

Exercise 5 – 2D advection

Adapt your Fortran program from Exercise 4 such that it solves the two-dimensional advection-diffusion equation for incompressible flow ($\nabla \cdot \vec{v} = 0$):

$$\frac{\partial T}{\partial t} + \vec{v} \cdot \nabla T = \kappa \nabla^2 T \quad (1)$$

The flow is given by the stream function ψ :

$$\psi = B \cdot \sin(\pi x / x_{max}) \cdot \sin(\pi y / y_{max}) \quad (2)$$

where B is a constant.

1. Read in the control parameters from a namelist:

- Number of grid points in x and y direction `nx, ny`.
- Diffusion coefficient `kappa`
- Integration time `total_time`
- Time step constants `a_adv` (advection) and `a_diff` (diffusion)
- Flow constant `B`

2. Initialize the variables

- Temperature: 2D array of size `(nx, ny)` with random numbers or spike
- Stream function: 2D array of size `(nx, ny)` with equation 2
- Grid spacing: `h=1. / (ny-1.)`

3. Compute the velocities in x and y direction from the stream function with **centered differences**:

$$u_{i,j} = \left(\frac{\partial \psi}{\partial y} \right)_{i,j} \approx \frac{\psi_{i,j+1} - \psi_{i,j-1}}{2h} \quad v_{i,j} = - \left(\frac{\partial \psi}{\partial x} \right)_{i,j} \approx - \frac{\psi_{i+1,j} - \psi_{i-1,j}}{2h} \quad (3)$$

4. Compute the time step from the maximum velocity in the model domain:

- Time step: `dt=MIN(a_diff*h**2/kappa, a_adv*h/vmax)`
- Number of time steps: `nsteps=INT(total_time/dt)`

5. Perform `nsteps` time steps with the following computations for all `i=2, nx-1` and `j=2, ny-1`:

- Compute $\nabla^2 T_{i,j}^n$ using the subroutine from Exercise 4.
- Write a function or subroutine that computes $\vec{v}_{i,j} \cdot \nabla T_{i,j}^n$ with the upstream scheme, e.g.:

$$\vec{v}_{i,j} \cdot \nabla T_{i,j}^n \approx u_{i,j} \cdot \frac{T_{i,j}^n - T_{i-1,j}^n}{h} + v_{i,j} \cdot \frac{T_{i,j}^n - T_{i,j-1}^n}{h} \quad \text{if } u_{i,j} > 0 \text{ and } v_{i,j} > 0 \quad (4)$$

(This needs to be adapted based on the wind direction.)

- Integrate forward in time:

$$T_{i,j}^{n+1} = T_{i,j}^n + \Delta t \cdot \left(\kappa \cdot \nabla^2 T_{i,j}^n - \vec{v}_{i,j} \cdot \nabla T_{i,j}^n \right) \quad (5)$$

- Use the boundary conditions $T = 1$ at $y = 0$ ($j=1$), $T = 0$ at $y = y_{max}$ ($j=ny$) and $\partial T / \partial x = 0$ at $x = 0$ ($i=1$) and $x = x_{max}$ ($i=nx$).

6. Run your program and test it.

- Use different values for B (e.g. 1, 10, and 100).
- After a sufficient integration time the system should reach a steady state, i.e. the temperature should not change any more. The steady state is independent of the initial conditions.
- Save the resulting temperature fields in a text file and plot them with your favorite plotting tool.

Deadline: Please hand in your solutions (.f90 files and plots) by **Tuesday, 23 April 2024, 23:59**.