Chapter Two. Finite Difference Methods

1. The Concept of Finite Difference Method

In FDM, we represent continuous fluid flow problems in a discrete manner, when the fluid continuum is replaced by a mesh of discrete points. The same is true for the time variable.

FDM are the simplest of all approximations, and involve a mapping:

$$\text{PDE} \xrightarrow{\text{Discretization}} \text{System of algebraic equtions}$$

Calculus → algebra

Derivative → difference

We focus on the following:

- Properties of FDM
- Derivation via several methods
- Physical interpretation in terms of characteristics
- Application to selected problems

First, we lay down a convention for notion:

Time level - superscript n - ρ^n ~ ρ at time level n

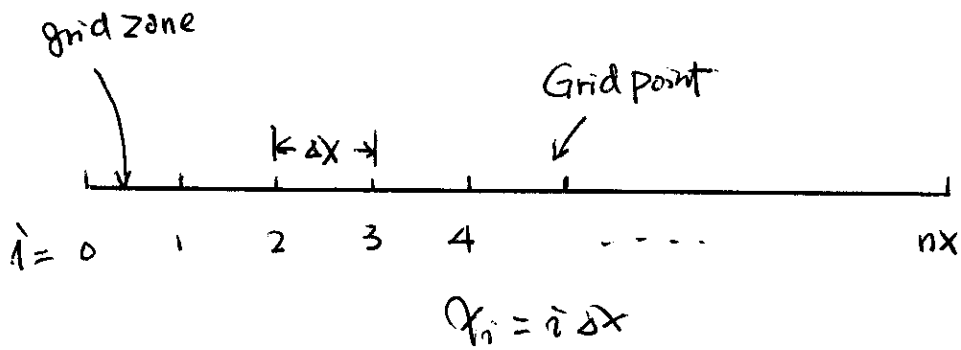
Δt = time interval = $t^{n+1} - t^n$.

Most times, we use constant Δt. Occasionally, Δt changes with time.

n-1 ~ past
n ~ present
n+1 ~ future

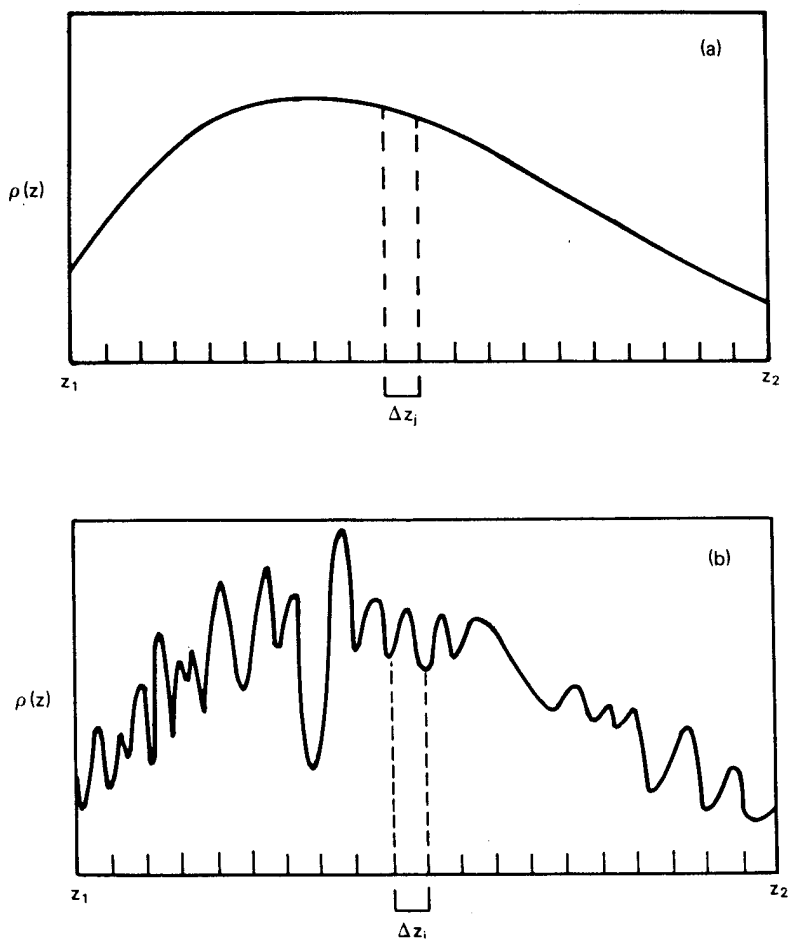
$t = n\,\Delta t$ where n = number of time steps = 0, 1, 2, 3, ….., N
$T = N\,\Delta t$ = final time.

Spatical Location – subscript i, j, k, for x, y, and z.



Δx – constant grid interval
$x_i = i\,\Delta x$

Note: Discretization ➔ information loss – the greater the number of points, the more accurate will be the representation. See Figure.



2. Quantitative Properties of Numerical Algorithms

The governing equations (PDE's) obey has certain properties, and their computational counterparts should also so.

1) Conservation – Typically the governing equations are written as conservation laws (which means that the integral properties over a closed volume don't change with time).

E.g., the mass conservation equation

$$\frac{\partial r}{\partial t} = -\nabla \cdot (r\vec{V}).$$

If we integrate this over a closed box

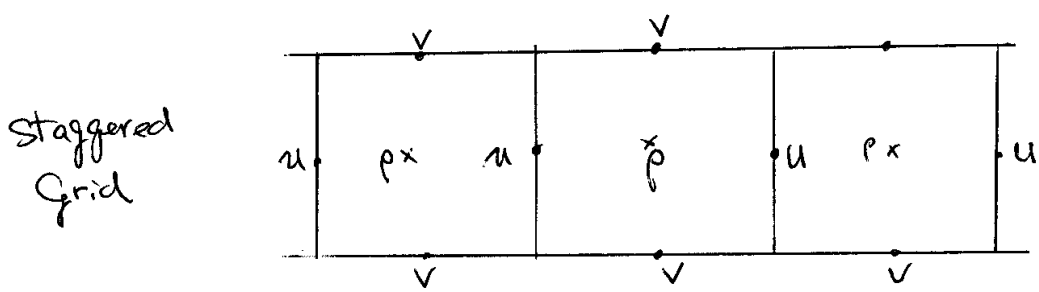
$$\frac{\partial}{\partial t}\int_{\Omega} r\, dV = -\int_{\Omega} \nabla \cdot (r\vec{V}) dv = 0$$

Mathematically, we can also write this as

$$\frac{\partial r}{\partial t} = -r\nabla \cdot \vec{V} - \vec{V} \cdot \nabla r$$

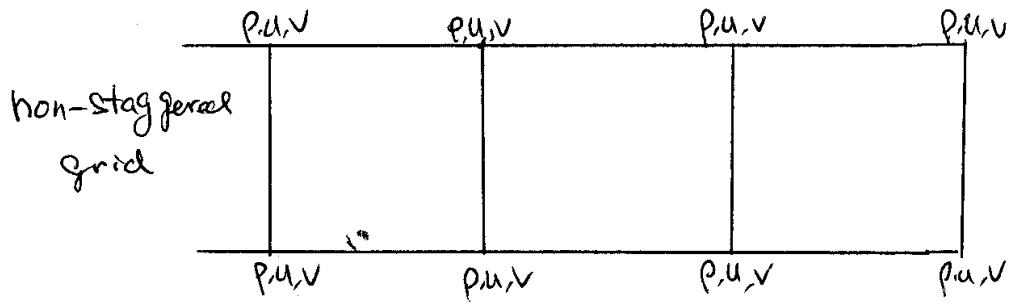
Will the numerical solution obey these rules? Not necessarily.

Consider the situation where ρ and $\vec{V}$ are defined at separate points, … this is how the continuity equation is really derived:



The mass within the zone changes due the fluxes through the side. To get $r\vec{V}$ at a point, we have to average, which smears out gradients!

Consider an alternative structure:



To calculate the fluxes through the sides of the grid cells shown above (which is a non-staggered Arakawa A-grid, by the way – we will come to it later), we have to perform different averages, which result in different conservation properties of the numerical scheme.

For numerical solution to obey conservation, you must be very careful how you set up the grid and solve the equations!!

2) Positivity – Physically positive quantities (mass, energy, water vapor) cannot become negative. This is not guaranteed with numerical solutions, however. Care must be taken to prevent negative values from being generated. Schemes that do so are called positive definite schemes. A more general type is the monotonic schemes that also ensure positive definiteness, because the cannot generate new extrema not found in the original field.

3). Reversibility – Says that the equations are invariant under the transform t → - t. This is mportant for pure transport problems, but clearly not appropriate for diffusion problems. Reversibility is actually hard to achieve even for simple advection/transportation due to unavoidable numerical errors.

4). Accuracy – Accuracy generally involves Computer precision, Spatial or temporal resolution, and algorithm robustness, etc.

Some of the most accurate schemes don't satisfy the above properties!!

2. Methods for Obtaining FD Expressions

There are several, and we will look at a few:

1) Taylor series expansion – the most common, but purely mathematical.

2) Polynomial fitting or interpolation – the most general ways. Taylor series is a subset of this method. Interpolation takes us back to the M.O.C. and thus has a more physical interpreatation.

3) Control volume approach – also called finite volume (FV) – we solve the equations in integral rather than differential form. Popular in engineering where complex geometries and coordinate transformations are involved. For Cartesian grids, simplest FV methods → FD.

We will look at only the first two approaches.

2.1. Taylor Series Expansion Method

Recall the definition of a derivative:

$$\left.\frac{\partial u}{\partial x}\right|_{x_0, y_0} = \lim_{\Delta x \to 0} \frac{u(x_0 + \Delta x, y_0) - u(x_0, y_0)}{\Delta x}$$

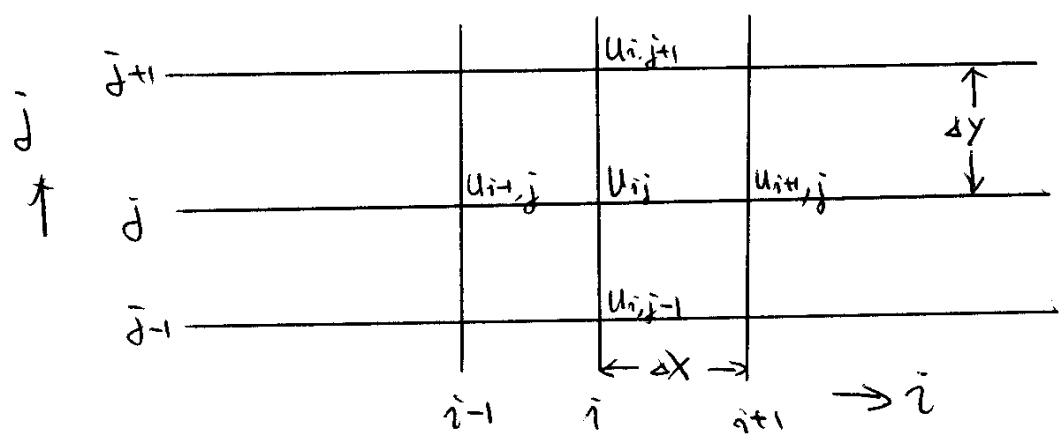
The Taylor series approach works backwards – want to approximate $\partial u/\partial x$ by a discrete difference, i.e., for finite Δx.

Given $u(x_0, y_0)$, we can write a Taylor series expansion for $u(x_0+\Delta x, y_0)$, as

$$u(x_0 + \Delta x) = u(x_0) + \Delta x \left.\frac{\partial u}{\partial x}\right|_{x_0, y_0} + \frac{(\Delta x)^2}{2!} \left.\frac{\partial^2 u}{\partial x^2}\right|_{x_0, y_0} + \frac{(\Delta x)^3}{3!} \left.\frac{\partial^3 u}{\partial x^3}\right|_{x_0, y_0} + \ldots = \sum_{n=0}^{\infty} \frac{(\Delta x)^n}{n!} \frac{\partial^n u}{\partial x^n}$$

This expression is exact if we retain all terms!

The grid mesh or stencil looks like:

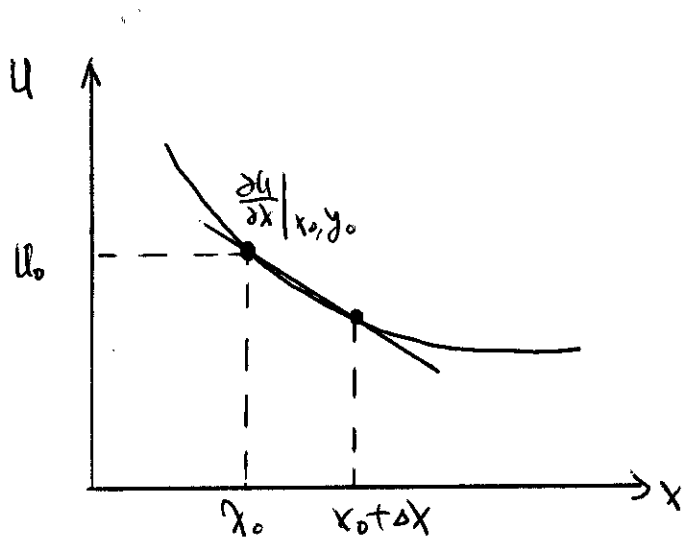


Here, $u_{i+1,j} = u(x_0+\Delta x, y_0)$
$u_{i,j+1} = u(x_0, y_0+\Delta y)$
etc.

If we solve for $\partial u/\partial x$ from the Taylor series, we have

$$\left.\frac{\partial u}{\partial x}\right|_{x_0,y_0} = \frac{u(x_0+\Delta x, y_0) - u(x_0, y_0)}{\Delta x} - \frac{\Delta x}{2!}\left.\frac{\partial^2 u}{\partial x^2}\right|_{x_0,y_0} - \frac{(\Delta x)^2}{3!}\left.\frac{\partial^3 u}{\partial x^3}\right|_{x_0,y_0} + \ldots \qquad (1)$$

The first term on the RHS is simply the slope of the function u(x,y,), using the current and the points to the right.



Therefore, $\left.\frac{\partial u}{\partial x}\right|_{x_0,y_0} = \frac{u(x_0+\Delta x, y_0) - u(x_0, y_0)}{\Delta x} + \mathrm{O}(\Delta x)$.

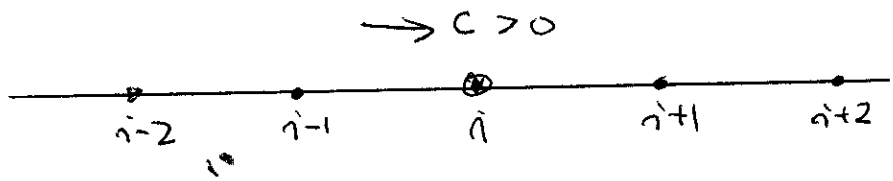
One can also use the value at $x_0 - \Delta x$ instead to get

$$\left.\frac{\partial u}{\partial x}\right|_{x_0,y_0} = \frac{u(x_0, y_0) - u(x_0-\Delta x, y_0)}{\Delta x} + \frac{\Delta x}{2!}\left.\frac{\partial^2 u}{\partial x^2}\right|_{x_0,y_0} - \frac{(\Delta x)^2}{3!}\left.\frac{\partial^3 u}{\partial x^3}\right|_{x_0,y_0} + \ldots \qquad (2)$$

Both (1) and (2) provides an expression for ∂u/∂x, but <u>numerically</u> the answers will be different.

(1) is called a <u>forward difference</u>
(2) is called a <u>backward difference</u>

Consider a 1-D example:



If we have the equation

$$\frac{\partial u}{\partial t} + c\frac{\partial u}{\partial x} = 0, \quad \text{where } c > 0,$$

we might want to use

$$\frac{\partial u}{\partial t} + c\frac{u_i - u_{i-1}}{\Delta x} \approx 0,$$

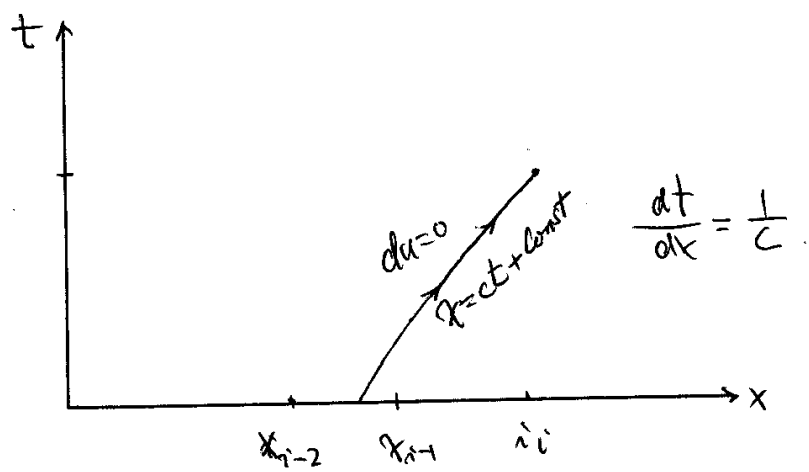
which is called <u>Upwind Difference</u>. Alternatively, in

$$\frac{\partial u}{\partial t} + c\frac{u_{i+1} - u_i}{\Delta x} \approx 0,$$

<u>Downstream Difference</u> is used.

Upstream difference is better than downstream difference for this problem, because for this pure advection problem, signals move from upstream (left) to downstream (right). The value of u at point i at a future time should be influenced by the values of u upstream, not downstream.

Think of it in terms of characteristics:



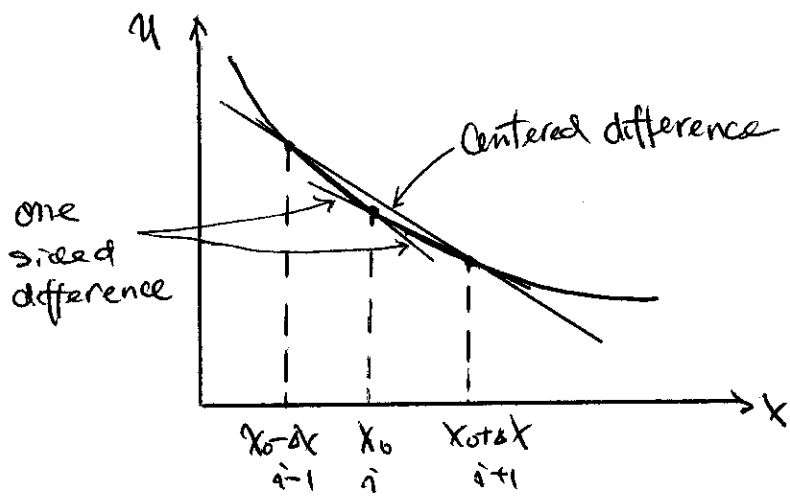
No information is coming from the +x direction. This of course depends on the sign of c. If C<0, then upstream means the left side of the current point.

Note that upstream or upstream-biased schemes are usually better choices in CFD. We will see some time later, that 1st-order downstream difference is absolutely unstable for the above problem.

We can get another discrete approximation to ∂u/∂x by adding (1) and (2):

$$\left.\frac{\partial u}{\partial x}\right|_{x_0,y_0} = \frac{u_{i+1}-u_{i-1}}{2\,\Delta x} - \frac{(\Delta x)^2}{3!}\left.\frac{\partial^3 u}{\partial x^3}\right|_{x_0,y_0} + \ldots \qquad (3)$$

This is called Centered Difference. Note, it doesn't even use value of u at the current point i. It approximates the slope using two neighboring points:



Note that this will not be accurate if u has very sharp gradient (∧). Note also that the extra term in (3) are different – they control the accuracy.

We can build many different approximations to the derivatives through linear combinations of various T.S. expansions.

For <u>high-order derivatives</u>, we follow the same approach.

Consider $\partial^2 u/\partial x^2$. The Taylor series gives

$$u(x+\Delta x) = u(x) + \Delta x \frac{\partial u}{\partial x} + \frac{(\Delta x)^2}{2!}\frac{\partial^2 u}{\partial x^2} + \frac{(\Delta x)^3}{3!}\frac{\partial^3 u}{\partial x^3} + \ldots \qquad (4)$$

We can solve for $\frac{\partial^2 u}{\partial x^2}$ from (4), but we don't want to have the unknown $\frac{\partial u}{\partial x}$.
We can replace it with one of the earlier approximations to it, or make use of the following:

$$u(x-\Delta x) = u(x) - \Delta x \frac{\partial u}{\partial x} + \frac{(\Delta x)^2}{2!}\frac{\partial^2 u}{\partial x^2} - \frac{(\Delta x)^3}{3!}\frac{\partial^3 u}{\partial x^3} + \ldots \qquad (5)$$

and add (4) and (5) and solve for $\frac{\partial^2 u}{\partial x^2}$:

$$\frac{\partial^2 u}{\partial x^2} = \frac{u(x+\Delta x) - 2u(x) + u(x-\Delta x)}{(\Delta x)^2} + \mathrm{O}(\Delta x^2)$$

which is a centered difference for $\frac{\partial^2 u}{\partial x^2}$.

<u>Truncation Error</u>

The high-order terms (H.O.T.) of $O(\Delta x^n)$ are called the <u>Truncation Error</u>, and are a measure of the error associated with representing a PDE by a <u>truncated T.S.</u> – we can't retain all terms, so we <u>neglect</u> the terms of order $O(\Delta x^2)$ and above, for example.

$$\text{PDE} = \text{FDE} + \tau$$

where τ is the <u>Truncation Error</u>.

If we change the nature of the H.O.T., by using a different approximation, the accuracy of the F.D. expression will also change.

It is important to understand the impact of τ, and this is the subject of our computer problem #4.

From the above, it's clear that we want $\tau \rightarrow 0$ when PDE = FDE. Otherwise, we have a problem (consistency)!

Let $\Delta x \rightarrow 0$, then τ should $\rightarrow 0$ => our <u>discrete</u> system approaches <u>continuum</u> and our FDE $\rightarrow$ PDE.

<u>Order of the F.D. Approximation</u>.

The power to which the leading discrete interval in τ is raised is called the <u>Order of the F.D. Approximation</u>.

Example: $\dfrac{u_i - u_{i-1}}{\Delta x}$ is first-order accurate in space

because $t = \dfrac{\Delta x}{2!}\dfrac{\partial^2 u}{\partial x^2} + \ldots = O(\Delta x^1)$

$\dfrac{u_{i+1} - 2u_i + u_{i-1}}{\Delta x^2}$ is second order accurate.

Comments on τ:

(1) If there are several independent variables, each has a truncation error, e.g., $O(\Delta x^2 + \Delta t)$, we say it's first order in time and second order in space.

(2) The <u>order</u> of a scheme also depends on the <u>local</u> properties of the function, it may be much less than the <u>formal</u> or <u>theoretical</u> order near sharp gradients. Recall that Taylor series are valid only for <u>smooth and continuous functions</u>.

(3) Typically, we prefer <u>higher-order scheme</u> because τ is smaller. However, τ is not <u>necessarily</u> related to <u>accuracy</u> because, <u>for a given Δx</u>, a first-order scheme may give more accurate results because the coefficient is smaller. What does τ tells us is how the error will <u>change</u> as we change the resolution. Higher order τ decreases faster when we decreases Δx.

e.g., $\tau = \kappa\,(\Delta x)^3$ versus $\tau = \kappa\,\Delta x$.

Remember that τ includes κ, which might be $\dfrac{\partial^4 u}{\partial x^4}$, etc and it matters.

(4) Truncation error is cumulative – adds up per time step.

(5) Truncation error is usually much larger than <u>machine round-off error</u>. Also for most problems, τ(space) >> τ(time), because solution often evolves smoothly in time but have rapid changes in space. Time step size is often not as large as we wish because of stability constraint.

General Method for Deriving FD Expressions

Let's write a generalized form of the Taylor series as

$$u_{i+1} = \sum_{m=0}^{\infty} \left(\frac{\partial^m u}{\partial x^m} \right)_i \frac{(\Delta x)^m}{m!}$$

Now, suppose we want to use a 3-point stencil at i-1, i, i+1. Then, we can write a generic expression (PDE = FDE + τ) as:

$$\frac{\partial u}{\partial x} = a u_{i-1} + b u_i + c u_{i+1} + \mathrm{O}(\Delta x)^m$$

where a, b and c are unknown constants to be determined and m is the order of the approximation. (General rule: If F.D. spans n points, you can derive an n-1 order F.D. scheme).

Using our Taylor series for u_{i-1}, u_i and u_{i+1}, we can write

$$\begin{aligned} a u_{i-1} + b u_i + c u_{i+1} = & \\ & a(u_i - \Delta x \frac{\partial u}{\partial x} + \frac{(\Delta x)^2}{2!}\frac{\partial^2 u}{\partial x^2} - \frac{(\Delta x)^3}{3!}\frac{\partial^3 u}{\partial x^3} + \ldots) \\ & + b u_i \\ & c(u_i + \Delta x \frac{\partial u}{\partial x} + \frac{(\Delta x)^2}{2!}\frac{\partial^2 u}{\partial x^2} + \frac{(\Delta x)^3}{3!}\frac{\partial^3 u}{\partial x^3} + \ldots) \\ = (a+b+c)u_i + (-a+c)\Delta x \frac{\partial u}{\partial x} & + (a+c)\frac{(\Delta x)^2}{2!}\frac{\partial^2 u}{\partial x^2} + (-a+c)\frac{(\Delta x)^3}{3!}\frac{\partial^3 u}{\partial x^3} + \ldots \quad (6) \end{aligned}$$

Since we want (6) to have the form of $\frac{\partial u}{\partial x} + O(\Delta x)^m$, therefore we set

(a+b+c) =0,
(-a+c)Δx=1
a+c=0.

From them we can find b=0, c = -a = 1/(2Δx), therefore

$$\frac{\partial u}{\partial x} = \frac{u_{i+1} - u_{i-1}}{2\Delta x} + O(\Delta x)^3$$

which is a second-order centered difference scheme.

This method can be used in a general manner for symmetric or non-symmetric difference formula. You just pick which points you want to use.