Exercise – Week 1

Modelling 2D diffusion

The diffusion equation can, for example, be used to describe the distribution and temporal evolution of heat in a medium:

$$\frac{\partial T}{\partial t} = \kappa\left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2}\right) = \kappa \nabla^2 T$$
,

where κ is the thermal diffusivity (m^2/s) and T is temperature (K).

A finite-difference approximation can be used to calculate the first- and second-order derivatives on a grid:

$$\frac{T_{i,j}^{t+\Delta t} - T_{i,j}^{t}}{\Delta t} = \kappa\left(\frac{T_{i-1,j}^{t} - 2T_{i,j}^{t} + T_{i+1,j}^{t}}{\Delta x^2} + \frac{T_{i,j-1}^{t} - 2T_{i,j}^{t} + T_{i,j+1}^{t}}{\Delta y^2}\right)$$

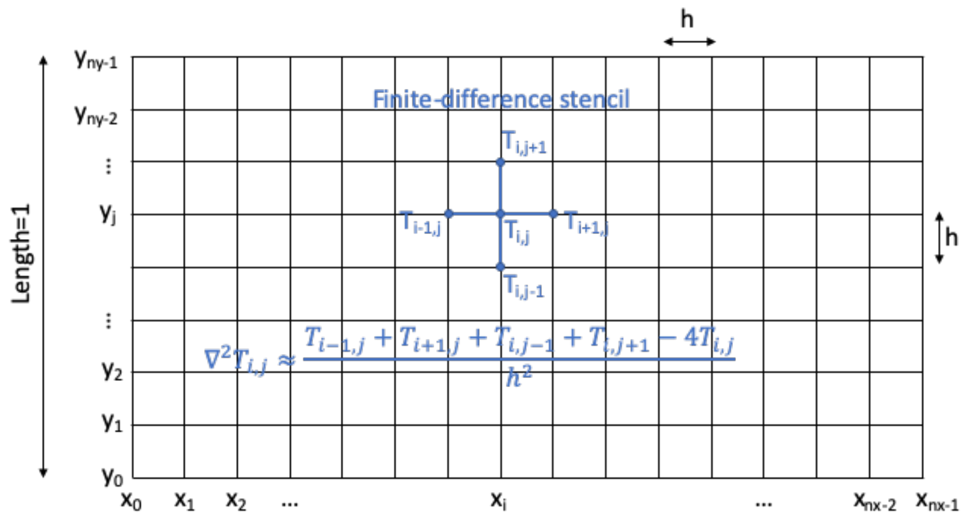
The computation of the next time step depending only on quantities of the current time step can be written as

$$T_{i,j}^{t+\Delta t} = T_{i,j}^{t} + \Delta t \kappa\left(\frac{T_{i-1,j}^{t} - 2T_{i,j}^{t} + T_{i+1,j}^{t}}{\Delta x^2} + \frac{T_{i,j-1}^{t} - 2T_{i,j}^{t} + T_{i,j+1}^{t}}{\Delta y^2}\right)$$

To simplify, we choose $\Delta x = \Delta y = h$:

$$T_{i,j}^{t+\Delta t} = T_{i,j}^{t} + \Delta t \kappa\left(\frac{T_{i-1,j}^{t} + T_{i+1,j}^{t} + T_{i,j-1}^{t} + T_{i,j+1}^{t} - 4T_{i,j}^{t}}{h^2}\right)$$

Represent T on an evenly spaced grid as illustrated in the following figure:



Exercise

Write a Matlab or Python program that solves the 2-D heat diffusion equation as detailed above.

The method is unstable if the time step is too large. This means that the simulation results in oscillations with amplitude that grows exponentially with time (you will learn more about stability in future lectures). The stability depends on the grid spacing: $\Delta t < ah^2/\kappa$, where a is a constant. Try to increase a so that the simulation becomes unstable. Can you determine the value above which the simulation becomes unstable?

Bonus exercise: Modify your program so that you can specify a thermal diffusivity that can vary in space, i.e. $\kappa(x, y)$ and simulate heat diffusion in a heterogeneous medium.

Suggestions of how to structure program

- Specify the initial parameters
  - Number of grid points: nx, ny (Start with nx=100, ny=50)
  - Diffusivity constant: k (Start with k=1)
  - Total integration time: t_end (Start with t_end=0.05)
  - Time step constant: a (Start with a=0.2)
- Initialize the domain
  - Initial temperature field: T0 (either a random field or a spike, start with a spike at x=50,y=15, i.e. $T_{50,25}=1$)
  - Grid spacing: h=1/(ny-1), assume the domain size = 1 in y
  - Time step: dt=ah2/k where a is a constant between 0.1 and 1.0
  - Assume T=0 at the boundaries
    - Make sure the initial field has T=0 at the boundary points
    - Make sure the T field has T=0 at the boundary after each time step
- Calculate the second order derivative
  - Write a function that takes T and h as input parameters and outputs the Laplacian of T as shown on the previous slides
  - Boundary condition: assume the Laplacian is 0 at the boundary points
- Write a while loop to update the field until the total integration time is reached
  - Compute the Laplacian of T by calling your function
  - Update the T field: T=T+dt*kappa*laplace(T)