Numerical Algorithms for HPC

Solution of Time-Dependent PDEs

epcc

1

Overview

- Boundary Value Problem: pollution model
- Solution using Jacobi
- Discretisation of time-dependent problem
- Euler equations
- Stability
- Implicit methods
- Error Analysis
- Other equations

epcc 2

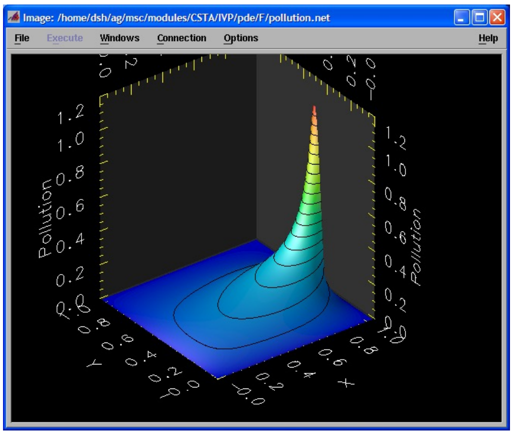
2

Pollution Model

- Previously posed as a boundary value problem
  - i.e. static in time
- What is the solution to: $-\left(\frac{\partial^2}{\partial x^2}+\frac{\partial^2}{\partial y^2}\right)u(x,y)=0$
  - with $u\ =\ 0$ on north, south and west boundaries
  - and $u\ =$ hump function on east boundary (location of chimney)
- Discretised equations using standard recipes
  - e.g. 5-point stencil for derivative
- Can be solved using standard methods
  - e.g. Jacobi, Over-Relaxed Gauss-Seidel, Conjugate Gradient (later), ...

Example Solution

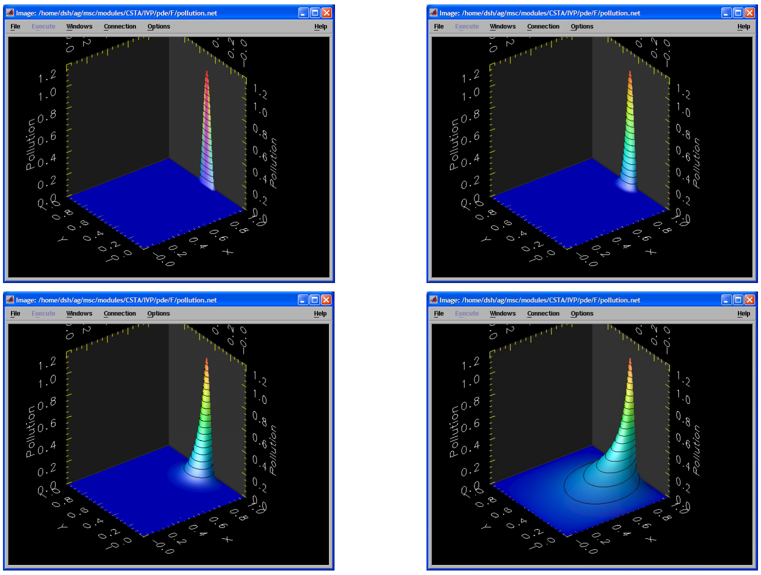
- Use a 72×72 grid
  - included a north-easterly wind of strength $(10.0, 4.0)$



Evolution of the Solution

- Initial guess to solution is the hump function
  - i.e. the situation when chimney is just switched on
- Final output is the static solution
  - i.e. the situation when the chimney has been on for a long time
- What about the intermediate solutions
  - are they related in any way to the actual evolution in time?
- With Jacobi, yes ...

Jacobi (after 0, 10, 100, 1000 iterations)



How do we get real time evolution?

- This is an Initial Value Problem (IVP)
  - specify the solution at time $t = 0$
  - need to know the solution at some later time t
- Similar issues to orbits practical
  - initialise position
  - compute force
  - update position
- Except we have many variables to update
  - here, pollution values at more than 5000 points (on 72×72 grid)
  - force term is calculated from the grid

epcc

Diffusion Equation

- We have considered : $\nabla^2 u(x,y) = \left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}\right) u(x,y) = 0$
  - this is an example of an *elliptic equation*
- Full problem is actually: $\frac{\partial u}{\partial t} = \nabla^2 u(x,y)$
  - this is an example of a *parabolic equation*
- We have already solved the static solution
  - i.e. have set time derivative to zero
  - but what if we want to track the evolution in time?

epcc

Discretisation in space and time

- Use a simple five-point stencil for the RHS with spacing h
- Use a simple forward difference for LHS with timestep dt

$$\frac{\partial u}{\partial t} \approx \frac{u^{(t+dt)} - u^{(t)}}{dt}$$

  – superscript refers to real time t and not "computer time" n

- Full equations are: $\frac{\partial u}{\partial t} = \nabla^2 u(x, y)$
  – discretised: $\frac{u_{i,j}^{(t+dt)} - u_{i,j}^{(t)}}{dt} = u_{i,j-1}^{(t)} + u_{i-1,j}^{(t)} - 4u_{i,j}^{(t)} + u_{i+1,j}^{(t)} + u_{i,j+1}^{(t)}$
  – $u_{i,j}^{(t+dt)} = u_{i,j}^{(t)} + dt\left(u_{i,j-1}^{(t)} + u_{i-1,j}^{(t)} - 4u_{i,j}^{(t)} + u_{i+1,j}^{(t)} + u_{i,j+1}^{(t)}\right)$

epcc

9

9

Euler update

- A simple *explicit* scheme
- Jacobi: $u_{i,j}^{(n+1)} = \frac{1}{4}\left(u_{i,j-1}^{(n)} + u_{i-1,j}^{(n)} + u_{i+1,j}^{(n)} + u_{i,j+1}^{(n)}\right)$
- $\Rightarrow \quad u_{i,j}^{(n+1)} = u_{i,j}^{(n)} + \frac{1}{4}\left(u_{i,j-1}^{(n)} + u_{i-1,j}^{(n)} - 4u_{i,j}^{(n)} + u_{i+1,j}^{(n)} + u_{i,j+1}^{(n)}\right)$
- Time dependent:

$$u_{i,j}^{(t+dt)} = u_{i,j}^{(t)} + dt\left(u_{i,j-1}^{(t)} + u_{i-1,j}^{(t)} - 4u_{i,j}^{(t)} + u_{i+1,j}^{(t)} + u_{i,j+1}^{(t)}\right)$$

- Notes
  – Jacobi update corresponds to *time integration* with $dt = 0.25$!
  – smaller values of dt will give more accurate intermediate solutions
    - but will require more timesteps and hence more work
  – should always arrive at the same static solution (eventually) if it exists
  – in real situations there might be no static solution (e.g. turbulent flow)
  – Update of the form $u^{(t+dt)} = u^{(t)} + dt\, f(u^{(t)})$ known as *Euler update*

epcc

10

10

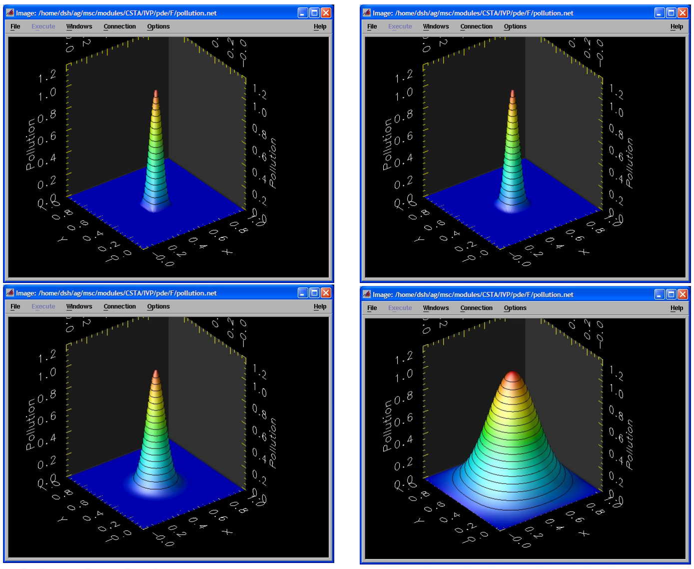
Checking the accuracy

- Initial Value Problems are a leap in the dark
  - how do we know if the solution is correct?
  - Can no longer simply check convergence
- Need to be careful
  - can sometimes monitor conserved quantities, for example the total amount of pollution
  - also look at solution visually
- Start with the hump in the middle (Following week's exercise)
  - easier to understand: pollution stays away from boundaries for longer
  - pollution does not disappear for the initial timesteps
- Note that the height actually decreases as the width widens
  - visualisation software used in next slide (unfortunately) rescales the z-axis

epcc 11

11

$dt = 0.1; t = 0.0, 1.0, 10.0, 100.0$



epcc 12

12

Stability

- Can perform a formal analysis of stability
  - following Von Neumann
- This shows that, for stability in the 2D diffusion problem we require $dt <= 0.25$
  - Jacobi algorithm therefore uses the maximum timestep
- But what happens if we go beyond this?
  - see exercise!

epcc

13

13

Other Schemes

- Imagine evaluating the derivative at time $t + dt$

$$\frac{u_{i,j}^{(t+dt)} - u_{i,j}^{(t)}}{dt} = \nabla^2 u^{(t+dt)}$$

- One arrangement gives an *implicit* scheme:

$$(1 - dt\nabla^2)u^{(t+dt)} = u^{(t)}$$

  - must solve a full boundary value problem at every timestep!
  - however, it is always stable so can use a much larger dt
- Many other integration schemes exist
  - more stable than explicit scheme
  - allow larger timesteps

epcc

14

14

Error analysis: second derivative

$$\nabla^2 u(x) \approx u_{i-1} - 2u_i + u_{i+1}$$

$$u_{i-1} = u(x-h) = u(x) - h\frac{du}{dx} + \frac{h^2}{2}\frac{d^2u}{dx^2} - \frac{h^3}{6}\frac{d^3u}{dx^3} + O(h^4)$$

$$u_i = u(x)$$

$$u_{i+1} = u(x+h) = u(x) + h\frac{du}{dx} + \frac{h^2}{2}\frac{d^2u}{dx^2} + \frac{h^3}{6}\frac{d^3u}{dx^3} + O(h^4)$$

$$\frac{1}{h^2}(u_{i-1} - 2u_i + u_{i+1}) = \frac{d^2u}{dx^2} + O(h^2)$$

- Expression is therefore accurate to *second order* in h
  - straightforward to extend to 2D
  - note we've previously been rather lax about units, i.e. factors of h^2 etc.

Error analysis: first derivative

$$\frac{\partial u}{\partial t} \approx \frac{u^{(t+dt)} - u^{(t)}}{dt}$$

$$\frac{u^{(t+dt)} - u^{(t)}}{dt} = \frac{\partial u}{\partial t} + O(dt)$$

- Euler integration is only accurate to first order
  - but very simple!

Units for diffusion equation

- Actual diffusion equation is: $\frac{\partial u}{\partial t} = D\nabla^2 u$
  - diffusion constant *D* is large for a gas, small for treacle, ...
- Full update equations are:

$$u_{i,j}^{(t+dt)} = u_{i,j}^{(t)} + \frac{Ddt}{h^2}\left(u_{i,j-1}^{(t)} + u_{i-1,j}^{(t)} - 4u_{i,j}^{(t)} + u_{i+1,j}^{(t)} + u_{i,j+1}^{(t)}\right)$$

  - previously we set $D = h^2$ for simplicity
- Stability condition is $dt <= h^2/(4D)$
  - this is the Courant–Friedrichs–Lewy (CFL) condition
- Euler is impractical for the diffusion equation
  - halving h means reducing dt by a factor of 4 ...

17

17

Other equations

- An important equation is the wave equation (1D and 2D)

$$\frac{\partial^2 u}{\partial t^2} = c^2 \frac{\partial^2 u}{\partial x^2}$$

$$\frac{\partial^2 u}{\partial t^2} = c^2 \nabla^2 u$$

- Describes waves travelling with velocity c
  - sound
  - water
  - electromagnetic
  - ...
- This is a *hyperbolic equation*

18

18

Boundary conditions

- We have used very simple fixed boundary conditions
  - these are Dirichlet conditions
  - e.g. pollution is zero on the boundary
  - e.g. simply set $u_{i,0} = 0$ to force this on the southern boundary
- May want conditions on the derivative
  - e.g. pollution is constant across the southern boundary
  - i.e. $\frac{\partial u}{\partial y} = 0$ for all values of x *at* $y = 0$
- Compute discrete form using Taylor expansion
  - here it is simply $u_{i,0} = u_{i,1}$ on the southern boundary
  - re-impose this condition every time we update u

epcc

19

19

Exercise (next week)

- Look at stability of Euler Integration for 2D diffusion
  - vary dt
  - what happens when the CFL condition is violated?
  - is the total amount of pollution conserved?
  - what happens when boundary conditions are changed?

epcc

20

20