Simulation of a coronal mass ejection.
Amari et al./Nature; link

ATMS 502
CSE 566

NUMERICAL
FLUID
DYNAMICS

THU. JAN. 17, 2019

ATMS 502 - Spring 2019 1 1/18/19

2

ATMS 502
CSE 566

Thursday,
17 January 2019

Class #2

Plan for Today

- 1) METHODS :
  Lax-Wendroff
  Periodic LBCs
  Finite differences
- 2) CODE/DATA:
  Using Stampede
  Program #1 changes
  Boundary condition coding
  Plotting
- 3) FLUID FLOW equations
  Advective terms; linearity

ATMS 502 - Spring 2019 1/18/19

Methods: *Lax-Wendroff*

3

AN APPROXIMATION TO THE LINEAR 1-D ADVECTION EQUATION

References:
- C001 (Lax-Wendroff)
- C003 (time levels)
- C005 (periodic boundaries)
- C006 (finite differences)

ATMS 502 - Spring 2019 1/18/19

Lax-Wendroff

4

1. Scheme used for Program 1

$$s_j^{n+1} = s_j^n - \frac{\nu}{2}\left(s_{j+1}^n - s_{j-1}^n\right) + \sigma\left(s_{j+1}^n - 2s_j^n + s_{j-1}^n\right)$$

This is the Lax-Wendroff method in 1-D.

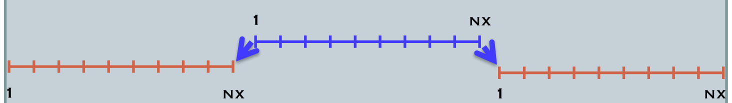
2. We can summarize this numerical method as –
   - An approximation to the *1-way wave equation*
   - A *finite difference* method
     - contrast with: finite volume, semi-Lagrangian … other methods
   - An *explicit* method
     - "explicit" means: given current (time *n*) data, we can solve for the future (time *n+1*) state directly, one point at a time.
   - A *2-time-level* method: *involves only two levels (n, n+1).*
   - Uses a *3-point stencil: uses grid points (j-1, j , j+1)*

ATMS 502 - Spring 2019 C.001: Lax-Wendroff 1/18/19

Our periodic boundary conditions

5

- We are using a *3-point-stencil* for our problem.
  - solution at point "j" requires knowing values at (j-1, j, j+1)
  - if our arrays were dimensioned [1...nx] then
    - at left end, array(j-1) would be undefined
    - at right end, array(j+1) would be undefined
  - we employ a "trick."
    - dimension arrays with 1 extra "ghost" point at either end.
    - set appropriately, in *bc* routine
    - now solve s2(1, ..., nx) = our solver method(1, ..., nx)



ATMS 502 - Spring 2019 | C005: Periodic BCs | 1/18/19

Finite difference approximations to derivatives

6

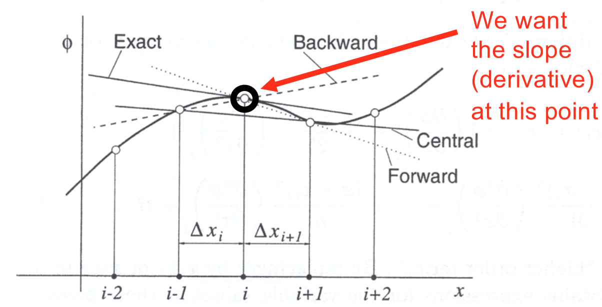


Fig. 3.2. On the definition of a derivative and its approximations

ATMS 502 - Spring 2019 | C006: Finite differences | Ferziger and Peric (2002), p. 41 | 1/18/19

Approximations

7

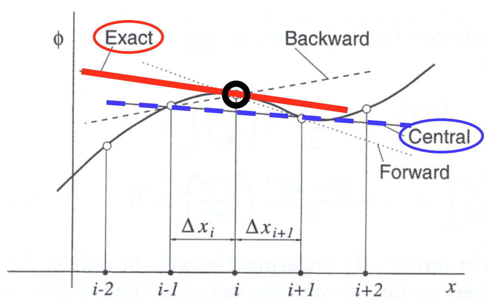


Fig. 3.2. On the definition of a derivative and its approximations

ATMS 502 - Spring 2019 | C006: Finite differences | 1/18/19

Notes

8

ATMS 502 - Spring 2019 | 1/18/19

Code/data: *Program #1*

9

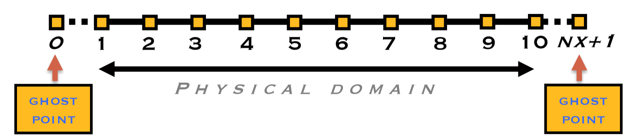
ARRAYS AND PERIODIC BCS
PROGRAMMING ON STAMPEDE
MAKING PLOTS

ATMS 502 - Spring 2019 1/18/19

Arrays & BCs: Fortran

10

- Say you have an array ***s***, dimensioned s(nx)
- [1...nx] covers the ***physical domain***
- Now say we will employ boundary conditions
  - ... and the ghost point region is 1-grid-point-wide
  - ... to do: dimension your array s(0:nx+1)

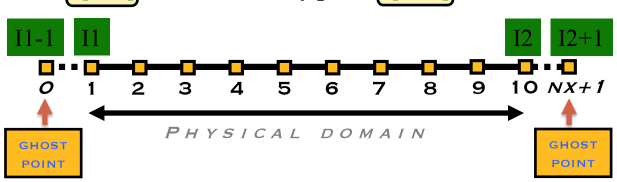


ATMS 502 - Spring 2019 1/18/19

Arrays and BCs: C

11

- In demo code, we use several definitions ...
- The array ***s*** is dimensioned s[NXDIM]
- NXDIM = NX grid points + 2 ghost points
- [I1...I2] covers the ***physical domain***
- s[I1-1] is first boundary point, s[I2+1] the last



ATMS 502 - Spring 2019 1/18/19

Getting started on Stampede

12

you'll need this!

Xsede 2-factor authentication

- First: login
  - My Xsede > Accounts > *Single sign-on login hub*
  - Better: login via *Xshell* (windows) or *iTerm2* (mac)
- C programming:
  - math:
    - use *#include <math.h>;* for powers, use *double pow(); float powf()*
  - loops
    - for() acts only on next line *unless* you declare a code block: { ... }
- Connecting with 2-FA:
  - PC apps may require *interactive* rather than *password* connection to deal with extra step with TACC token
  - File transfer apps should use SFTP, ***not*** FTP, for transfer

ATMS 502 - Spring 2019 1/18/19

Program 1

13

- Program changes:
  - Your name in program!
  - change from 25 to 75 points (this is dimension of your 1-D array)
  - BC routine: add code for our periodic boundary conditions
  - Advection routine: add code for Lax-Wendroff method.
- Due date
  - *one week from* today
  - Drop off plots; upload code.
- Nuts and bolts
  - I hope you can login to Stampede, today
  - I hope soon you can
    - edit Stampede files remotely (*vi, nano, emacs*)
    - transfer files back to your PC and edit them locally
    - see web page references!
  - We will discuss now how to convert your plot files and print them.

ATMS 502 - Spring 2019 1/18/19

Program 1: plots

14

- I hope you can run the demo program and get a *NCAR Graphics* plot file named *gmeta*. If so…
  - Run my metagif program to make GIFs
    ~tg457444/502/Tools/metagif gmeta -all -zip
  - This creates "gmeta.zip"; transfer it back to your PC; extract files from .zip file; put GIFs into a MS Word or Open Office document; print locally.
- Viewing alternatives.
  - *idt gmeta* on Stampede. X-windows must be running.
  - *Web viewer* – a bit later.

ATMS 502 - Spring 2019 1/18/19

Fluid flow equations:
1D advection

15

- 1-D transport ("advection")
- Linear vs. nonlinear problems

References:
- A001 (fluid flow)
- A002 (advection)
- A003 (nonlinearity)
- A005 (1-way wave equation)

ATMS 502 - Spring 2019 1/18/19

Left side of "=" … time rate of change of each variable.

Right side: *advective* terms for each variable.

$$\frac{\partial u}{dt} = -\vec{V}\cdot\vec{\nabla}u - \frac{1}{\bar{\rho}}\frac{\partial p}{\partial x} + \nu\nabla^2 u$$

$$\frac{\partial v}{dt} = -\vec{V}\cdot\vec{\nabla}v - \frac{1}{\bar{\rho}}\frac{\partial p}{\partial y} + \nu\nabla^2 v$$

$$\frac{\partial w}{dt} = -\vec{V}\cdot\vec{\nabla}w - \frac{1}{\bar{\rho}}\frac{\partial p}{\partial z} + g\frac{\theta}{\bar{\theta}} + \nu\nabla^2 w$$

$$\frac{\partial \theta}{\partial t} = -\vec{V}\cdot\vec{\nabla}\theta + Q(x,y,z,t) + \nu\nabla^2\theta$$

$$\frac{\partial p}{\partial t} = -c_s^2\left[\bar{\rho}\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right) + \frac{\partial(\bar{\rho}z)}{\partial z}\right] + \nu\nabla^2 p$$

VARIABLES:
- u: X-wind component
- v: Y-wind component
- w: Z-wind component
- θ: potential temperature
- p: pressure

Units: ____________

ATMS 502 - Spring 2019 16 1/18/19

Temperature equation, in 1-D

17

- Now just look at the θ equation … in 1-D
  - This: $\frac{\partial\theta}{dt} = -\vec{V}\cdot\vec{\nabla}\theta$ … in 1D reduces to this: $\frac{\partial\theta}{\partial t} = -u\frac{\partial\theta}{\partial x}$
  - This describes the local rate of change of temperature θ:
    - θ changes faster (+ or -) if the wind speed u is larger
    - θ changes faster (+ or -) if temperature gradient dθ/dx is larger
  - Example
    - let X extend from Bloomington to Champaign
    - suppose temperature is colder in the west: dθ/dx is positive
    - suppose the wind blows west-to-east: u is positive
    - then dθ/dt is negative; temperature decreases with time.

ATMS 502 - Spring 2019 A.002: advection 1/18/19

Linear or nonlinear?

18

- Linear and nonlinear forms for scalar s(t)
  - This: $\frac{\partial s}{dt} = -\vec{V}\cdot\vec{\nabla}s$ … in 1D reduces to this: $\frac{\partial s}{\partial t} = -u\frac{\partial s}{\partial x}$
  - This describes advection of s by u.
    - We haven't described the form of u here.
  - A purely linear form for θ $\frac{\partial\theta}{\partial t} = -c\frac{\partial\theta}{\partial x}$
    - Replace u by c, *perhaps c(x)*

θ(x) translates from left to right at speed c. Its shape should not change!

  - A nonlinear form for θ:
    - Replace u by the temperature: $\frac{\partial\theta}{\partial t} = -\theta\frac{\partial\theta}{\partial x}$

θ(x) can change shape! θ moves right where θ>0, and moves left where θ<0.

    - θ is advected by 'speed' of … θ!
      - This doesn't make *physical* sense.
      - Mathematically, we understand … $u_t = -uu_x$ is OK.

ATMS 502 - Spring 2019 A.003: nonlinearity 1/18/19

PDE *Order, Linearity*

19

- The order of a PDE is that of the highest-order partial derivative

$$\frac{\partial\theta}{\partial t} = -c\frac{\partial\theta}{\partial x}$$

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = 0$$

- A *linear* PDE is linear in the unknown functions and their derivatives.

$$\frac{\partial u}{\partial t} = -c\frac{\partial u}{\partial x}$$

$$\frac{\partial u}{\partial t} = -u\frac{\partial u}{\partial x}$$

  - … thus the coefficients depend *only* on *independent* variables.

ATMS 502 - Spring 2019 A.004: PDE order, linearity 1/18/19

Notes

20

ATMS 502 - Spring 2019 1/18/19

An initial value problem

21

"PREDATOR-PREY"

AKA LOTKA-VOLTERRA EQUATIONS (1925)

Following Golub and Ortega, 1992

ATMS 502 - Spring 2019 1/18/19

Predator-Prey

22

- Objective: see how populations of predators and prey vary with time
- Our objective: example of numerical solution of a simple system
- Assume some initial population of each
- Simple (ordinary) differential equations describe behavior

ATMS 502 - Spring 2019 1/18/19

www.fletcherledger.com

Predator-Prey

23

- Predator (y):
  - Population decreases if prey removed
  - Assume # of kills depends on probability of meeting prey, proportional to x*y
- Prey (x):
  - If alone, population increases at rate proportional to x, i.e. exponential growth.

$$\frac{dx}{dt} = \alpha x + \beta xy \qquad \frac{dy}{dt} = \gamma y + \delta xy$$

ATMS 502 - Spring 2019 1/18/19

Predator-Prey

24

$$\frac{dx}{dt} = \alpha x + \beta xy \qquad \frac{dy}{dt} = \gamma y + \delta xy$$

- Given initial x(time=0), y(t=0) …
- We can *step forward* x(t) and y(t)

- Need to approximate the time derivatives dx/dt, dy/dt

ATMS 502 - Spring 2019 1/18/19

Simplifying

25

$$f(t+\Delta t) = f(t) + \Delta t f'(t) + \frac{\Delta t^2}{2!} f''(t) + \ldots$$

We'll skip the last term, thereby using a forward derivative here. For the predator-prey equations, f'(t) will be dx/dt and dy/dt.

ATMS 502 - Spring 2019 1/18/19

Simplifying

26

Exact

$$\frac{dx}{dt} = \alpha x + \beta xy \qquad \frac{dy}{dt} = \gamma y + \delta xy$$

Approximate

$$x(t+\Delta t) \approx x(t) + \Delta t[\alpha x + \beta xy]_t$$
$$y(t+\Delta t) \approx y(t) + \Delta t[\gamma y + \delta xy]_t$$

The numerical solution (crude!) is simple here:
1) Take current x(t),y(t), compute right side terms.
2) Use new x(t+Δt),y(t+Δt) in next iteration.

ATMS 502 - Spring 2019 1/18/19

Solution

27

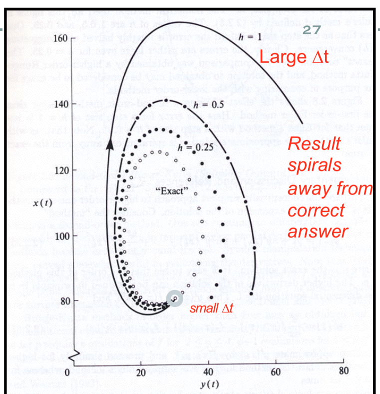


Figure 2.7: Euler's Method for (2.2.34), (2.2.35) Using Different Step Sizes

Golub and Ortega (1992), p. 33

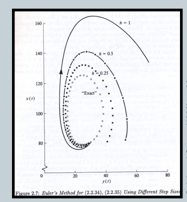
- The "exact" solution here is shown with open circles.
- For larger time steps ("h"), the solution diverges with time, ever farther from truth.

ATMS 502 - Spring 2019 1/18/19

Discussion

28

- Different methods are available for solving the same problem. We discussed just one.
- The numerical solution *approaches the* correct answer under certain conditions. This is a check that the method is satisfactory.
- Larger Δt here gave poor results, but smaller Δt took more computer time. There is an obvious tradeoff.



Golub and Ortega (1992), p. 33

ATMS 502 - Spring 2019 1/18/19

Generalizing…

29

- For various physical (mathematical) problems -
  - Evaluate methods
  - Determine time *(and space)* steps (discretization)
  - What defines a good answer?

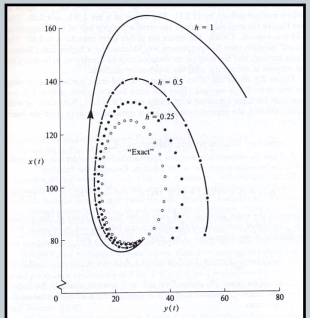


Figure 2.7: Euler's Method for (2.2.34), (2.2.35) Using Different Step Sizes

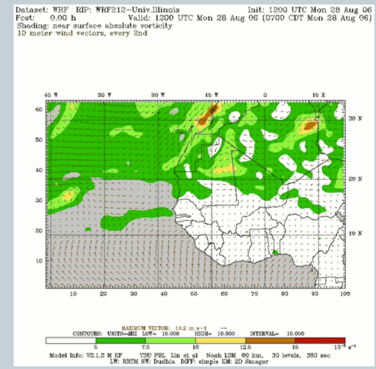
ATMS 502 - Spring 2019 1/18/19

Resolution

30

- A coarse resolution forecast ($\Delta x = \Delta y = 60km$)
- Advantages:
  - quick run
  - large domain ok
- Disadvantages:
  - accuracy
  - limited detail

TROPICAL WAVES OFF AFRICA



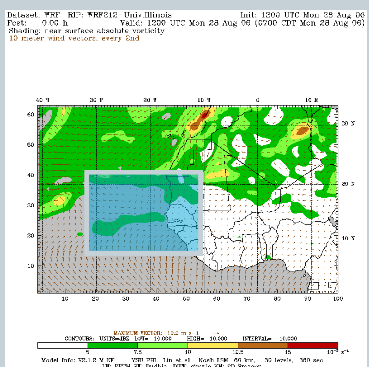
ATMS 502 - Spring 2019 1/18/19

Resolution

31

- A coarse resolution forecast ($\Delta x = \Delta y = 60km$)
- Nest placement
  - higher spatial resolution
  - … but only over a limited area

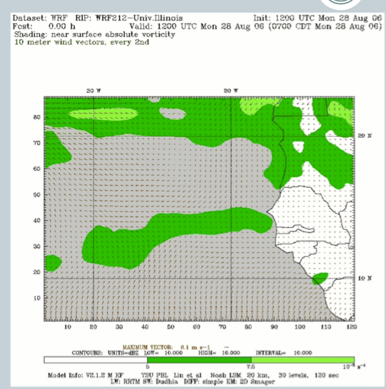
TROPICAL WAVES OFF AFRICA



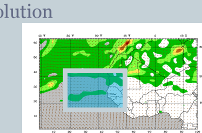
ATMS 502 - Spring 2019 1/18/19

Nested grids

32



- Subdomain
- 3x resolution
- 20 rather than 60 km
- Smaller Δx means:
  - Less *discretization* error -- more accurate
  - Finer details
  - More computation cost (for a given region size) than coarse resolution



ATMS 502 - Spring 2019 1/18/19