Selected Topics in CFD - homework 1

Paweł Jędrejko

November 2025

Consider a velocity field $\mathbf{u}$ defined on the square domain

$$\Omega = [0, \tfrac{\pi}{2}] \times [0, \tfrac{\pi}{2}].$$

The field is decomposed into its curl-free part $\mathbf{u}_r$ and its divergence-free part $\mathbf{u}_d$:

$$\mathbf{u} = \mathbf{u}_d + \mathbf{u}_r.$$

The exact components are given by

$$\mathbf{u}_r(x, y) = \begin{bmatrix} \sin(x)\cos(y) \\ -\cos(x)\sin(y) \end{bmatrix}, \qquad \mathbf{u}_d(x, y) = \begin{bmatrix} \sin(x)\cos(y) \\ \cos(x)\sin(y) \end{bmatrix}.$$

Using only the numerical values of $\mathbf{u}$ in the interior of the domain and the normal component $\mathbf{u}_d \cdot \hat{n}$ on the boundary $\partial\Omega$, numerically reconstruct $\mathbf{u}_d$ and $\mathbf{u}_r$.

- For the purpose of this exercise, keep the discretized Laplacian in a (dense) `np.array`. Feel free to use `np.linalg.solve`, which can handle singular matrices
- Use 2nd order discretization on the staggered (Arakawa C) grid
- Implement the following set of functions (or methods)
  - `plotGrid` – plots the grid, marking nodes for u, v and scalars. For that purpose, use a smaller number of grid nodes to make the figure readable.
  - `pQuiver` – interpolates u and v to the scalar nodes and plots the vector field
  - `checkGauss` – checks if your discretization satisfies Gauss divergence theorem for the entire domain
  - `grad`, `div` – compute gradient and divergence of a given field
  - `lap` – returns the discretized Laplacian
- make the following plots:

  - example of a (small) grid
  - eomparison of $\boldsymbol{u}_d$ known a priori with the one determined numerically. Add a note about the largest difference
  - similar comparison for $\nabla \cdot \boldsymbol{u}_d$

- Try to keep the code clean and readable. Add comments if necessary
- Send the code and a pdf file with the plots to `p.jedrejko@uw.edu.pl`
- Deadline: end of 8.12.2025