equation **?@eq-crank2** links the call values $C$, $C_{\text{up}}$ and $C_{\text{down}}$ to the previously calculated $C'$, $C'_{\text{up}}$ and $C'_{\text{down}}$, whereas in the implicit method they were linked only to $C'$ (which we called $C_{\text{right}}$).

## European Options

To value a European option, one simply defines the values at the final date as the intrinsic value and then backs up to the initial date, using any of the methods described (explicit, implicit, or Crank-Nicolson). The value that should be returned is the value at the middle node at the initial date, which corresponds to the initial price of the underlying.

The boundary conditions normally used at the bottom and top of the grid are conditions that the first derivative $\partial C/\partial S$ of the option value are known to satisfy as $S \to 0$ and $S \to \infty$. These are conditions of the form

$$\lim_{S\to\infty} \frac{\partial C}{\partial S} = \lambda_0 \ , \qquad \text{and} \qquad \lim_{S\to 0} \frac{\partial C}{\partial S} = \lambda_\infty \ , \tag{16}$$

for constants $\lambda_0$ and $\lambda_\infty$. In the case of a call option, we have $\lambda_0 = 0$ and $\lambda_\infty = 1$. For a put option, we have $\lambda_0 = -1$ and $\lambda_\infty = 0$.

These conditions are implemented on the grid by forcing each value $C$ at a point $(t_i, x_{-M})$ on the bottom of the grid to satisfy

$$C - C_{\text{up}} = \lambda_0 (S - S_{\text{up}}) \tag{17}$$