CLIM-715
Numerical Methods for Climate Modeling

(Finite Difference Methods)

Notes # 5

Finite Differences

- There are two conceptually different ways to represent continuous functions on digital computers
  - As a finite set of grid points or
  - As a finite set of series-expansion functions
- The grid-point approach is used in conjunction with finite-difference methods, the most widely used technique
- Time dependence is almost always discretized and treated with finite differences.

Finite Differences

- Basic premise of finite differences is that the concept of derivative is based on a limit of the form

$$\frac{df}{dx}(x_0) = \lim_{\Delta x \to 0} \frac{y(x_0 + \Delta x) - y(x_0)}{\Delta x}$$ Right, downstream

$$\frac{df}{dx}(x_0) = \lim_{\Delta x \to 0} \frac{y(x_0) - y(x_0 - \Delta x)}{\Delta x}$$ Left, upstream

$$\frac{df}{dx}(x_0) = \lim_{\Delta x \to 0} \frac{y(x_0 + \Delta x) - y(x_0 - \Delta x)}{2\Delta x}$$ Center

- If the derivative of *f(x)* is continuous at x_0, all three expressions produce the same unique answer
- If we assume that the limit is reached in a "smooth" way, then there is some range of Δx for which the derivative is closely approximated by the difference

Finite Differences

- When Δx is finite, the finite-difference approximations are not equivalent

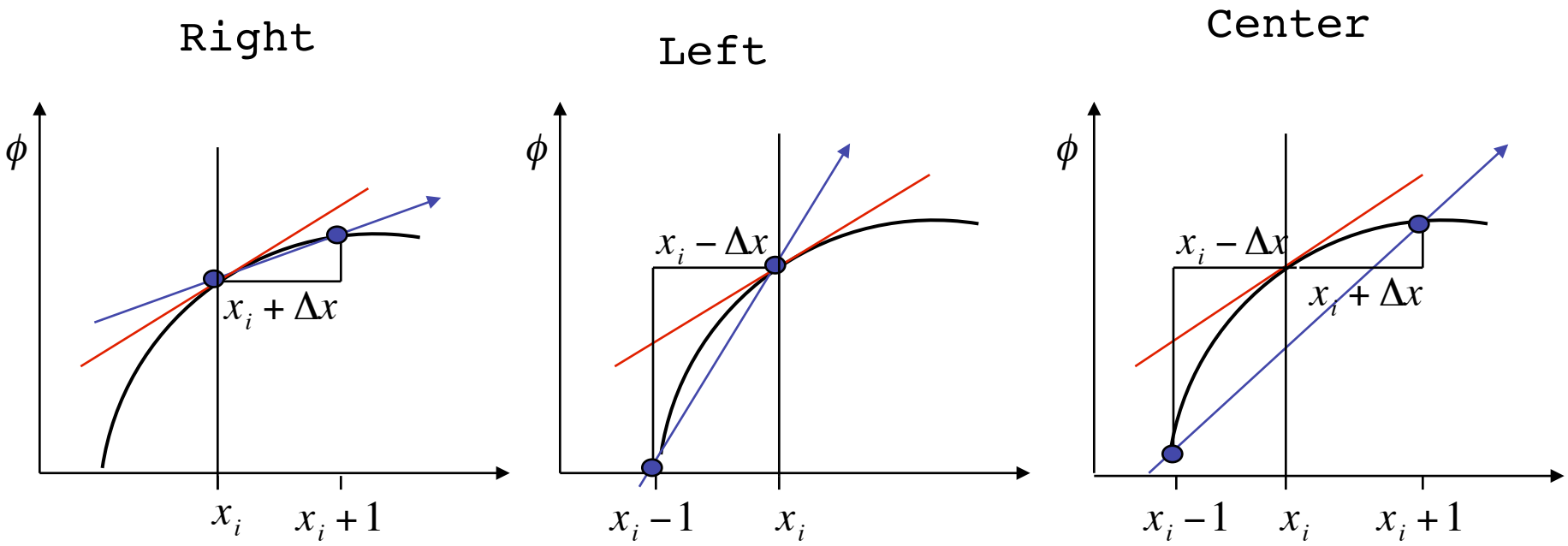
$$\frac{df}{dx}(x_0) = \lim_{\Delta x \to 0} \frac{y(x_0 + \Delta x) - y(x_0)}{\Delta x}$$ Right, downstream

$$\frac{df}{dx}(x_0) = \lim_{\Delta x \to 0} \frac{y(x_0) - y(x_0 - \Delta x)}{\Delta x}$$ Left, upstream

$$\frac{df}{dx}(x_0) = \lim_{\Delta x \to 0} \frac{y(x_0 + \Delta x) - y(x_0 - \Delta x)}{2\Delta x}$$ Center

- They differ in their accuracy and structure
- The differences in the structures of these algebraic equations can have a great influence on the stability of the numerical solution

Derivative with Finite Differences



- Blue line - estimate of derivative, red line - true derivative
- Discretizing space creates a grid spacing Δx
- Therefore, we introduce errors that should increase as Δx get larger.
- However, as Δx go to zero, our numerical solution should convergence on the correct solution.

Derivative with Finite Differences

- The exact derivative can be calculated to within an arbitrarily small error by insuring that Δx is sufficiently small
- But computers put restriction

Properties

- Accuracy
- Consistency
- Stability
- Convergence

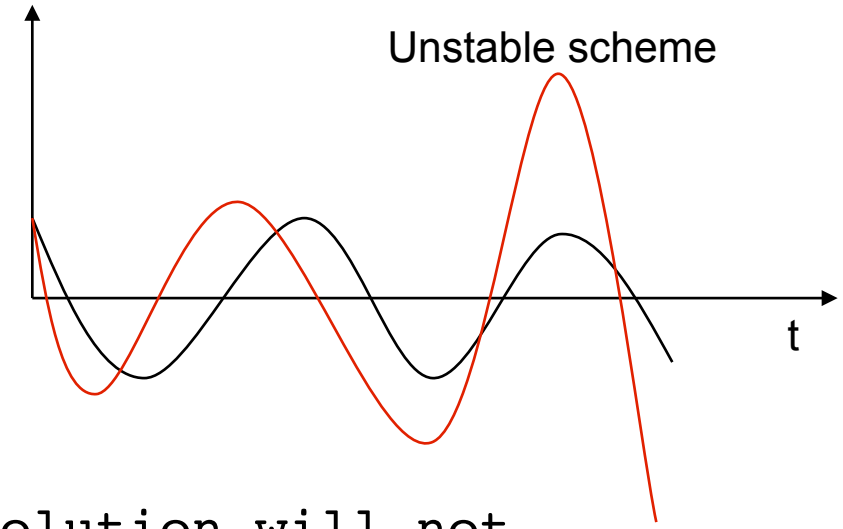
Properties

- Accuracy
  - How well does finite difference expression approximate the derivative?
  - How big is the difference between the expression and the derivative it is attempting to approximate?
- Consistency
- Stability
- Convergence

Properties

- Accuracy
- Consistency
  - Consistency means we are solving the correct problem
  - Does the approximation go to the true limit as the mesh spacing goes to zero?
- Stability
- Convergence

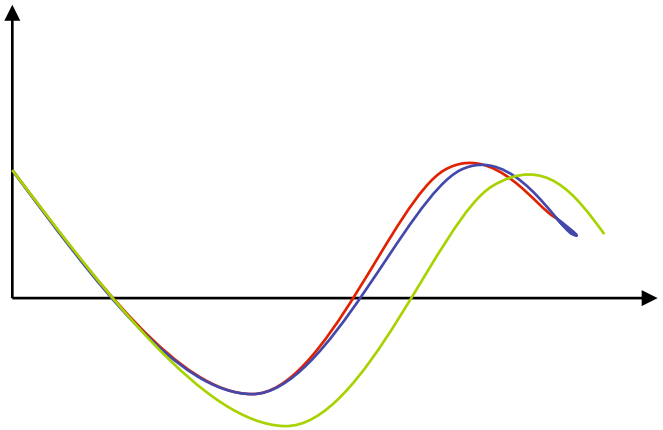
Properties



- Accuracy
- Consistency
- Stability
  - Stability means the solution will not grow unbounded with time.
  - If a scheme solves a time-integration problem, then does the solution for a long (infinite) time remain bounded, provided that the differential equation remains bounded?
- Convergence

Properties

- Accuracy
- Consistency
- Stability
- Convergence
  - The equivalence (Lax Richtmyer, 1956) theorem tells us that convergence is assured if the numerical scheme is both consistent & stable
  - So, convergence related to stability, but not the same
  - A scheme is convergent if integration to the same time with finer and finer resolution gives an answer that approaches to a limit



Accuracy

- Start with Taylor series
- Expand the terms like $f(x_0 \pm \Delta x)$ in Taylor series about x_0 and substituting these expressions into the finite-difference formula (e.g., into downstream scheme)

$$f\left(x_0+\Delta x\right)=f\left(x_0\right)+\Delta x\frac{df}{dx}\left(x_0\right)+\frac{\left(\Delta x\right)^2}{2!}\frac{d^2f}{dx^2}\left(x_0\right)+\frac{\left(\Delta x\right)^3}{3!}\frac{d^3f}{dx^3}\left(x_0\right)+.....$$

$$\frac{f\left(x_0+\Delta x\right)-f\left(x_0\right)}{\Delta x}=\frac{df}{dx}\left(x_0\right)+\frac{\Delta x}{2!}\frac{d^2f}{dx^2}\left(x_0\right)+\frac{\left(\Delta x\right)^2}{3!}\frac{d^3f}{dx^3}\left(x_0\right)+.....$$

Accuracy

$$\frac{f(x_0+\Delta x)-f(x_0)}{\Delta x}=\frac{df}{dx}(x_0)+\frac{\Delta x}{2!}\frac{d^2f}{dx^2}(x_0)+\ldots\ldots$$

Approximation *Expression* *Error*

- As $\Delta x \to 0$, our error becomes smaller
- This error is know as "***Truncation Error***"
- It is the error associated by truncating the Taylor series

Accuracy

- Our error is reduced by half when the spacing is reduced by half
- Lowest power of Δx in the truncation error determines the order of accuracy of the finite difference
- The one-sided difference (downstream scheme) is "*First-Order*" accurate

$$\frac{f(x_0+\Delta x)-f(x_0)}{\Delta x}=\frac{df}{dx}(x_0)+O(\Delta x)$$

Accuracy

- For higher-order finite-difference approximations, follow the example in Chapter 2 of Durran
- The maximum order of a finite difference approximation to a derivative is one less than the number of terms in the stencil

Accuracy of Entire Scheme

- We now turn from the consideration of individual finite difference (FD) to examine the accuracy of an entire finite-difference scheme.
- We will examine the FD approximation to the simple 1D inviscid advection equation

$$\frac{\partial u}{\partial t} + c\frac{\partial u}{\partial x} = 0$$

$$u_t + cu_x = 0$$

- What is the accuracy of our entire scheme?

Accuracy

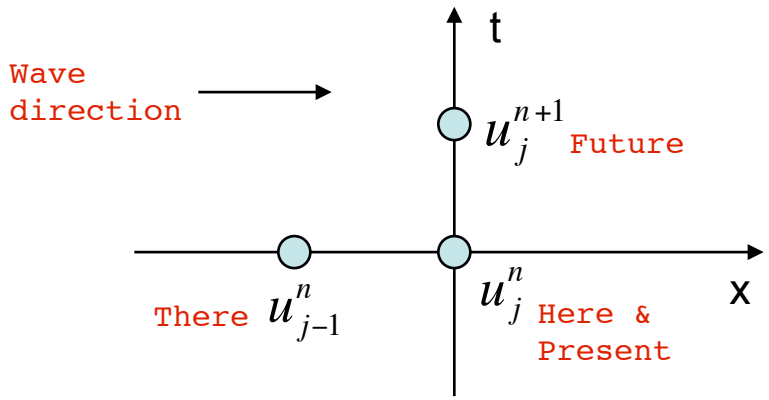
- Let us use one-sided “upstream” or “up-wind scheme” as our practical example, not because it is good (actually, *it is pretty bad*) but owing to its simplicity and pedagogical value

Accuracy

- The upstream approximation to 1D inviscid advection equation

$$u_t + cu_x = 0$$

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} + c\left[\frac{u_j^n - u_{j-1}^n}{\Delta x}\right] = 0$$

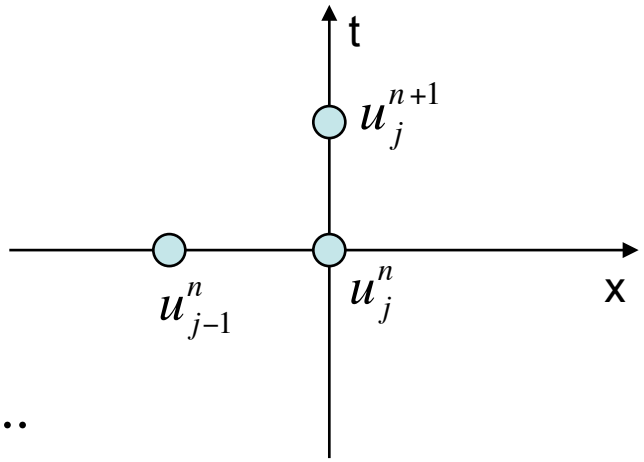


- "c" must be nonnegative in this example, so the wave is moving from left to right
- The upstream scheme is "*forward-in-time*" and "*upstream-in-space*"
- The upstream scheme is "*explicit*", because u_j^{n+1} is the only unknown
- What is the accuracy of our entire scheme?

Accuracy

- With Taylor Series expansions

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} + c\left[\frac{u_j^n - u_{j-1}^n}{\Delta x}\right] = 0$$



$$u_j^{n+1} = u_j^n + u_t \Delta t + u_{tt}\frac{(\Delta t)^2}{2!} + u_{ttt}\frac{(\Delta t)^3}{3!} + ...$$

$$u_{j-1}^n = u_j^n - u_x \Delta x + u_{xx}\frac{(\Delta x)^2}{2!} - u_{xxx}\frac{(\Delta x)^3}{3!} + ...$$

- How do you know how many terms to keep in the Taylor series?

Accuracy

- With Taylor Series expansions

$$u_t + cu_x = 0$$

$$\frac{1}{\Delta t}\left[u_j^n + u_t\Delta t + u_{tt}\frac{(\Delta t)^2}{2!} + u_{ttt}\frac{(\Delta t)^3}{3!} - u_j^n\right] + \frac{c}{\Delta x}\left[u_j^n + u_x\Delta x - u_{xx}\frac{(\Delta x)^2}{2!} + u_{xxx}\frac{(\Delta x)^3}{3!} - u_j^n\right] = 0$$

$$u_t + cu_x = \underbrace{-\frac{\Delta t}{2}u_{tt} + \frac{c\Delta x}{2}u_{xx} - \frac{(\Delta t)^2}{3!}u_{ttt} - \frac{c(\Delta x)^2}{3!}u_{xxx} + O\left[(\Delta t)^3, (\Delta x)^3\right]}_{\tau}$$

- According to above original PDE, RHS (truncation term, τ) should be zero
- But this is not likely to be true, at least for finite values of Δx and Δt

Accuracy

- For nonzero Δx and Δt, it is seen that we are NOT exactly solving

$$u_t + cu_x = 0$$

- Because the truncation error is nonzero

Accuracy

- For nonzero Δx and Δt, we can manipulate the truncation term (τ) to gain some physical insight on the problem we are actually solving

$$u_t + cu_x = \underbrace{-\frac{\Delta t}{2}u_{tt} + \frac{c\Delta x}{2}u_{xx} - \frac{(\Delta t)^2}{3!}u_{ttt} - \frac{c(\Delta x)^2}{3!}u_{xxx} + O\left[(\Delta t)^3, (\Delta x)^3\right]}_{\tau}$$

- First perform $\frac{\partial}{\partial t}$ to the above equation to yield

$$u_{tt} + cu_{xt} = -\frac{\Delta t}{2}u_{ttt} + \frac{c\Delta x}{2}u_{xxt} - \frac{(\Delta t)^2}{3!}u_{tttt} - \frac{c(\Delta x)^2}{3!}u_{xxxt} + O\left[(\Delta t)^3, (\Delta x)^3\right]$$

Accuracy

- Then perform $-c\frac{\partial}{\partial x}$ to the above equation to yield

$$-cu_{tx} - c^2 u_{xx} = \frac{c\Delta t}{2} u_{ttx} - \frac{c^2 \Delta x}{2} u_{xxx} + \frac{c(\Delta t)^2}{3!} u_{tttx} + \frac{c^2 (\Delta x)^2}{3!} u_{xxxx} + O\left[(\Delta t)^3, (\Delta x)^3\right]$$

- Add the last two expressions together and solve for

$$u_{tt} = c^2 u_{xx} + \Delta t\left[-\frac{u_{ttt}}{2} + \frac{c}{2} u_{ttx}\right] + \Delta x\left[\frac{c}{2} u_{xxt} - \frac{c^2}{2} u_{xxx}\right] + O\left[(\Delta t)^2, (\Delta x)^2\right]$$

- Similarly, the details and explicitly keeping fewer terms, we can produce and substitute into the above equation

$$u_{ttt} = -c^3 u_{xxx} + O\left[\Delta t, \Delta x\right]$$

$$u_{ttx} = c^2 u_{xxx} + O\left[\Delta t, \Delta x\right]$$

$$u_{xxt} = -c\ u_{xxx} + O\left[\Delta t, \Delta x\right]$$

Accuracy

- And then if we plug the result into the following equation,

$$u_t + cu_x = \underbrace{-\frac{\Delta t}{2}u_{tt} + \frac{c\Delta x}{2}u_{xx} - \frac{(\Delta t)^2}{3!}u_{ttt} - \frac{c(\Delta x)^2}{3!}u_{xxx} + O\left[(\Delta t)^3, (\Delta x)^3\right]}_{\tau}$$

- We can obtain a new expression for the truncation error on the RHS

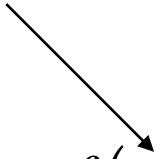
$$u_t + cu_x = \frac{c\Delta x}{2}\left[1 - \frac{c\Delta t}{\Delta x}\right]u_{xx} + O\left[(\Delta t)^2, (\Delta x)^2\right]$$

- For nonzero truncation term, the equation we are actually solving numerically is

$$u_t + cu_x = f\left(u_{xx}\right)$$

Accuracy

- Note that the leading part of the truncation error is a diffusion term

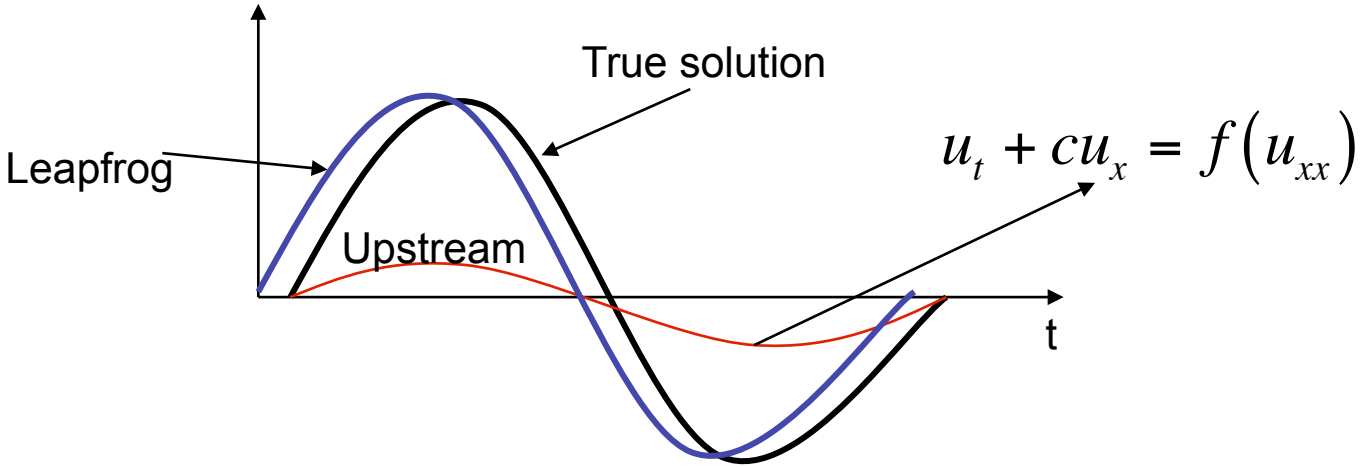


$$u_t + cu_x = f\left(u_{xx}\right)$$

- This shows that the fatal flaw of the "upstream" scheme is that it contains large implicit diffusion
- Integrated for a sufficiently long period, the upstream scheme will diffuse any initial perturbation away, which makes it useless for long term integrations

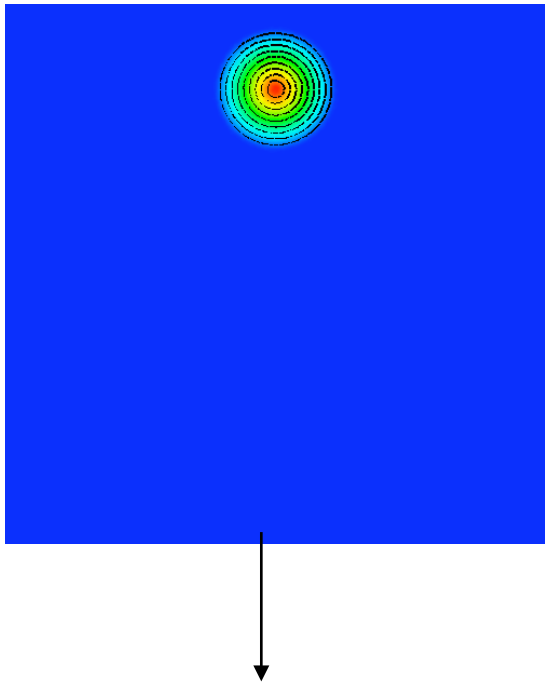
Accuracy

- A qualitative look at the performance of the upstream scheme

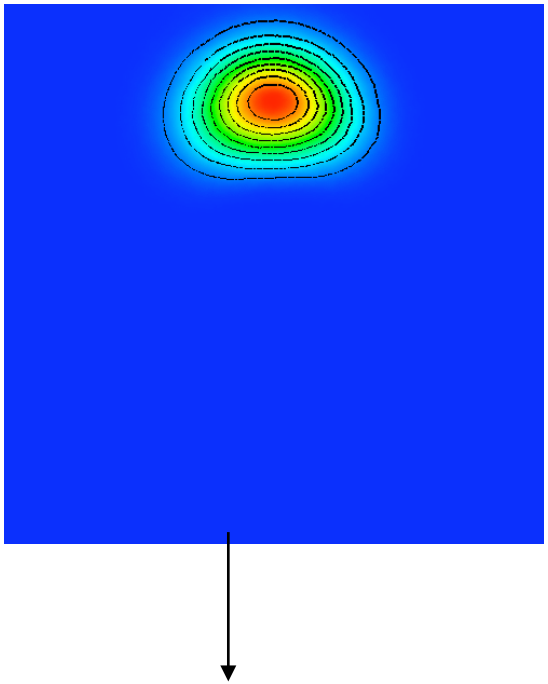


- Owing to the extreme inherent diffusiveness of the upstream scheme, the amplitude has been greatly damped
- However, the wave is in about the right place, it has been advected at about the right speed

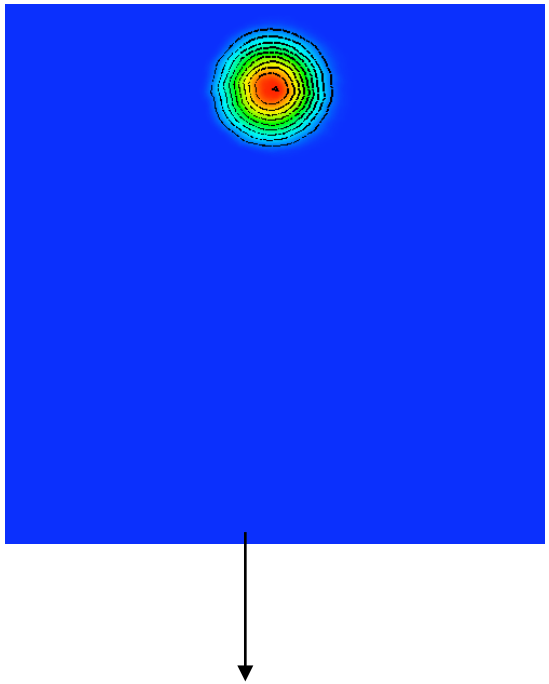
Rotating Cone Test



Initial Condition



1st-order solution exhibits large numerical diffusion



2nd-order solution retains over 90% of the initial tracer maxima. There are no dispersion and phase error

Consistency

- Consistency means we are solving the correct problem
- Does the approximation go to the true limit as the mesh spacing goes to zero?

$$u_t + cu_x = \underbrace{-\frac{\Delta t}{2}u_{tt} + \frac{c\Delta x}{2}u_{xx} - \frac{(\Delta t)^2}{3!}u_{ttt} - \frac{c(\Delta x)^2}{3!}u_{xxx} + O\left[(\Delta t)^3, (\Delta x)^3\right]}_{\tau}$$

$$u_t + cu_x = \frac{c\Delta x}{2}\left[1 - \frac{c\Delta t}{\Delta x}\right]u_{xx} + O\left[(\Delta t)^2, (\Delta x)^2\right]$$

- $\tau \to 0$ as Δt and $\Delta x \to 0$, so the scheme is *consistent*
- Although it is not difficult to design consistent difference schemes, this property should not be taken for granted
- Sometimes methods requires additional relationship

Stability & Convergence

- The difference between a finite difference approximation and a derivative (the accuracy) does NOT describe the difference between a numerical solution and the true solution
- Let us denote the numerical solution to ϕ_j^n at $t = n\Delta t, x = j\Delta x$
- Let us denote the true solution to the equation as $\psi(n\Delta t, j\Delta x)$
- For stability, we want to know the difference between the two at some long time

Norm

- If we want to know the difference between two "things", we need to establish a "norm" (a way of to measure its size which is the most direct measure of the quality of the numerical solution)
- We denote a norm of a with $\| a \|$

Maximum norm (Max of the grid point values:

$$\| \phi \|_\infty = \max_{1 \le j \le N} | \phi_j | \qquad \| \phi \|_\infty \approx \left(\int | \phi |^\infty \, dx \right)^{1/\infty}$$

ℓ_2 norm (the length of N dimensional vector):

$$\| \phi \|_2 \equiv \left(\sum_{j=1}^{N} | \phi_j |^2 \, \Delta x \right)^{1/2} \qquad \| \phi \|_2 \approx \left(\int | \phi |^2 \, dx \right)^{1/2}$$

Order & Convergence

- The maximum norm is the most natural one to compute when working with grid-point values
- However, ℓ_2 norm is also useful, because it is more closely related to conserved quantities, such as total energy
- A finite difference scheme is convergent to order (p,q) if in the limit $\Delta t, \Delta x \to 0$

$$\| \psi(n\Delta t, j\Delta x) - \phi_j^n \| = O(\Delta t^p) + O(\Delta x^q)$$

- The relationship between convergence and consistency is described by the **Lax equivalence theorem**

Order & Convergence

- Lax Equivalence Theorem
  - The Lax equivalence theorem tells us that convergence is assured if the numerical scheme is both consistent & stable
  - So, convergence related to stability, but not the same
  - A scheme is convergent if integration to the same time with finer and finer resolution gives an answer that approaches to a limit
  - If a finite-difference scheme is linear, stable, and accurate of order (p,q), then it is convergent of order (p,q)

Accurate of order: As before

Convergent: As before

What about stability?

Stability

- In other words, there are lots of schemes that are linear, accurate to first order or better, but are not stable
- So no matter how fine the time step, you can not be sure that the solution after some short fixed time will be any good

- The fundamental definition of stability has no reference to the property of the true solution!
- Scheme is "unstable" if it generates a numerical solution ϕ_j^n that grows much more rapidly than the true solution $\psi(n\Delta t, j\Delta x)$
- It is only the least-restrictive additional constraint that we can apply to ensure the convergence of solutions (i.e., stable system)

Stability

- For the Lax equivalence theorem to be valid, we need

$$\| \phi^n \| \leq C_T \| \phi^0 \| \qquad \text{for all} \quad n\Delta t \leq T$$

- In this expression, C_T (a constant) may depend on for the time T, but not on $\Delta t, \Delta x$ or the number of time steps n
- This permits solutions that grow with time (since C_T may grow), but it rules out solutions that grow as a function of the number of time steps
- If a scheme is unstable, it fails to satisfy the above relationship

Stability

- This says nothing about the quality of the solution that might be obtained when using finite values of $\Delta t, \Delta x$

- It only ensures that an accurate solution will be obtained in the limit $\Delta t, \Delta x \to 0$
- Schemes that are stable according to this criteria may also generate solutions that "blow up"
- In many applications, the norm of the true solution is constant with time, i.e.,

$$\| \phi^n \| \leq \| \phi^0 \| \qquad \text{for all n}$$

- This criteria guarantees that the solution will not blow up

Stability

- It is relatively easy to formulate consistent difference schemes and to determine their truncation error and order of accuracy
- The analysis of stability can, however, be far more difficult
- First we will focus on simple case, linear finite-difference schemes with constant coefficients
- Later we will examine nonlinear and linear schemes with variable coefficients

Energy Method

- Energy method is used less frequently
- This method can be applied to nonlinear equations
- Basic idea - to find a positive definite quantity like

$$\sum_j \left(\phi_j^2\right)^2$$

- And to show that this quantity is bounded for all *n*
- If $\sum_j \left(\phi_j^2\right)^2$ is bounded, then the solution is stable with respect to ℓ_2 norm (i.e., $\ell_2$is also bounded)

Energy Method

- Example, let us investigate stability of upstream scheme

$$\frac{\phi_j^{n+1} - \phi_j^n}{\Delta t} + c\left[\frac{\phi_j^n - \phi_{j-1}^n}{\Delta x}\right] = 0$$

$$\mu = c\Delta t / \Delta x$$

$$\phi_j^{n+1} = (1-\mu)\phi_j^n + \mu\phi_{j-1}^n$$

Energy Method

- Square both sides and sum over all *j*

$$\sum_j \left(\phi_j^{n+1}\right)^2 = \sum_j \left[(1-\mu)^2\left(\phi_j^n\right)^2 + 2\mu(1-\mu)\phi_j^n\phi_{j-1}^n + \mu^2\left(\phi_{j-1}^n\right)^2\right]$$

- If we can assume periodic boundary conditions, then

$$\sum_j \left(\phi_{j-1}^n\right)^2 = \sum_j \left(\phi_j^n\right)^2$$

- So $$\sum_j \left(\phi_j^{n+1}\right)^2 = \sum_j \left[(1-2\mu(1-\mu))\left(\phi_j^n\right)^2 + 2\mu(1-\mu)\phi_j^n\phi_{j-1}^n\right]$$

$$\sum_j \left(\phi_j^{n+1}\right)^2 = \sum_j \left(\phi_j^n\right)^2 - 2\mu(1-\mu)\sum_j \left[\left(\phi_j^n\right)^2 - \phi_j^n\phi_{j-1}^n\right]$$

Energy Method

$$\sum_j \left(\phi_j^{n+1}\right)^2 = \sum_j \left(\phi_j^n\right)^2 - 2\mu(1-\mu)\sum_j \left[\left(\phi_j^n\right)^2 - \phi_j^n \phi_{j-1}^n\right]$$

- Now use Schwardz inequality:

  for two vectors, the magnitude of the dot product $|u.v|$ is smaller than the product of the ℓ_2 norms:

$$|u.v| \le |u.u|^{1/2} |v.v|^{1/2}$$

so

$$\sum_j \phi_j^n \phi_{j-1}^n \le \left[\sum_j \left(\phi_j^n\right)^2\right]^{1/2} \left[\sum_j \left(\phi_{j-1}^n\right)^2\right]^{1/2} = \sum_j \left(\phi_j^n\right)^2$$

and the last term above is positive

Energy Method

- As long as $\mu(1-\mu) \geq 0$
- Then

$$\sum_j \left(\phi_j^{n+1}\right)^2 \leq \sum_j \left(\phi_j^n\right)^2 \longrightarrow \| \phi^{n+1} \|_2 \leq \| \phi^0 \|_2$$

- This implies that the scheme is "stable"
- If $\mu > 0$, the sufficient condition $\mu(1-\mu) \geq 0$, for stability leads to $\mu \leq 1$
- Therefore the total constraint on μ is

$$0 \leq \mu \leq 1$$

- The stability condition may then be written

$$0 \leq c\frac{\Delta t}{\Delta x} \leq 1$$

Energy Method

- There is a maximum limit on the time step beyond which scheme is unstable, and the stability limit becomes more severe as the spatial resolution is increased

$$0 \le c\frac{\Delta t}{\Delta x} \le 1$$

Von Neumann's Method

- Von Neumann's method is simpler, but applies to a more restricted set of problems, i.e., to linear finite-difference equations with constant coefficients

- Basic idea: to represent the discretized solution at some particular time step by a finite Fourier series of the form

$$\phi_j^n = \sum_{k=-N}^{N} a_k^n e^{ikj\Delta x}$$

- The total solution will be stable if and only if every Fourier component is stable

Von Neumann's Method

- Now isolate a single wavenumber

$$\phi_j^n = a_k^n e^{ikj\Delta x}$$

- What are the values at neighboring points?

$$\phi_{j+m}^n = a_k^n e^{ik(j+m)\Delta x} = \phi_j^n e^{ikm\Delta x}$$

- A difference of the form

$$\frac{d\phi}{dx} = a\phi_{j+2} + b\phi_{j+1} + c\phi_j + f\phi_{j-1} + g\phi_{j-2}$$

- Can be written as

$$\frac{d\phi}{dx_j} = C\phi_j$$

Eigenfunctions of the derivative

- Where

$$C = ae^{i2k\Delta x} + be^{ik\Delta x} + c + fe^{-ik\Delta x} + ge^{-i2k\Delta x}$$

Von Neumann's Method

- The typical finite difference scheme can be written

$$a_k^{n+1} = A_k a_k^n$$

- Recall *a* is a Fourier coefficient (amplitude of the *kth* finite Fourier component at the *nth* time step)
- A_k is the amplification factor (it can be complex)
- Over long times, we have

$$a_k^n = A_k a_k^{n-1} = (A_k)^n a_k^0$$

- It follows that the stability of each Fourier component is determined by the modules of its amplification factor

Von Neumann's Method

- The von Neumann stability criterion, which is necessary and sufficient for the stability of a linear constant-coefficient finite difference equation

$$|A_k| \leq 1 + \gamma \Delta t$$

- This criterion assures that with finer and finer time steps, the solutions may grow, but only according to the Lax criterion
- The more convenient criterion for solutions that are known to be bounded is

$$|A_k| \leq 1$$

- When the von Neumann condition is satisfied, the full solution is stable

Von Neumann's Method

- Example: consider our earlier problem

$$\phi_j^{n+1} = (1-\mu)\phi_j^n + \mu\phi_{j-1}^n$$

$$A_k e^{ikj\Delta x} = (1-\mu)e^{ikj\Delta x} + \mu e^{ik(j-1)\Delta x}$$

$$A_k = 1 - \mu + \mu e^{-ik\Delta x}$$

- To find the magnitude of A_k, multiply by itself and take the square root

$$|A_k|^2 = (1-\mu+\mu e^{-ik\Delta x})(1-\mu+\mu e^{-ik\Delta x}) = 1 - 2\mu(1-\mu)(1-\cos k\Delta x)$$

- So our system will be stable for wavenumber *k* if

$$1 - 2\mu(1-\mu)(1-\cos k\Delta x) \le 1$$

Von Neumann's Method

$$1 - 2\mu(1-\mu)(1-\cos k\Delta x) \le 1$$

- A common task with the von Neumann method is to find the limits of various expressions over the entire range of wavenumbers
- The wave numbers that can be resolved on the grid are from k=-N…N where

$$N = \frac{\pi}{\Delta x}$$

- Over this range $(1-\cos k\Delta x) > 0$ for all k, and our criterion is identical to the criterion obtained using the energy method

$$\mu(1-\mu) \ge 0$$

- Can be expressed as

$$0 \le c\frac{\Delta t}{\Delta x} \le 1$$

Von Neumann's Method

$$0 \le c\frac{\Delta t}{\Delta x} \le 1$$

- This often termed the Courant-Fredrichs-Lewy (CFL) criterion.
- Some physical insight can be gained from it

$$s = \frac{\Delta x}{c\Delta t}$$

- This can be thought of as a "*sampling rate*"
- s=1 means the wave propagating at speed c will move one grid point Δx in one time step Δt
- Sampling too slowly yields instability; solution blows up!
- Intuition tells you that a higher sampling rate should be an improvement. Actually sampling more frequently is not always a good idea

Von Neumann's Method

- If $\mu = 1$, then $|A_k| = 1$, and there is no amplitude error (solution neither grows nor decays)
- If higher sampling rate is chosen, such that $\mu < 1$, then $|A_k| < 1$, and solution decays with time
- For $\mu \neq 1$, the amplitude will either grow or decay for all waves, because $|A_k| \neq 1$
- However, all waves are not affected equally, due to $\cos k\Delta x$ term, $|A_k|$ varies with wavenumber k
- This turns out that shortest waves always handled worst

Von Neumann's Method

- The smallest resolvable wave has a wavelength of

$$L_x = 2\Delta x$$

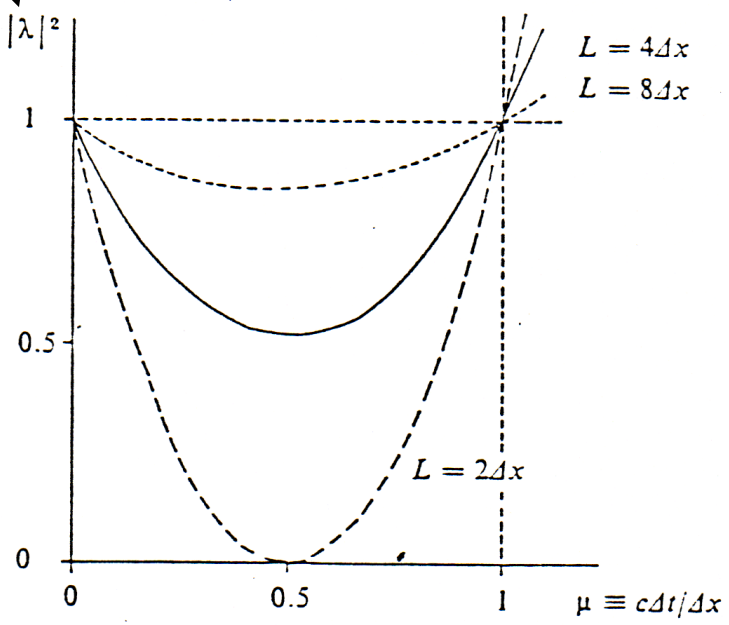
- Since $k = 2\pi/L_x$, $k\Delta x = \pi$ for $2\Delta x$ wave
- Therefore $\cos k\Delta x = \cos\pi = -1$

- In this case, $|A_k|^2 = 1 - 4\mu(1-\mu)$
- So, If $\mu > 1$, then $|A_k|^2 > 1$, the amplitude will grow swiftly with time
- Due to the presence of $\cos k\Delta x$ in the equation, it is the shortest waves that will amplify the fastest

Von Neumann's Method

- The equation suggest that $|A|^2$ curves are paraboles
- Recall that the minimum resolvable wavelength is $2\Delta x$
- As the wavelength, L, increases, the amplification factor approaches unity for any value of μ
- Within the stable regime, the scheme is damping for all values $\mu < 1$
- The damping increases as the wave length decreases

$$|A_k|^2 = 1 - 2\mu(1-\mu)(1-\cos k\Delta x)$$

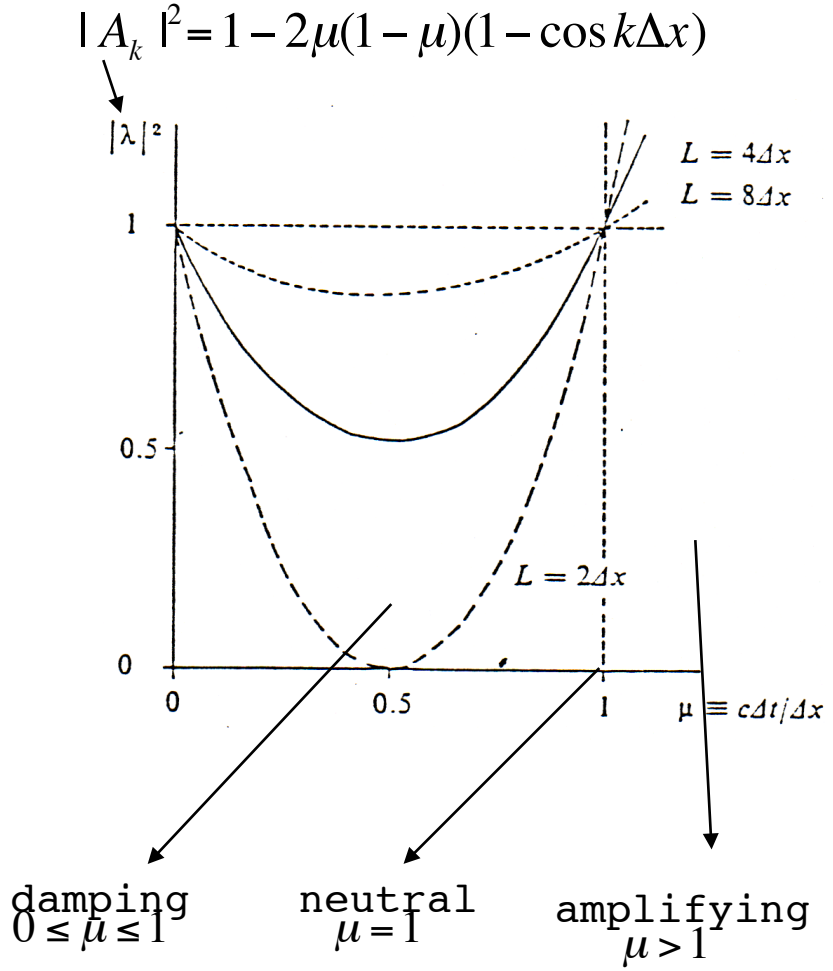


$$0 \le \mu \le 1$$

$$0 \le c\frac{\Delta t}{\Delta x} \le 1$$

Von Neumann's Method

- Since the true solution has a constant amplitude, this damping reveals an error due to finite differencing
- At $2\Delta x$, the error may be very great unless Δt is very small
- It is even possible for this wave to be completely removed after only a single time step!
- As the wavelength increases, the representation by a finite difference grid improves



Von Neumann's Method

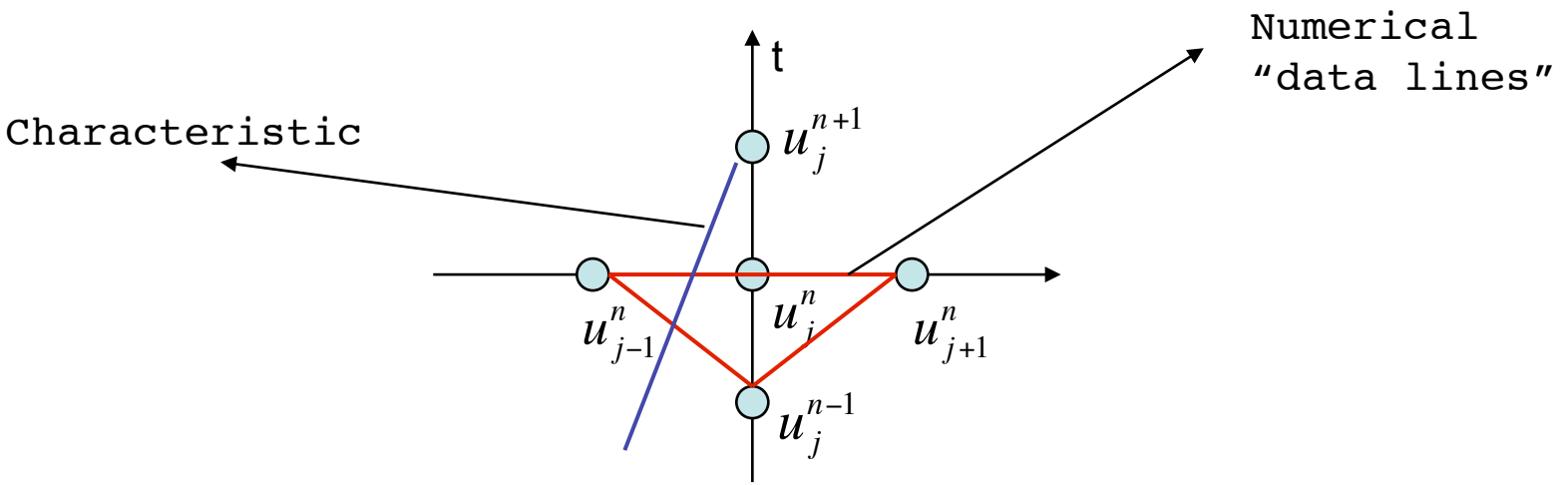
- The lesson to be learned is that the shortest waves tend to be mishandled, which implies
  - Do not invest much importance into these waves owing to their poor handling and
  - Make sure you choose your grid spacing to properly resolve the phenomenon you are trying to advect
  - For example, if your primary wave of interest is 10 km wide, choose Δx on the order of 1 km, so your feature is $10\Delta x$

Courant-Fredrichs-Levy (CFL) Condition

- The CFL stability criterion approaches the problem of stability in a different way:
  - We use the properties of solutions of PDEs along characteristics
- Recall that the solution at some point x and time t could be known if the characteristics that go through the point have intersected a data line
  - But we could NOT know the solution if the characteristics never touch a data line

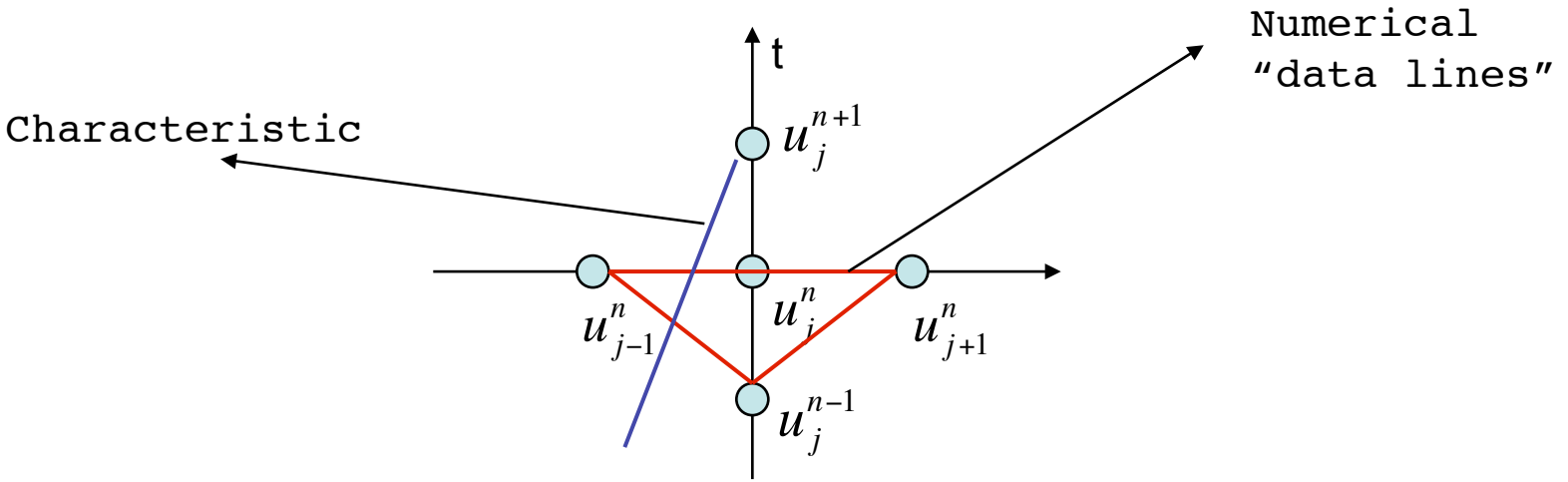
Courant-Fredrichs-Levy (CFL) Condition

- In a finite difference scheme, the stencil of points involved in the solution can be laid out in space and time
- We then treat data line(s) as thought they extend between all the points in the stencil



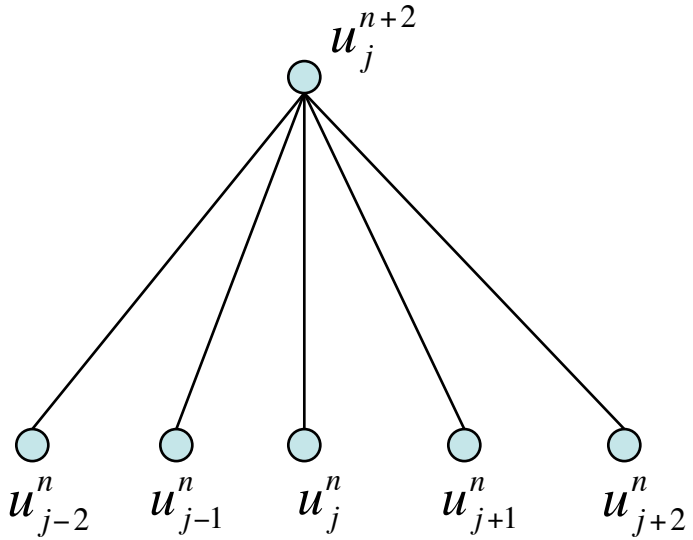
Courant-Fredrichs-Levy (CFL) Condition

- The CFL condition states that the characteristic defining the true solution at a grid point must cross a numerical line (i.e., inside the domain of dependence of the numerical solution at that point)
- That is, when $c\Delta t \le \Delta x$



Courant-Fredrichs-Levy (CFL) Condition

- Fourth order scheme has wider domain of dependence



Courant-Fredrichs-Levy (CFL) Condition

The CFL Condition does not guarantee stability. It only guarantees instability if violated.