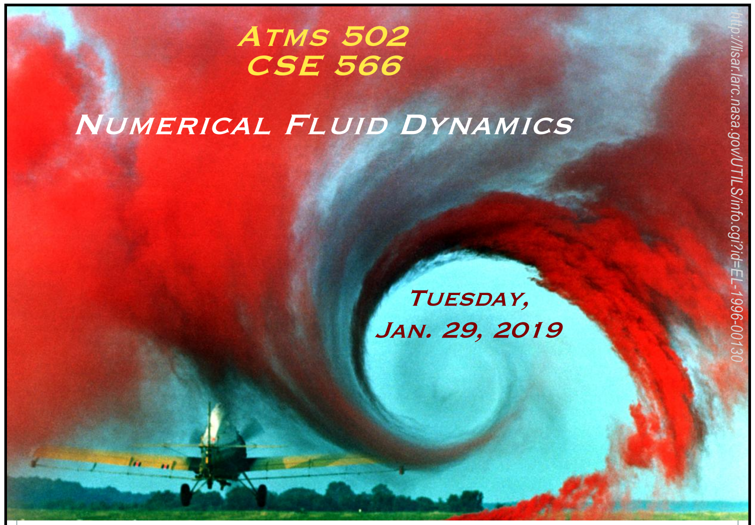


ATMS 502
CSE 566

NUMERICAL FLUID DYNAMICS

TUESDAY,
JAN. 29, 2019

http://lisar.larc.nasa.gov/UTILS/info.cgi?id=EL-1996-00130

ATMS 502 - Spring 2019 1 1/29/19

2

ATMS 502
CSE 566

Thursday,
29 January 2019

Class #5

Plan for Today

- 1) Review
  Truncation error & consistency
- 2) NUMERICAL METHODS :
  Takacs paper
  Staggered grids
  Changing from 1D to 2D
- 3) CODE/DATA:
  Program #2
- 4) NUMERICAL METHODS :
  Stability and polar plots

ATMS 502 - Spring 2019 1/29/19

Truncation error and Consistency

3

OBJECTIVES:

- *QUANTIFY THE DISCRETIZATION ERROR;*
- *DETERMINE IF A SCHEME IS CONSISTENT;*
- *LEARN TO ANTICIPATE ERROR CHARACTERISTICS.*

ATMS 502 - Spring 2019 1/29/19

Truncation error - again

4

- The process
  - Substitute Taylor series into scheme – for time, space separately
    - q(n+1) = q(n) + (Δt)*dq/dt + (Δt)^2/(2!)* ...
    - q(j+1) = q(j) + (Δx)*dq/dx + (Δx)^2/(2!)* ...
  - Keep like terms together (time vs. space) until canceling
  - Find original *PDE* terms, take to left side; everything else right
    - if the original PDE terms are not there: not a *consistent* approximation
    - everything on right hand side (RHS) is the *truncation error*
    - use *power of* Δx, Δt to identify the *order of accuracy*
- Example #1: *diffusion*
  - $q_t=K\bullet q_{xx}$ ... is $O(\Delta t, \Delta x^2)$ = 1st order accuracy in T, 2nd in X

ATMS 502 - Spring 2019 C008: Truncation error 1/29/19

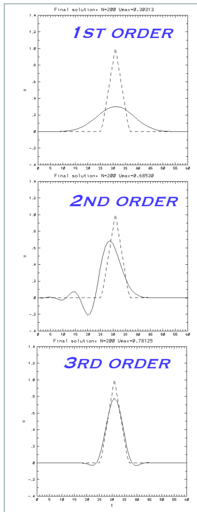
Truncation error: Derivation

$$\frac{q_j^{n+1}-q_j^n}{\Delta t}=K\frac{\left(q_{j+1}^n-2q_j^n+q_{j-1}^n\right)}{\Delta x^2}$$

Numerical methods: Takacs (1985)

- ORDER OF (SPATIAL) ACCURACY
- DOMINANT TYPES OF ERRORS
- HIS METHOD (USED LATER)

References:
- C001 (Lax-Wendroff)
- C006 (Finite differences)
- C007 (Taylor series)
- C052 (Advection)



Takacs pp. 1053-1054

- Amplitude error (Fig. 1)
  - θ is $k\Delta x$; μ is Courant number $c\Delta t/\Delta x$
  - Odd-order schemes most dissipative
  - Even-order schemes least dissipative
  - Higher-order schemes restrict errors to larger k
- Phase error (Fig. 2)
  - Odd order: smaller phase error
    - ✓ limited range of μ
  - Even order: larger phase error
    - ✓ over broader μ
  - Higher order: phase errors restricted to *larger* k

Forms of diffusion

- Behavior seen especially w/sharp gradients:

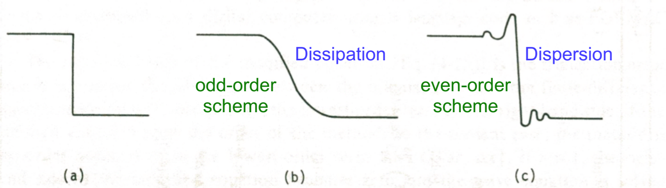


Figure 4-1 Effects of dissipation and dispersion. (a) Exact solution. (b) Numerical solution distorted primarily by dissipation errors (typical of first-order methods). (c) Numerical solution distorted primarily by dispersion errors (typical of second-order methods).

Anderson (1984)

Takacs (1985)

Takacs pp. 1055-1059

- Error computation
  - Total error is mean square error (6.1)

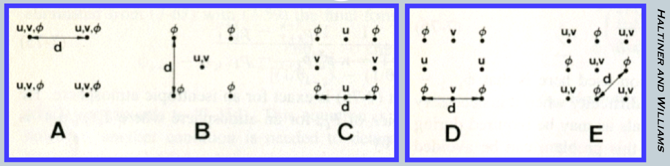
$$E_{TOT} = \frac{1}{N}\sum_{j}(q_T - q_D)^2$$

    - Dissipation error (6.6)
    - Dispersion error (6.7) = (total - dissipation), involves linear correlation coefficient ρ
    - If $\rho=1$, only error *is* due to dissipation
    - Test: cone (spike)
- *Least error* for $\alpha=(1+\mu)/6$ *(Figs. 5,7)*

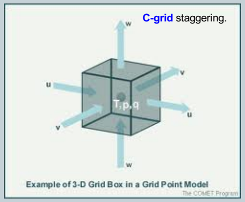
Numerical methods: Staggered grids

KEY ADVANTAGE:
IMPROVED PHASE BEHAVIOR

Staggered grids



- A-grid: No staggering
- C-grid: velocities on normal faces of mass points.
  - Different u,v locations; truncation errors - Coriolis terms
  - Better at higher resolution (than B-grid)



Non-orthogonal grids

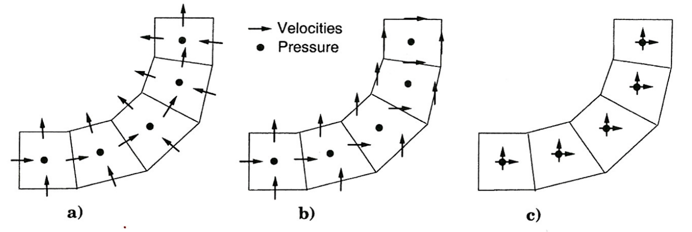
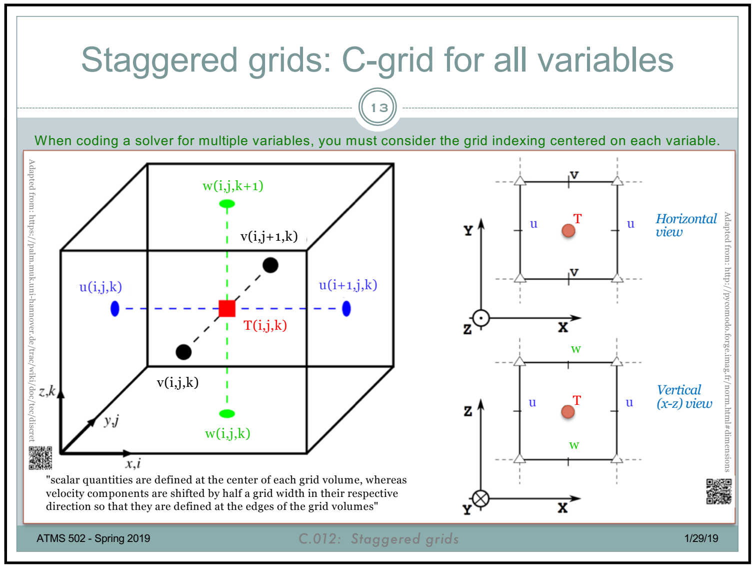


Fig. 8.5. Variable arrangements on a non-orthogonal grid: (a) – staggered arrangement with contravariant velocity components, (b) – staggered arrangement with Cartesian velocity components, (c) – colocated arrangement with Cartesian velocity components

Ferziger and Peric (2002), Chapter 8, *Complex Geometries*



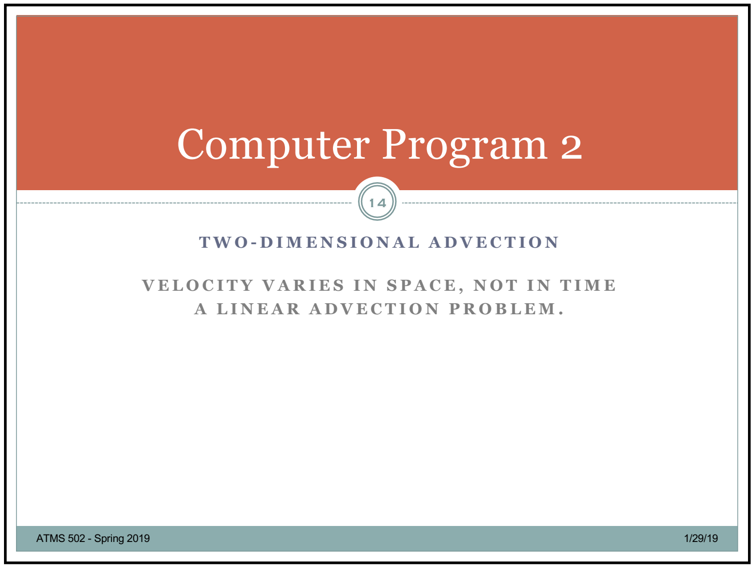
Staggered grids: C-grid for all variables

When coding a solver for multiple variables, you must consider the grid indexing centered on each variable.

Adapted from: https://palm.muk.uni-hannover.de/trac/wiki/doc/tec/discret

"scalar quantities are defined at the center of each grid volume, whereas velocity components are shifted by half a grid width in their respective direction so that they are defined at the edges of the grid volumes"

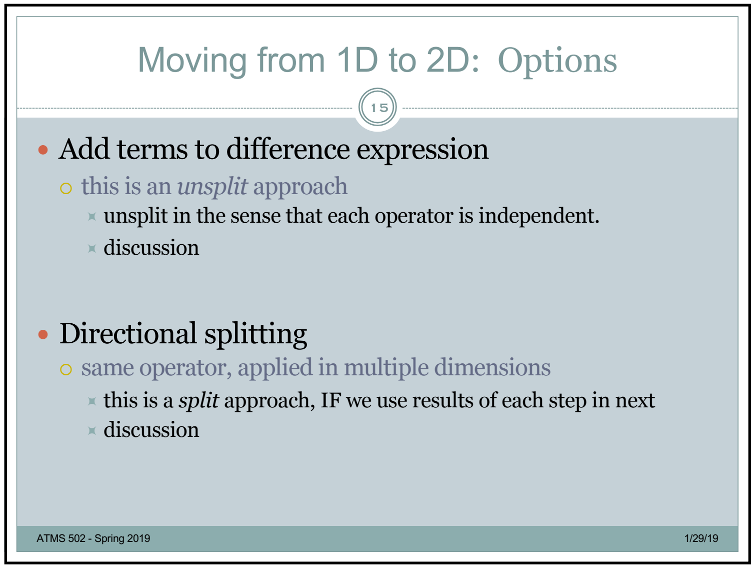
Adapted from: http://pycomodo.forge.imag.fr/norm.html#dimensions



Computer Program 2

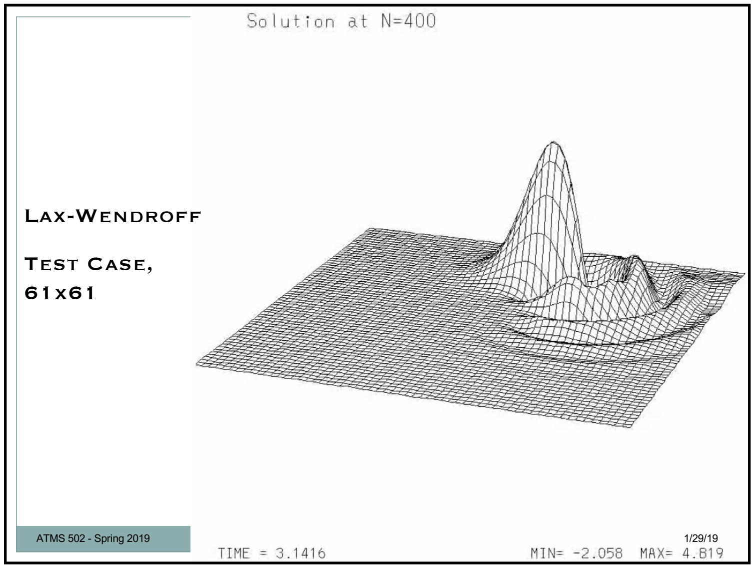
TWO-DIMENSIONAL ADVECTION

VELOCITY VARIES IN SPACE, NOT IN TIME
A LINEAR ADVECTION PROBLEM.



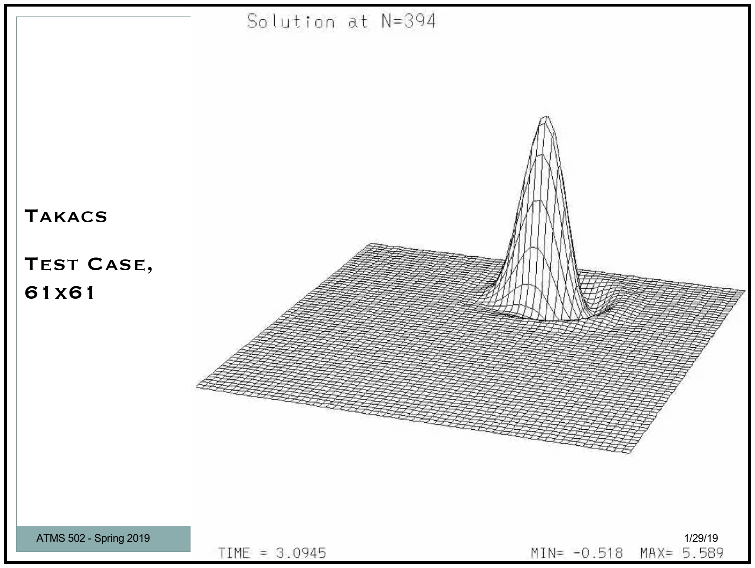
Moving from 1D to 2D: Options

- Add terms to difference expression
  - this is an *unsplit* approach
    - unsplit in the sense that each operator is independent.
    - discussion
- Directional splitting
  - same operator, applied in multiple dimensions
    - this is a *split* approach, IF we use results of each step in next
    - discussion



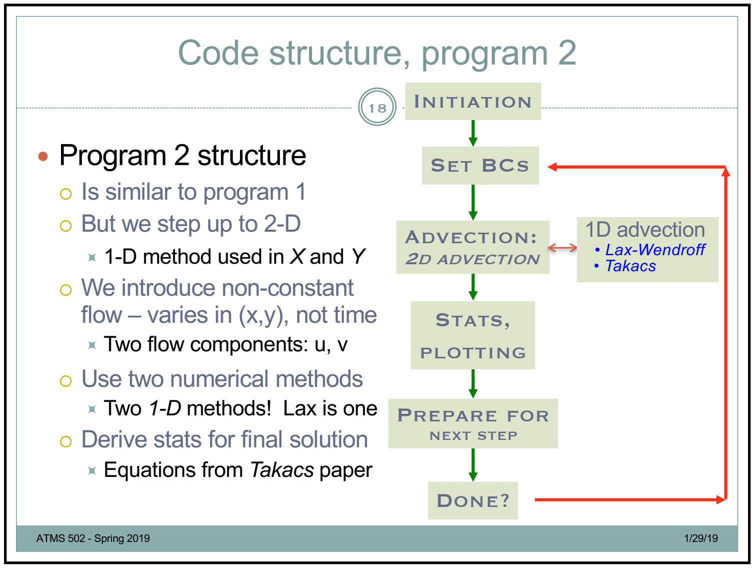
LAX-WENDROFF

TEST CASE,
61X61



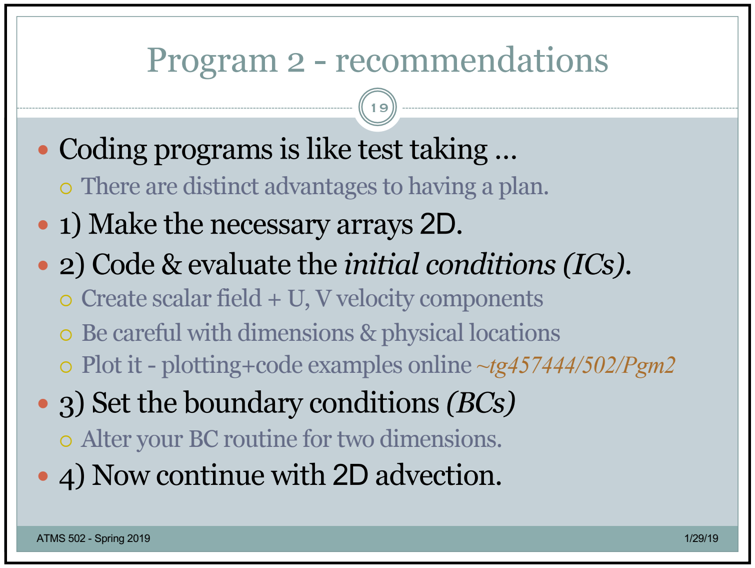
TAKACS

TEST CASE, 61X61



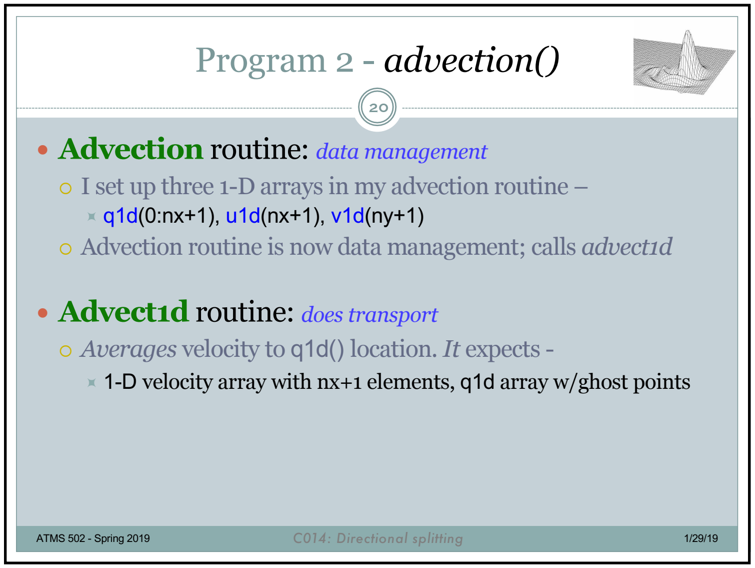
Code structure, program 2

- Program 2 structure
  - Is similar to program 1
  - But we step up to 2-D
    - 1-D method used in *X* and *Y*
  - We introduce non-constant flow – varies in (x,y), not time
    - Two flow components: u, v
  - Use two numerical methods
    - Two *1-D* methods! Lax is one
  - Derive stats for final solution
    - Equations from *Takacs* paper



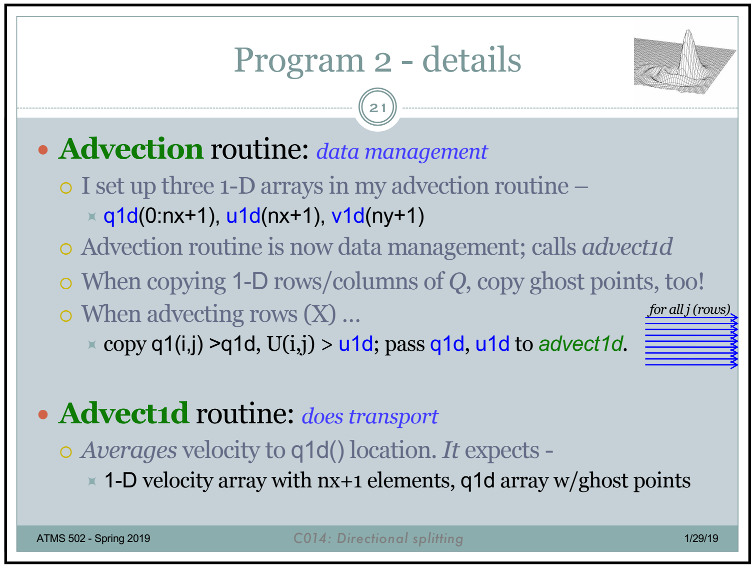
Program 2 - recommendations

- Coding programs is like test taking …
  - There are distinct advantages to having a plan.
- 1) Make the necessary arrays 2D.
- 2) Code & evaluate the *initial conditions (ICs)*.
  - Create scalar field + U, V velocity components
  - Be careful with dimensions & physical locations
  - Plot it - plotting+code examples online *~tg457444/502/Pgm2*
- 3) Set the boundary conditions *(BCs)*
  - Alter your BC routine for two dimensions.
- 4) Now continue with 2D advection.



Program 2 - *advection()*

- **Advection** routine: *data management*
  - I set up three 1-D arrays in my advection routine –
    - q1d(0:nx+1), u1d(nx+1), v1d(ny+1)
  - Advection routine is now data management; calls *advect1d*
- **Advect1d** routine: *does transport*
  - *Averages* velocity to q1d() location. *It* expects -
    - 1-D velocity array with nx+1 elements, q1d array w/ghost points



Program 2 - details

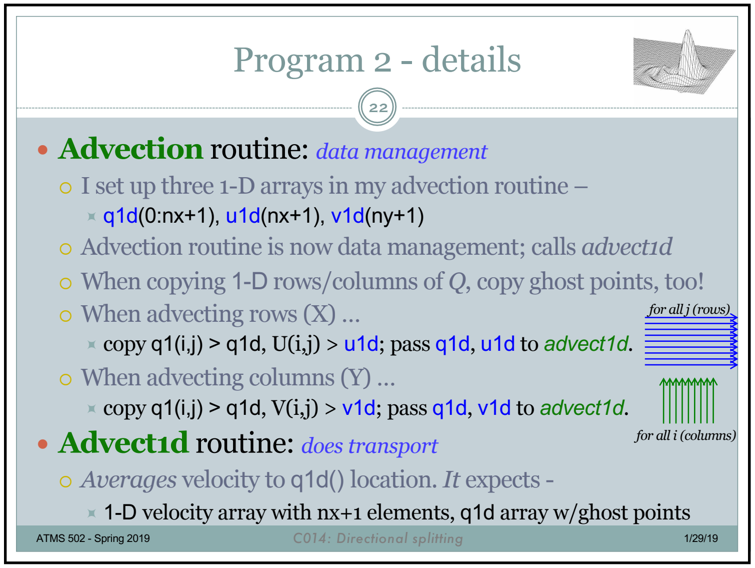
21

- **Advection** routine: *data management*
  - I set up three 1-D arrays in my advection routine –
    - q1d(0:nx+1), u1d(nx+1), v1d(ny+1)
  - Advection routine is now data management; calls *advect1d*
  - When copying 1-D rows/columns of *Q*, copy ghost points, too!
  - When advecting rows (X) …
    - copy q1(i,j) >q1d, U(i,j) > u1d; pass q1d, u1d to *advect1d*.

for all j (rows)

- **Advect1d** routine: *does transport*
  - *Averages* velocity to q1d() location. *It* expects -
    - 1-D velocity array with nx+1 elements, q1d array w/ghost points

ATMS 502 - Spring 2019 C014: Directional splitting 1/29/19



Program 2 - details

22

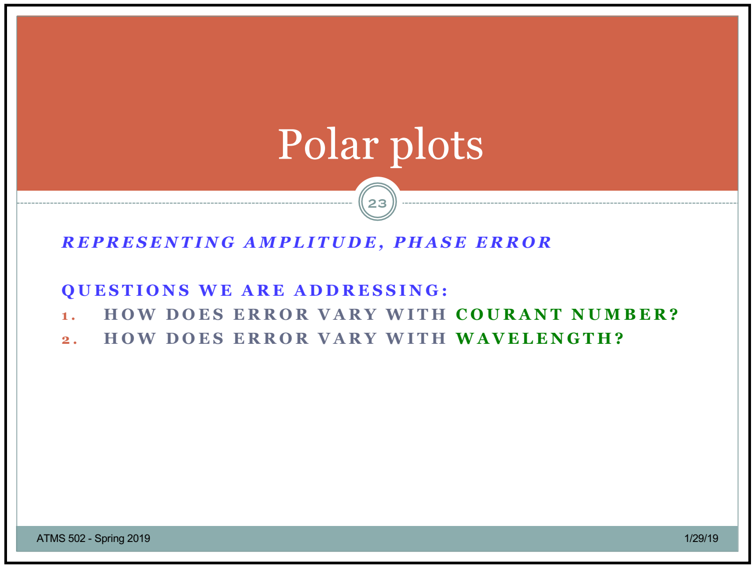
- **Advection** routine: *data management*
  - I set up three 1-D arrays in my advection routine –
    - q1d(0:nx+1), u1d(nx+1), v1d(ny+1)
  - Advection routine is now data management; calls *advect1d*
  - When copying 1-D rows/columns of *Q*, copy ghost points, too!
  - When advecting rows (X) …
    - copy q1(i,j) > q1d, U(i,j) > u1d; pass q1d, u1d to *advect1d*.
  - When advecting columns (Y) …
    - copy q1(i,j) > q1d, V(i,j) > v1d; pass q1d, v1d to *advect1d*.

for all j (rows)

for all i (columns)

- **Advect1d** routine: *does transport*
  - *Averages* velocity to q1d() location. *It* expects -
    - 1-D velocity array with nx+1 elements, q1d array w/ghost points

ATMS 502 - Spring 2019 C014: Directional splitting 1/29/19



Polar plots

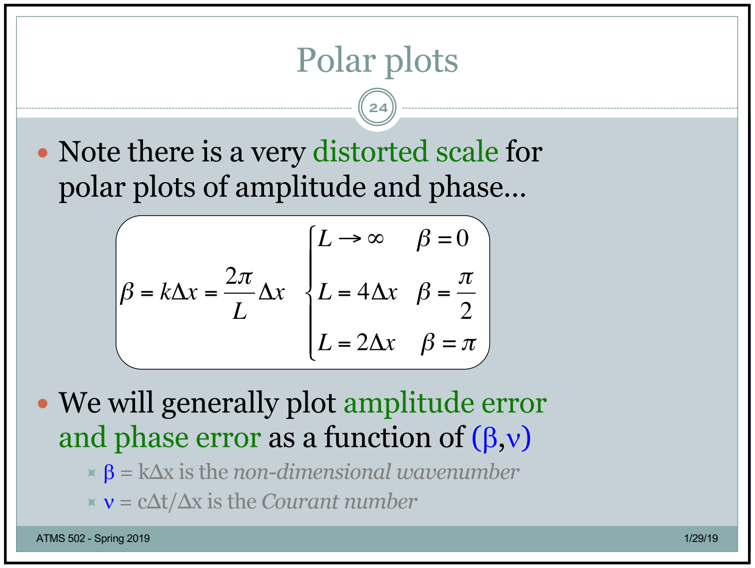
23

REPRESENTING AMPLITUDE, PHASE ERROR

QUESTIONS WE ARE ADDRESSING:

1. HOW DOES ERROR VARY WITH COURANT NUMBER?
2. HOW DOES ERROR VARY WITH WAVELENGTH?

ATMS 502 - Spring 2019 1/29/19



Polar plots

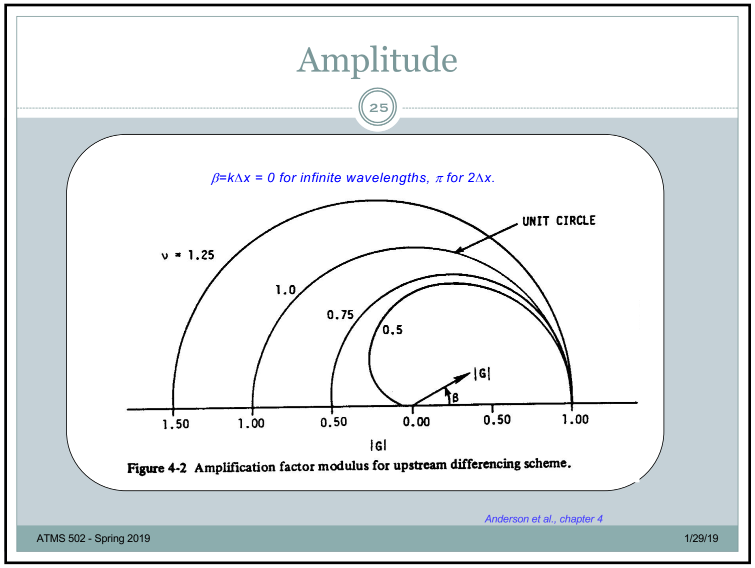
24

- Note there is a very distorted scale for polar plots of amplitude and phase…

$$\beta = k\Delta x = \frac{2\pi}{L}\Delta x \quad \begin{cases} L \to \infty & \beta = 0 \\ L = 4\Delta x & \beta = \frac{\pi}{2} \\ L = 2\Delta x & \beta = \pi \end{cases}$$

- We will generally plot amplitude error and phase error as a function of (β,ν)
  - β = kΔx is the *non-dimensional wavenumber*
  - ν = cΔt/Δx is the *Courant number*

ATMS 502 - Spring 2019 1/29/19

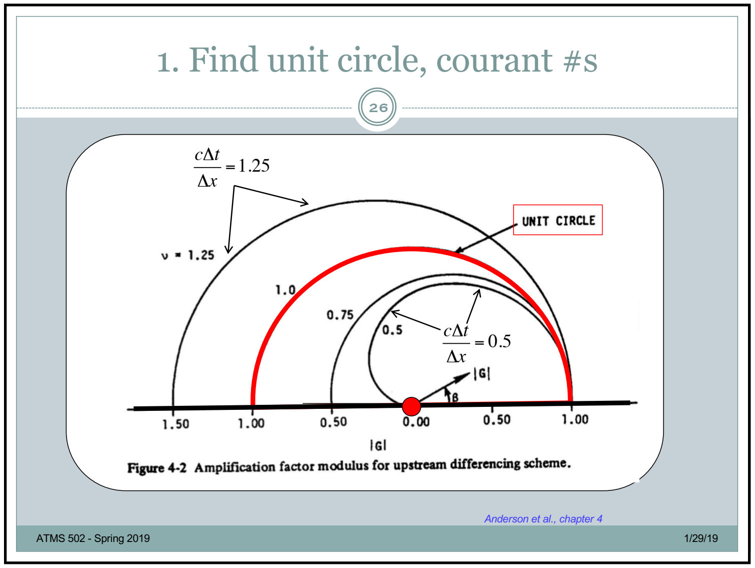


Amplitude

$\beta = k\Delta x$ = 0 for infinite wavelengths, π for $2\Delta x$.

Figure 4-2 Amplification factor modulus for upstream differencing scheme.

Anderson et al., chapter 4



1. Find unit circle, courant #s

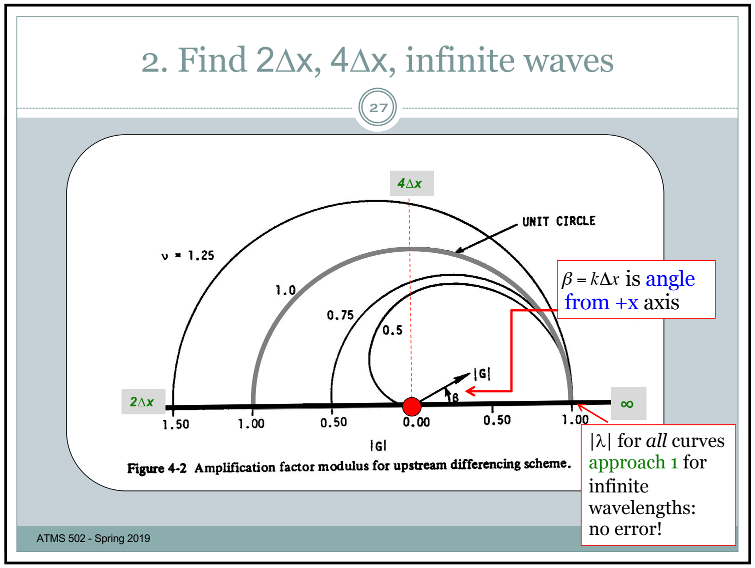
$\frac{c\Delta t}{\Delta x} = 1.25$

UNIT CIRCLE

$\frac{c\Delta t}{\Delta x} = 0.5$

Figure 4-2 Amplification factor modulus for upstream differencing scheme.

Anderson et al., chapter 4



2. Find 2Δx, 4Δx, infinite waves

4Δx

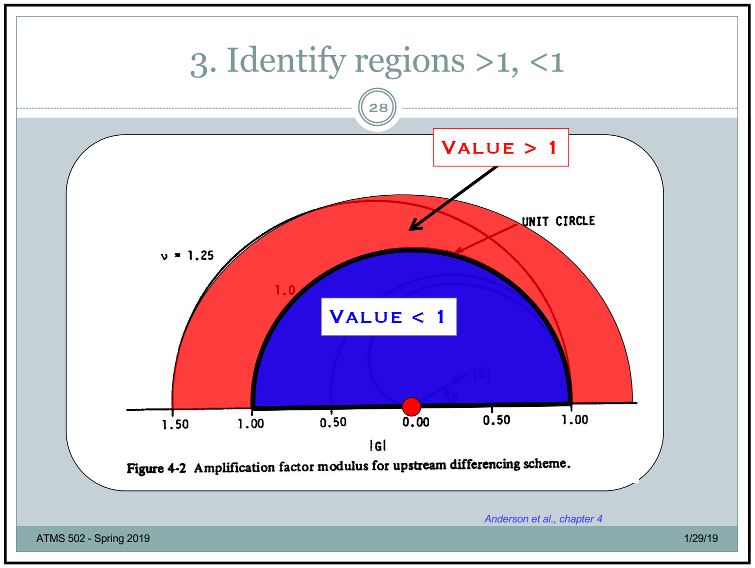
2Δx

∞

$\beta = k\Delta x$ is angle from +x axis

|λ| for *all* curves approach 1 for infinite wavelengths: no error!

Figure 4-2 Amplification factor modulus for upstream differencing scheme.



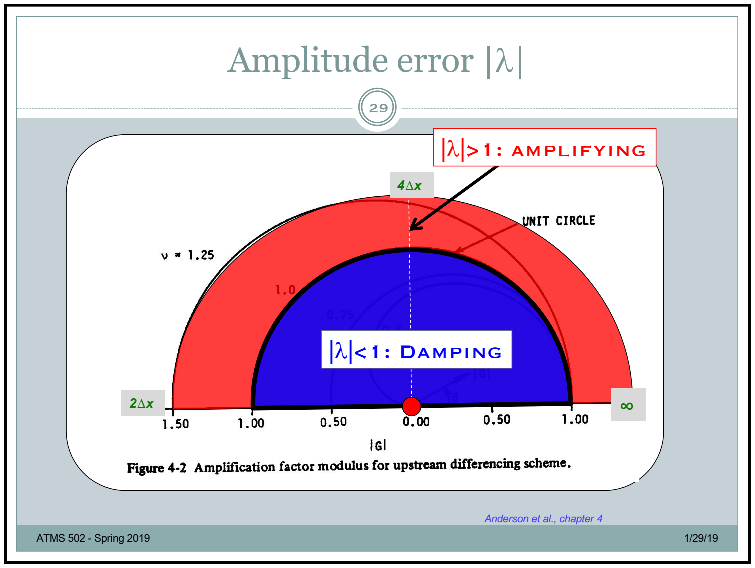
3. Identify regions >1, <1

VALUE > 1

VALUE < 1

Figure 4-2 Amplification factor modulus for upstream differencing scheme.

Anderson et al., chapter 4



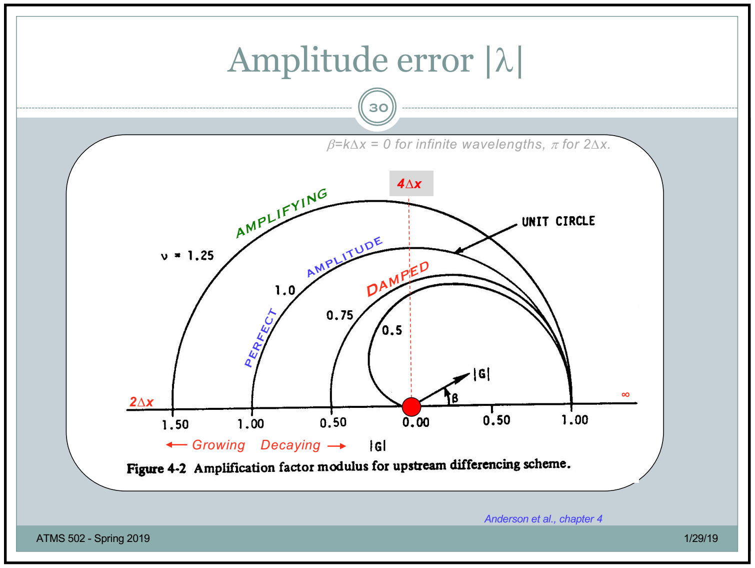
Amplitude error |λ|

29

Figure 4-2 Amplification factor modulus for upstream differencing scheme.

Anderson et al., chapter 4

ATMS 502 - Spring 2019 1/29/19



Amplitude error |λ|

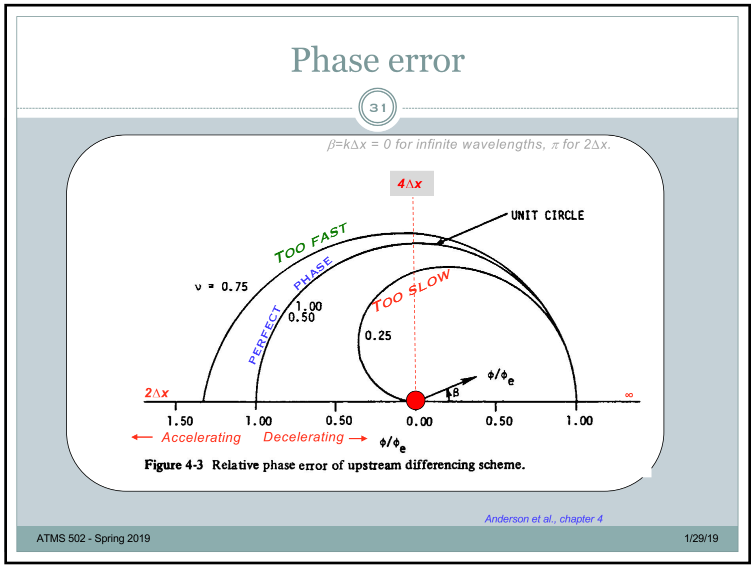
30

β=kΔx = 0 for infinite wavelengths, π for 2Δx.

Figure 4-2 Amplification factor modulus for upstream differencing scheme.

Anderson et al., chapter 4

ATMS 502 - Spring 2019 1/29/19



Phase error

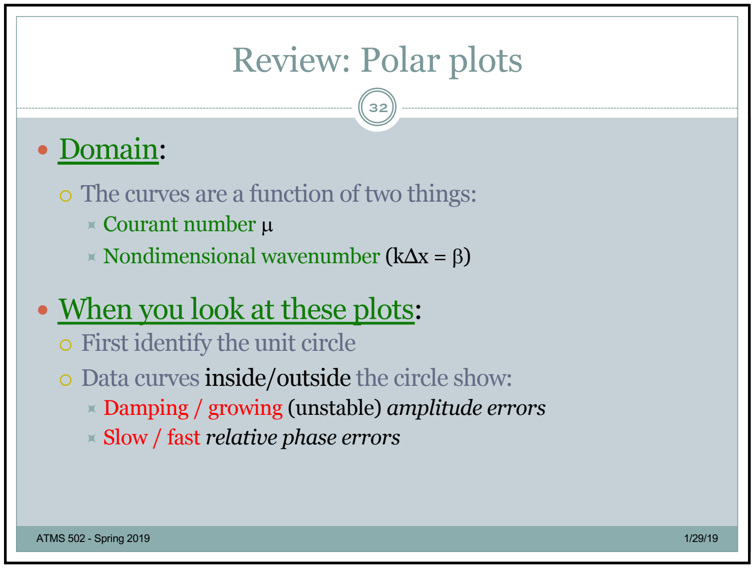
31

β=kΔx = 0 for infinite wavelengths, π for 2Δx.

Figure 4-3 Relative phase error of upstream differencing scheme.

Anderson et al., chapter 4

ATMS 502 - Spring 2019 1/29/19



Review: Polar plots

32

- Domain:
  - The curves are a function of two things:
    - Courant number μ
    - Nondimensional wavenumber (kΔx = β)
- When you look at these plots:
  - First identify the unit circle
  - Data curves inside/outside the circle show:
    - Damping / growing (unstable) *amplitude errors*
    - Slow / fast *relative phase errors*

ATMS 502 - Spring 2019 1/29/19