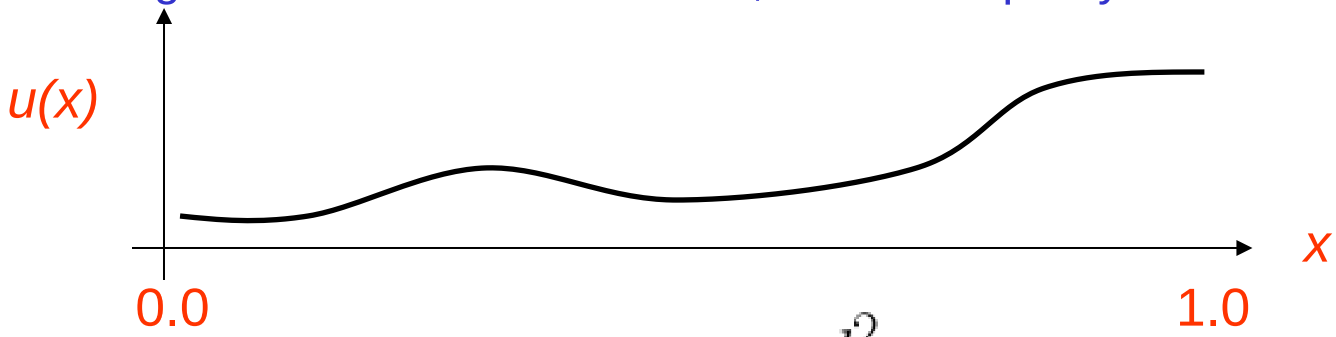
Parallel Numerical Algorithms

Discretised Partial Differential Equations

Overview of Lecture

- Pollution problem as a Partial Differential Equation
  - equations in one and two dimensions
  - boundary conditions
- Discretised equations
  - putting problem onto a lattice
  - PDE as a matrix problem
  - the five-point stencil
  - mapping between the 2D continuous and discrete problems
  - introducing a wind
- Notes
- Summary

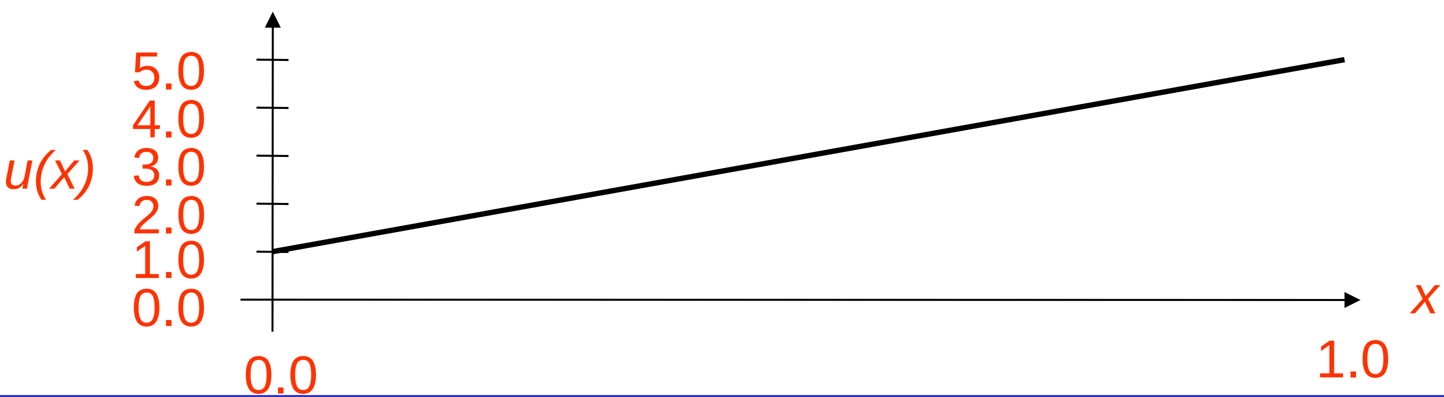
- Imagine one-dimensional problem *with no wind*
  - eg pollution in a valley
- Call the density of pollution *u*
  - distance along the valley is *x* which is in the range [0.0, 1.0]
    - in general the domain size is *L*, but for simplicity we take *L* = 1.0

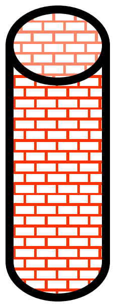


- Differential equation is: $$-\frac{d^2}{dx^2}u(x) = 0$$
  - initial minus sign is a useful convention (see later)
  - equation is for steady state solution that does not vary in time

Analytic Solution

- In one dimension, solution is a straight line
  - equation is: $u(x) = m\,x + c$
  - but what are the values of gradient m and intercept c?
- Actual solution depends on *boundary conditions*
  - differential equation gives the behaviour in the interior (0.0,1.0)
  - must also specify the behaviour at boundaries x=0.0 and x=1.0
  - for example, $u(0.0) = 1.0$ and $u(1.0) = 5.0$

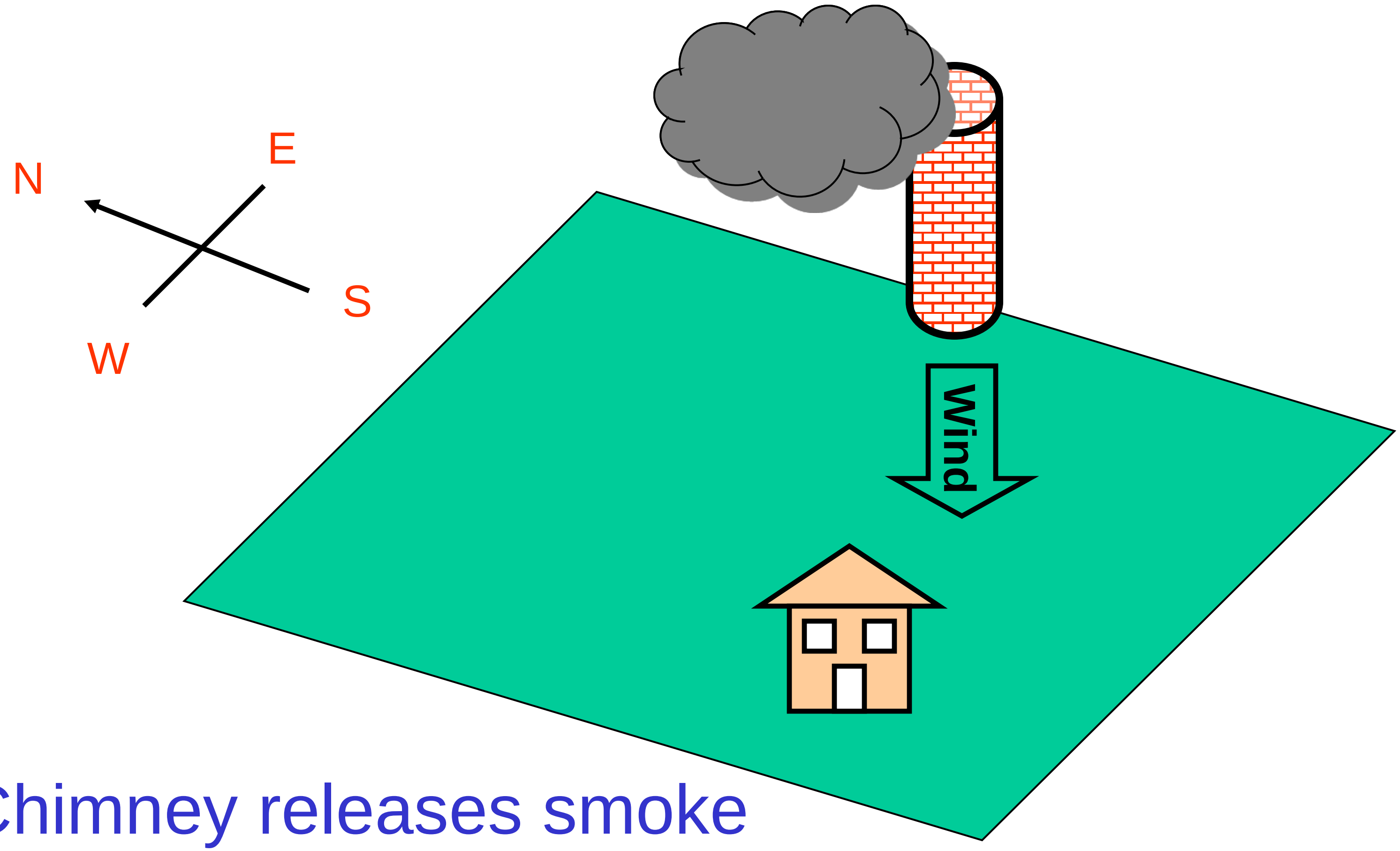




Boundary Conditions

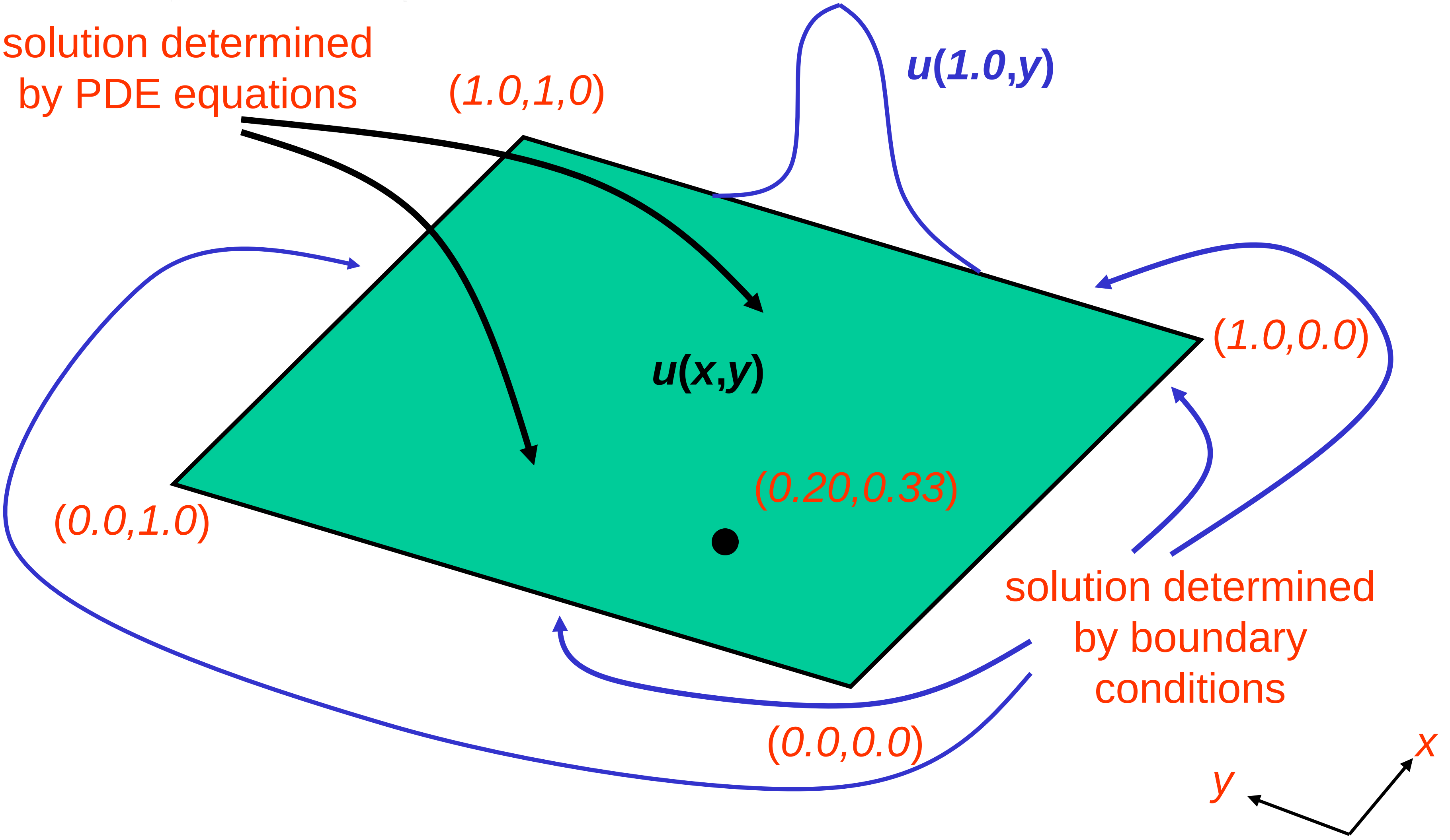
- We solved the equation: $-\frac{d^2}{dx^2}u(x) = 0$
  - with *u*(0.0) = 1.0 and *u*(1.0) = 5.0, the answer is *u*(*x*) = 4.0 x + 1.0
- In general
  - "What is the pollution in a valley" is a meaningless question
  - must ask: "What is the pollution in a valley when the pollution levels are **one** at the western end and **five** at the eastern end"
- Same applies in our two-dimensional problem
  - equations will determine solution *u*(*x*,*y*) in the **interior** region
  - we must independently specify behaviour on all the **boundaries**
- For this reason, steady state problems like this are called *Boundary Value Problems*

The Problem we want to solve



- Chimney releases smoke
  - how much arrives at house with prevailing north-easterly wind?

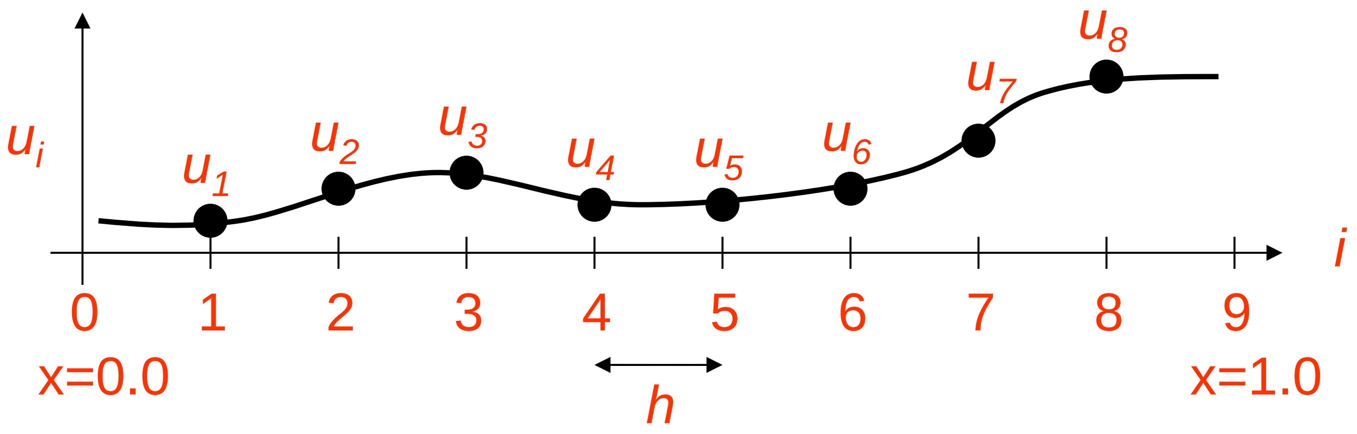
Use 2D Domain (x,y) of Size 1x1



Mathematical Problem in 2D

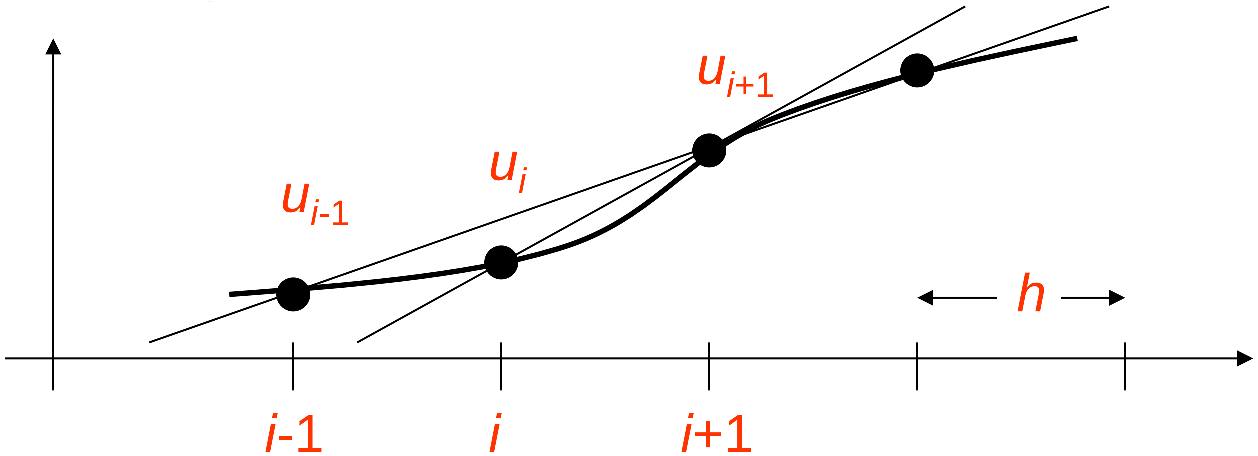
- PDE with no wind is $$-\left(\frac{\partial^2}{\partial x^2}+\frac{\partial^2}{\partial y^2}\right)u(x,y)=0$$
  - all solutions obey this Partial Differential Equation (PDE) in interior region
- Must also specify Boundary Conditions (BCs)
  - BCs must be **appropriate to our specific problem**
- In this case, a simple choice is:
  - set pollution on boundary to zero **everywhere** except at chimney
    - assume domain is large enough that no pollution gets to the edges
  - specify $u(1.0,y)$ as a hump concentrated around (1.0, 0.5)
    - this is a guess at the way pollution is emitted by the chimney
    - a single sharp peak at (1.0,0.5) causes technical problems later!
- Solve the equations somehow ...
  - and the pollution level at the house is the value of $u(0.20,0.33)$

Discretising the Problem



- Replace continuous real x by discrete integer i
  - divide domain into a lattice containing M+1 sections each of width h
  - eg in above diagram, M=8 and h = 1.0/(M+1) = 0.11
- Solve for N different variables u_i, i = 1, 2, ... N
  - in one dimension, N = M but not true in general (in 2D problem N=M^2)
  - boundary values are u_0 and u_{N+1} (above, u_0 and u_9)
- But what equations do the u_i variables satisfy?
  - and how do we decide on the boundary values?

Discretising the Equations



- Approximate gradients with lines,
  - eg a forward difference:

$$\frac{d}{dx}u(x) \approx \frac{u_{i+1} - u_i}{h}$$

  - or a central difference:

$$\frac{d}{dx}u(x) \approx \frac{u_{i+1} - u_{i-1}}{2h}$$

- All become more accurate as we reduce *h*
  - but for a given value of *h*, some will be more accurate than others
  - eg forward difference has errors proportional to *h*
    - central has errors proportional to h^2 and is therefore **more accurate**
  - can estimate errors by doing a Taylor expansion about $u(x)$ ...

epcc

Discretised Equations

- Write second derivative as: $$\frac{d^2}{dx^2}u(x) = \frac{d}{dx}\left(\frac{d}{dx}u(x)\right)$$
  - use forward difference for first derivative, then a backward for second

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

- Boundary conditions are straightforward
  - $u(0.0) = 1.0$: $u_0 = 1.0$
  - $u(1.0) = 5.0$: $u_{M+1} = 5.0$
- This gives us *N* equations in *N* unknowns

$$-u_{i-1} + 2u_i - u_{i+1} = 0, \quad i = 1, 2, \ldots N$$

- Converted differential equations into difference equations
  - larger *M* means a smaller *h* and more accurate equations
  - but also a larger *N* and much more work, especially in 2D or 3D problems!

Difference Equations for $N = 8$

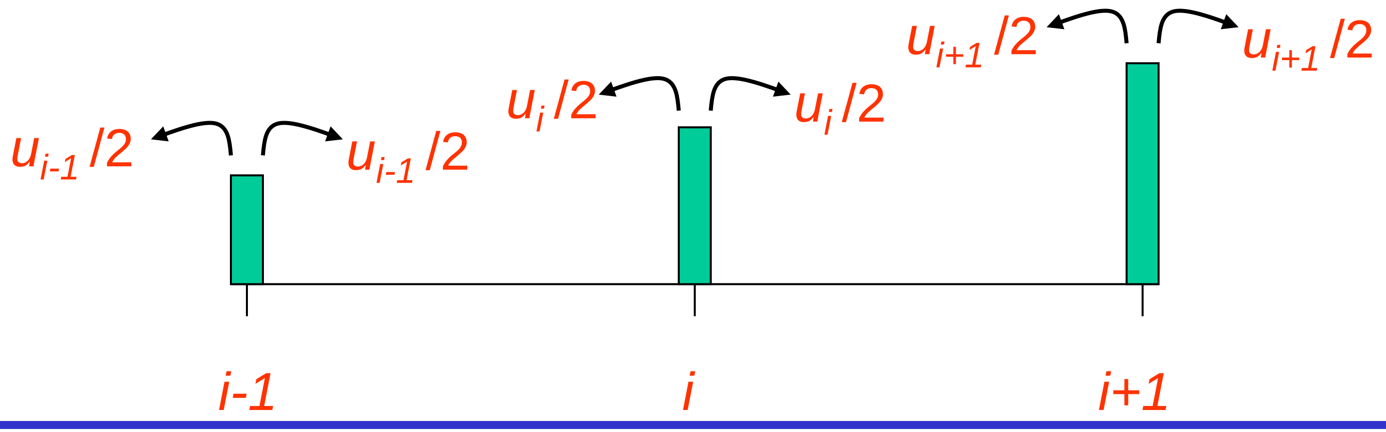
- Writing the eight equations out in full

$$
\begin{aligned}
2u_1 - u_2 &= 1 \\
-u_1 + 2u_2 - u_3 &= 0 \\
-u_2 + 2u_3 - u_4 &= 0 \\
-u_3 + 2u_4 - u_5 &= 0 \\
-u_4 + 2u_5 - u_6 &= 0 \\
-u_5 + 2u_6 - u_7 &= 0 \\
-u_6 + 2u_7 - u_8 &= 0 \\
-u_7 + 2u_8 &= 5
\end{aligned}
$$

  – Notes
    - have multiplied all the equations by h^2 for simplicity
    - first and last equations are different as we know u_0 and u_9
    - we write the known values on the right-hand-side for convenience

Interpreting Difference Equations

- Simple interpretation
  - every point equals the average of its nearest neighbours
  - what has this got to do with diffusion?
- Imagine pollution particles do “a random walk”
  - each step, particles at every lattice point move randomly left or right
  - let u_i be the number of particles at lattice point i



Steady State Random Walk

- At each step
  - population u_i is replaced by u_{i-1} /2 (from left) and u_{i+1} /2 (right)
  - for a steady state, $u_i = (u_{i-1} + u_{i+1}) /2$
  - same equations as before: $- u_{i-1} + 2\, u_i - u_{i+1} = 0, \quad i = 1, 2, \ldots N$
- Perhaps easier to understand than: $-\dfrac{d^2}{dx^2}\, u(x) = 0$
- Note that this is a dynamic equilibrium
  - just because pollution level $u(x)$ is constant doesn't mean that the pollution particles are static
  - eg density of air is constant even though molecules are moving!

Equations in Matrix Form

- These can be written in standard form *Au* = *b*

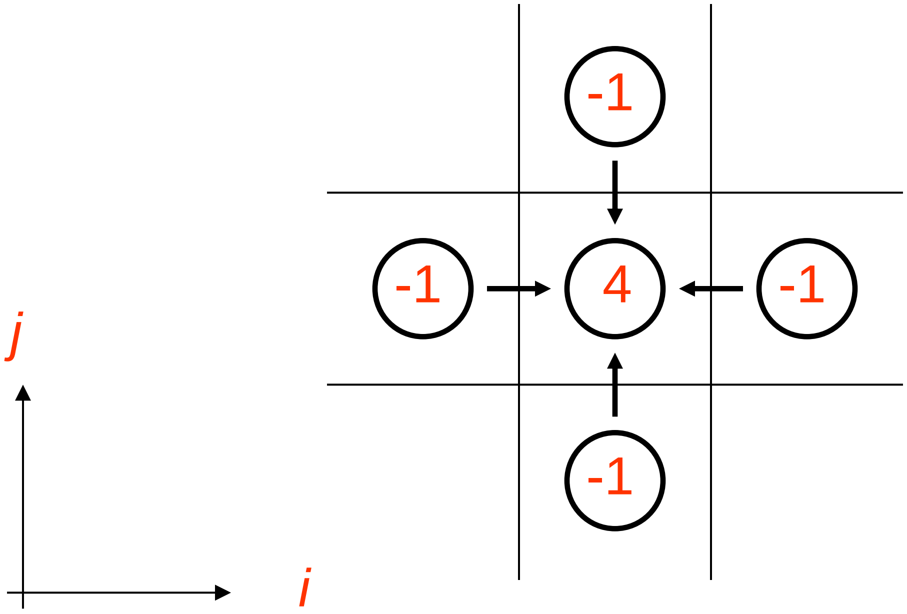
$$\begin{bmatrix} 2 & -1 & & & & & & \\ -1 & 2 & -1 & & & & & \\ & -1 & 2 & -1 & & & & \\ & & -1 & 2 & -1 & & & \\ & & & -1 & 2 & -1 & & \\ & & & & -1 & 2 & -1 & \\ & & & & & -1 & 2 & -1 \\ & & & & & & -1 & 2 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \\ u_7 \\ u_8 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 5 \end{bmatrix}$$

  – in this case, *A* is sparse and symmetric

Two Dimensional Problem

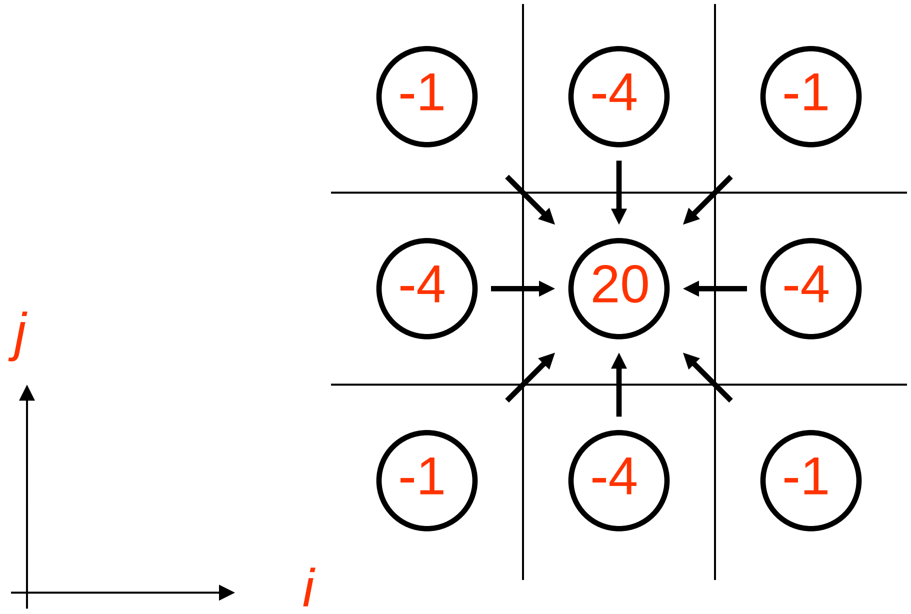
- Simple extension to two dimensions
  - impose a square lattice of size *M*+1 by *M*+1, spacing *h*
  - replace real continuous coordinates (*x*,*y*) by integers *i*,*j*
  - solution is now $u_{i,j}$ with $i = 1, 2, \ldots M$ and $j = 1, 2, \ldots M$
  - the number of unknowns *N* is now M^2
  - in 1D $$\frac{d^2}{dx^2}u(x) \approx \frac{u_{i-1} - 2u_i + u_{i+1}}{h^2}$$
  - in 2D: $$\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}\right)u(x,y) \approx \frac{u_{i,j-1} + u_{i-1,j} - 4u_{i,j} + u_{i+1,j} + u_{i,j+1}}{h^2}$$
  - every point is averaged with its **four nearest neighbours**

- The equation can be represented graphically
  - (remember the initial minus sign!)
  - again, can easily be interpreted as a random walk



More Accurate Stencils

- More accuracy means more complicated shape
  - eg a nine-point stencil for the same equation includes $u_{i+1,j+1}$, ...
  - can be understood as a random walk, now also including diagonals



Notation

- The vector *b* is often called the *source*
  - remember that it contains all the fixed boundary values of *u*
  - for 2D problem, corresponds to hump function around chimney
    - the hump is clearly the *source* of the pollution
- The 2D diffusion operator is very common
  - has a special name, “Grad Squared”, and symbol: ∇^2
- Can write the 2D equations as: $-\nabla^2 u(x,y) = 0$
  - the five-point stencil is a standard discretisation of ∇^2
  - different discretisations (or different equations) will lead to a different form for the matrix *A*
- Another notation indicates derivatives by ’

$$\frac{d}{dx} u(x) \rightarrow u'(x), \quad \frac{d^2}{dx^2} u(x) \rightarrow u''(x)$$

epcc

Grid Coordinates vs Real Space

- We store values on a discrete grid
  - $u_0, u_1, u_2, \ldots, u_{N-1}, u_{N+1}$
- What points do these represent in real space?
  - in 1D: $x = i*h$ $\quad u_i \rightarrow u(i*h)$
  - in 2D: $x = i*h$, $y = j*h$ $\quad u_{i,j} \rightarrow u(i*h, j*h)$
- Converting from real space to grid points?
  - much harder as coordinate x will not sit exactly on the grid
  - to get the value of $u(x)$ from the grid, must do some sort of interpolation of u_i from the nearby grid points
  - simplest solution is a weighted average – see exercise notes

Introducing a Wind

- More pollution moves in same direction as wind
  - in 1D, the equations for a wind of strength *a* (from the right) are

$$-\frac{d^2}{dx^2}u(x) - a\frac{d}{dx}u(x) = 0$$

$$\frac{-u_{i-1} + 2u_i - u_{i+1}}{h^2} - a\left(\frac{u_{i+1} - u_i}{h}\right) = 0$$

$$-\left(\frac{1}{h^2}\right)u_{i-1} + \left(\frac{2}{h^2} + \frac{a}{h}\right)u_i - \left(\frac{1}{h^2} + \frac{a}{h}\right)u_{i+1} = 0$$

  - more particles move left (from u_{i+1} to u_i) than right
    - makes the associated matrix *A* non-symmetric
    - straightforward to extend to two dimensions

In Two Dimensions

- 2D equations for a NE wind of strength (a_x, a_y)

$$-\left(\frac{\partial^2}{\partial x^2}+\frac{\partial^2}{\partial y^2}\right)u(x,y) - \ a_x\frac{\partial}{\partial x}u(x,y) - \ a_y\frac{\partial}{\partial y}u(x,y) = 0$$

- Use forward differences for first derivatives, eg:

$$\frac{\partial}{\partial x}u(x,y) \approx \frac{u_{i+1,j}-u_{i,j}}{h}$$

  – now straightforward to write out difference equations in full
  – on the computer we deal with the values a_x*h and a_y*h

- What about different boundary conditions?
  - fixed boundary conditions are called Dirichlet conditions
  - might want to specify the gradient at a boundary
    - eg “the slope of the pollution curve should be zero at the edges”
    - these are called Neumann boundary conditions
- Dirichlet conditions affect the right-hand-side *b*
  - Neumann conditions alter the matrix *A* near domain boundaries
- Non-Linear Equations
  - can easily be discretised using standard recipes
  - this will lead to equations like: $u_1^2 + 2\,u_2 + u_3 = 0$
  - this CANNOT be expressed as a matrix equation with constant *A*
    - ie not possible to solve using methods like Gaussian Elimination

Summary

- Many physical problems are expressed as PDEs
  - impose a regular lattice on the problem
  - discretise the differential equations using standard techniques
- This leads to set of *N* difference equations
  - converts PDE to a set of linear equations $Au=b$ which we can solve
  - *A* depends on the PDE, *b* on boundary conditions, solution is *u*
  - *N* may be very large indeed for 2D or 3D problems!
- We are solving an approximation to the PDE
  - even if we solve linear equations accurately, there is still an error
  - can reduce this error using a more accurate discretisation of PDE
    - or a larger *M* (ie smaller value of *h*) with the same discretisation
  - both these approaches require additional work