EMCH 792 Project 2- Solving the Incompressible Navier-Stokes Equation for a Lid-Driven Cavity and Passive Scalar Field

Spencer Schwartz

1 Time Step computation

The time step size constraints are

$$\Delta t < \frac{\Delta x}{\mathrm{CFL}|u_{max}|} \tag{1}$$

and

$$\Delta t < \frac{(\Delta x)^2}{4\nu}. \tag{2}$$

Eq. (1) and (2) refer to the constraints caused by the convective and diffusive terms, respectively. After considering the preceding equations, we use a conservative $\Delta t = 0.005$ for most cases to maximize stability and reduce the iterations of the pressure solver. For the following cases, simulations were run until the velocity difference between the time steps reach a threshold value, 1E-8, or the time reaches the maximum time, $U_0 t_{max}/L = 20$.

2 Initial Conditions

The domain is a 1x1 square with 128x128 cells using no slip boundary conditions for $\mathbf{u}$ and symmetric boundary conditions for the tracer field, T and the initial tracer field is defined by

$$T(x, y, t = 0) = exp\left[-\left(\frac{x - 0.75}{0.025\pi}\right)^2 - \left(\frac{y - 0.75}{0.025\pi}\right)^2\right] \tag{3}$$

and is illustrated in Figure 1. The top wall, to simulate a moving lid, has a tangential velocity

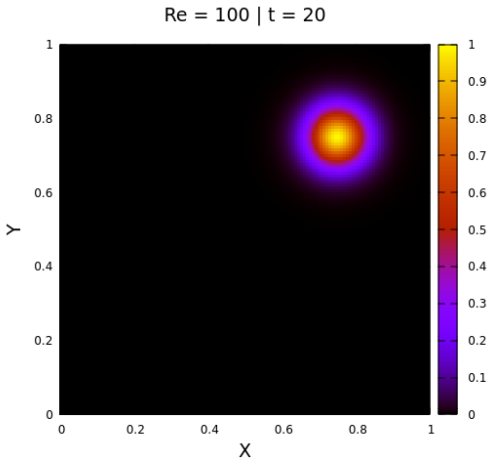

Figure 1: Initial tracer field at t = 0, used for all cases.

boundary condition of $U_0 = 1$. The domain and B.C's are summarized in Figure 2.

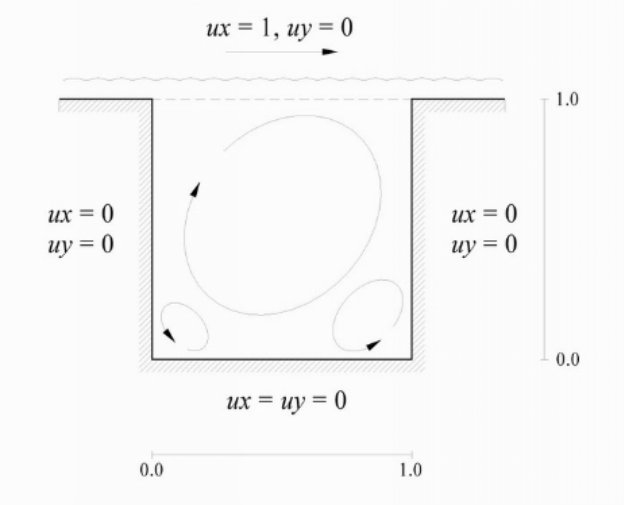

Figure 2: Initial tracer field at t = 0, used for all cases.

3 Results

For each case, $Re = 100, 400,$ and 1000, we test the 1st order upwind and 2nd order central method to solve the advection term in the N.S equation and tracer advection equation. For this project, the problem was solved using Basilisk and a custom C++ program.

3.1 Re = 100

First, we test our simulation code for a lid driven cavity at Re = 100. At this Re, we expect only small vortexes in the bottom left and right corners of the cavity and a velocity profile defined in a seminal paper by Ghia et al. 1982.

Figure 3 displays the horizontal and vertical centerline velocity profiles for each method. Good agreement is found between the Basilisk and C++ program. Furthermore, the upwind method under predicts the velocity profile in several regions, likely caused by the numerical diffusion introduced by the method. On the other hand, the 2^{nd} order central method agrees generally well with the reference.

The stream lines are also plotted, shown in Figure 4. Streamlines for the Basilisk results are neglected here because of the similarities with their C++ program counterpart. Again, the central scheme agrees much better than the upwind method: the minor corner vorticies are hardly formed in the upwind method while present with the central method.

The pressure fields for the central and upwind methods at Re = 100 are shown in Figure 5. The pressure fields are very similar to each other, except the upwind results feature a negative pressure region on the bottom wall.

Lastly, the passive tracer field for 4 different snapshots is displayed in Figure 6. After t = 5, the tracer field becomes very diffused and near homogeneously mixed throughout the domain.

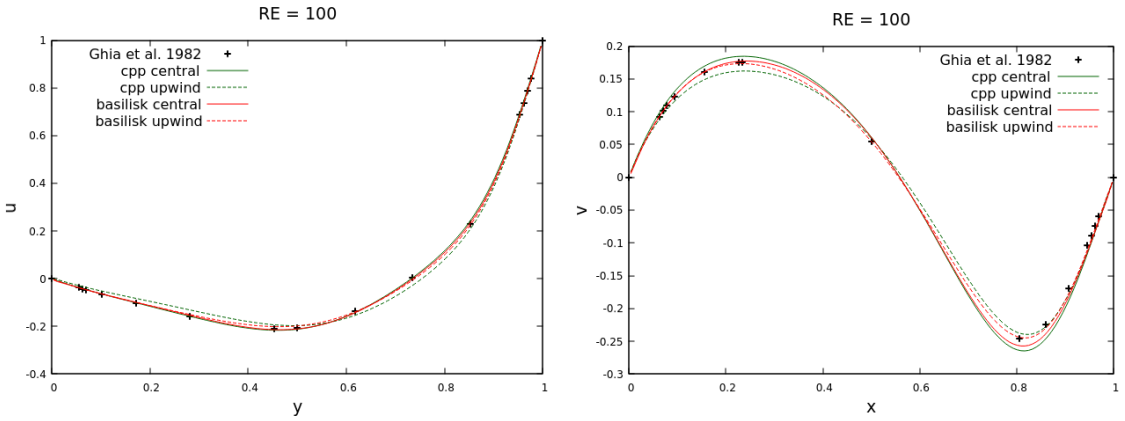

Figure 3: u velocity profile along a vertical line (left) and the v velocity profile along a horizontal line (right) for each program and each method (central —, upwind − − −) compared with results from Ghia et al. 1982 (black points).

3.2 Re = 400

Cases were then run for Re = 400. Again, the velocity profiles along centerline cuts are shown in Figure 7. As shown in Re = 100, the upwind results are smoothened from numerical diffusivity and do not agree as well as the central method. The Basilisk and C++ programs agree very well with Ghia's results and with each other.

Figure 8 shows streamlines for Re = 400. The corner vorticies are defined better using the central method and the main central vortex has the same overall structure as Ghia's. However, the bottom left vortex in the central method is not as defined as the reference, likely caused by plotting errors or the 1st order time stepping.

The pressure fields for the upwind and central methods are quite different, as shown in Figure 13. The central method's pressure field is more compartmentalized than the upwind results.

Lastly, the tracer field, compared to Re = 100, the tracer field does not diffuse as much due to the stronger vortex generated at a higher Re, as seen in Figure 10.

3.3 Re = 1000

The velocity profiles for horizontal and vertical centerlines are shown in Figure 11. The trend of the upwind method smoothening results continues, and the central method agrees well with Ghia's results. Basilisk and the C++ program have very similar results, so we will only look at the C++ program results in the following result comparisons.

Velocity streamlines for both methods and reference are shown in Figure 12. Again, the side vorticies for the central method are more prominent than the upwind results. Consequently, the central method agrees with Ghia's results much better than using the upwind method.

The pressure fields for the upwind and central method are shown in Figure 5. The central method case features a low pressure region in the center where negative vorticity is greatest.

The tracer field gets swirled more than previous Re's, as shown in Figure 14. This is likely caused by the stronger vortex formed at higher Re.

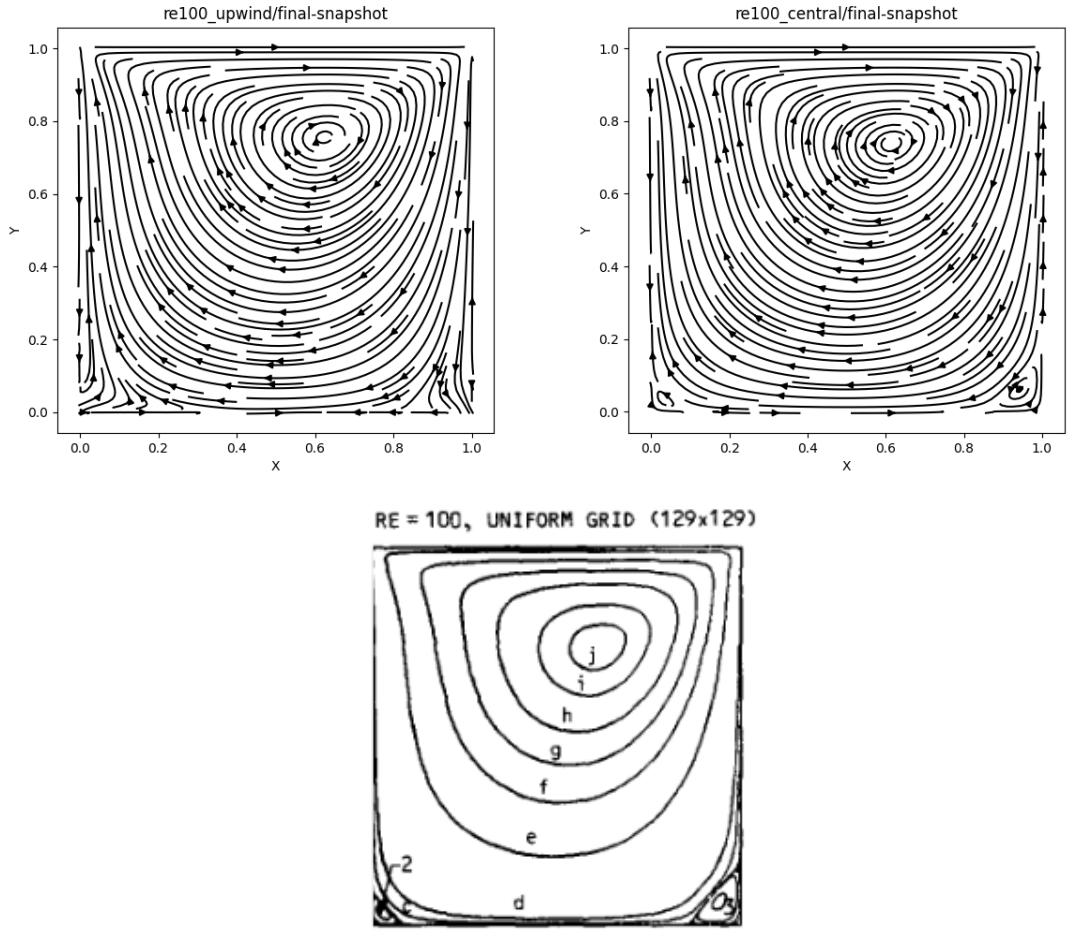

Figure 4: Velocity stream lines for the C++ program, upwind (left) and central (right), compared with that from Ghia's paper.

4 Summary

Simulations of a passive scalar field in a lid driven cavity are run by solving the Navier-Stokes equation using the projection method. The method is solved in a custom C++ program and the open-source CFD software Basilisk. In both programs, we used a first-order Forward Euler, 2nd order central method for the viscous term, and the Gauss-Siedel iterative method to solve the pressure-Poisson equation. The advection term is solved by using either the 1st order upwind method or the 2nd order central method. The pressure solver's residual tolerance is set to 1E-3 and a maximum allowable iterations of 500. Overall, both programs agree well with each other, with the central method achieving the best accuracy.

5 Resources

All code for this project can be found here:
https://github.com/SpencerSchwart/multiphase-cfd/tree/main/project2

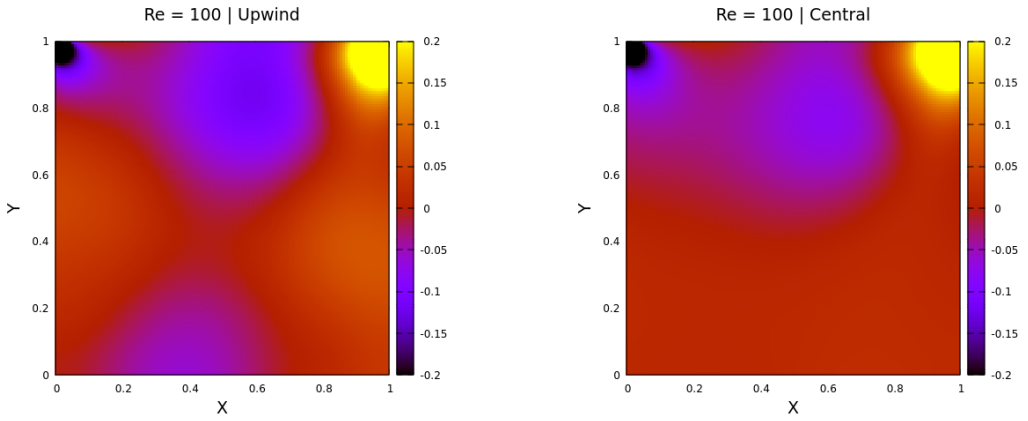

Figure 5: Pressure field for the C++ program, upwind (left) and central (right).

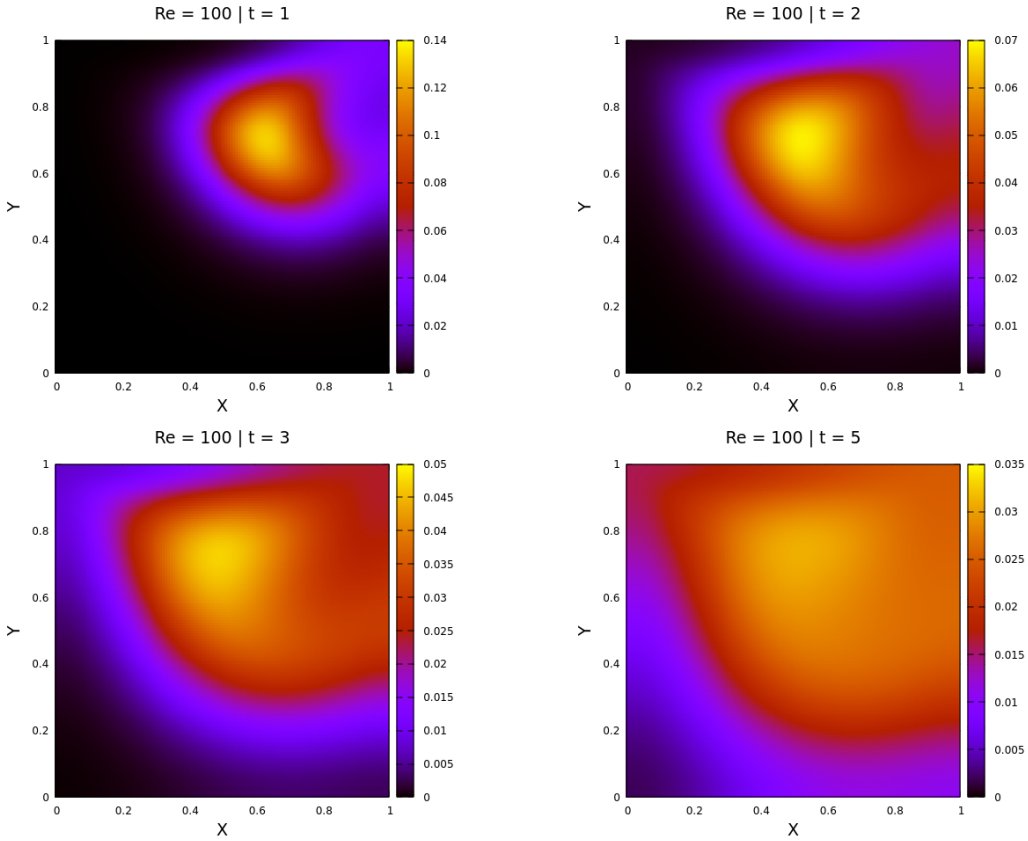

Figure 6: Tracer field at different time snapshots, t = 1, 2, 3, and 5. The color scale changes in each snapshot to account for the diffusing tracer field.

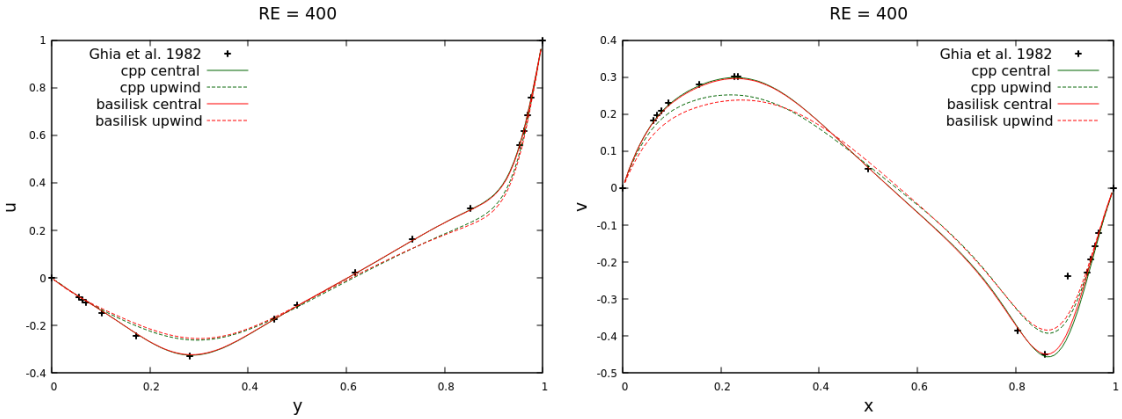

Figure 7: u velocity profile along a vertical line (left) and the v velocity profile along a horizontal line (right) for each program and each method (central —, upwind − − −) compared with results from Ghia et al. 1982 (black points).

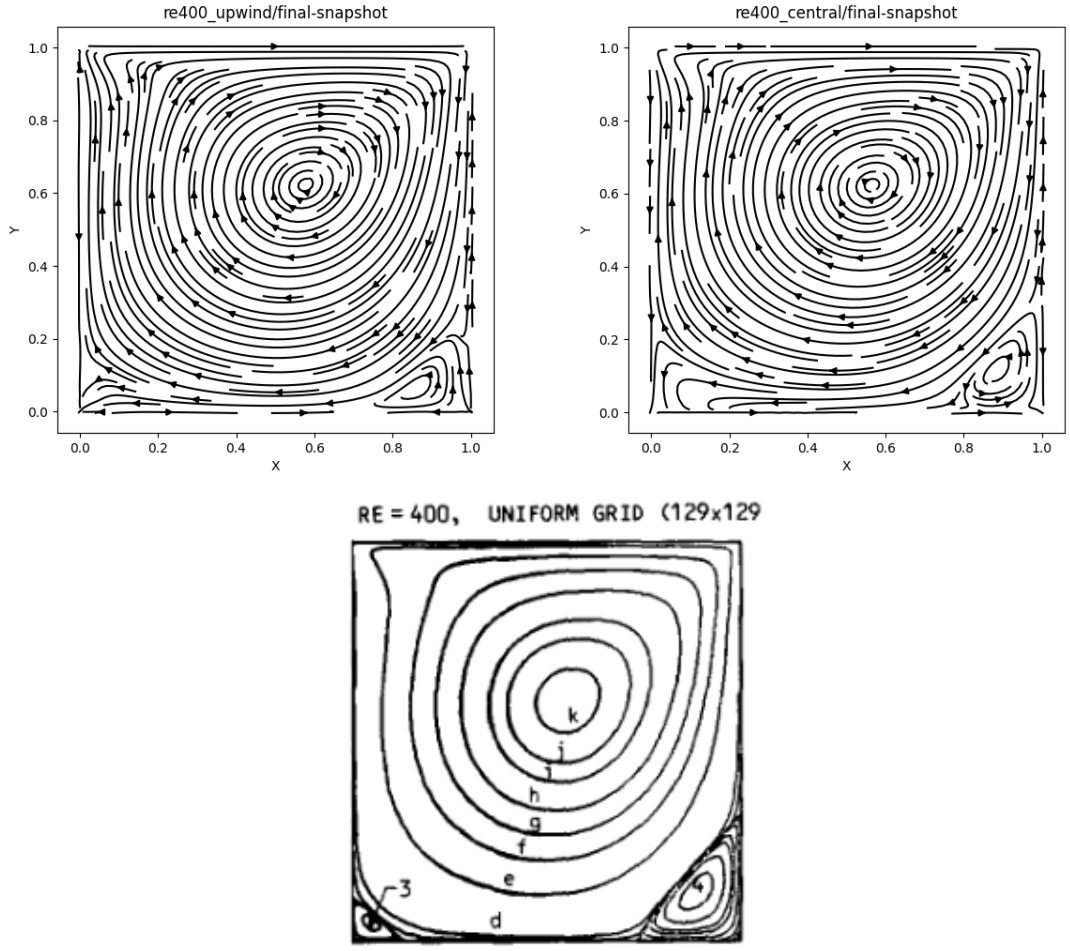

Figure 8: Velocity stream lines for the C++ program, upwind (left) and central (right), compared with that from Ghia's paper.

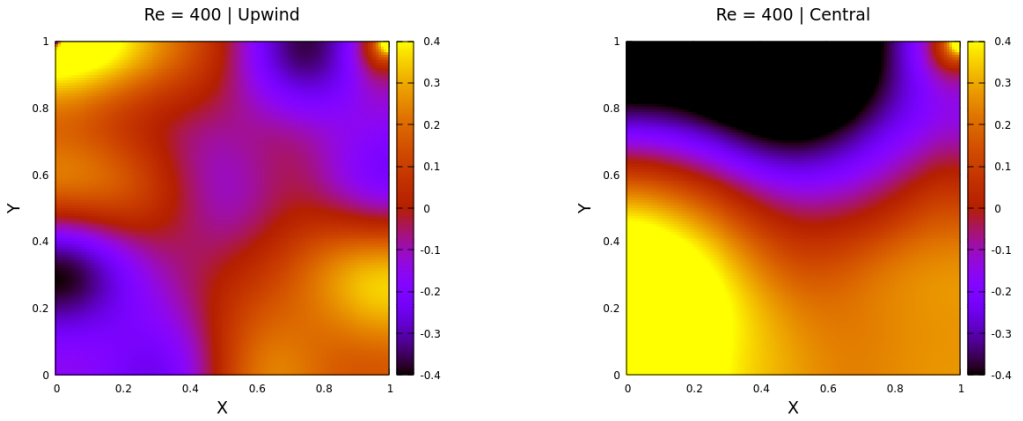

Figure 9: Pressure field for the C++ program, upwind (left) and central (right) at Re = 400.

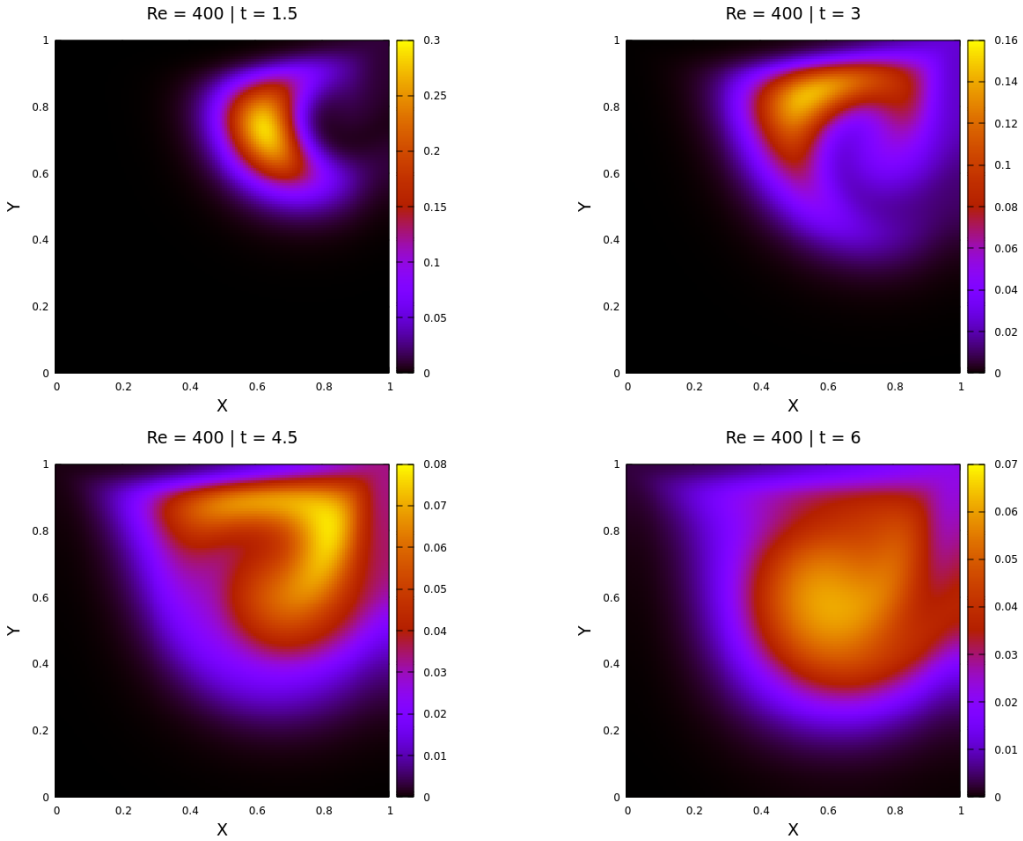

Figure 10: Tracer field at different time snapshots, t = 1, 2, 3, and 5. The color scale changes in each snapshot to account for the diffusing tracer field.

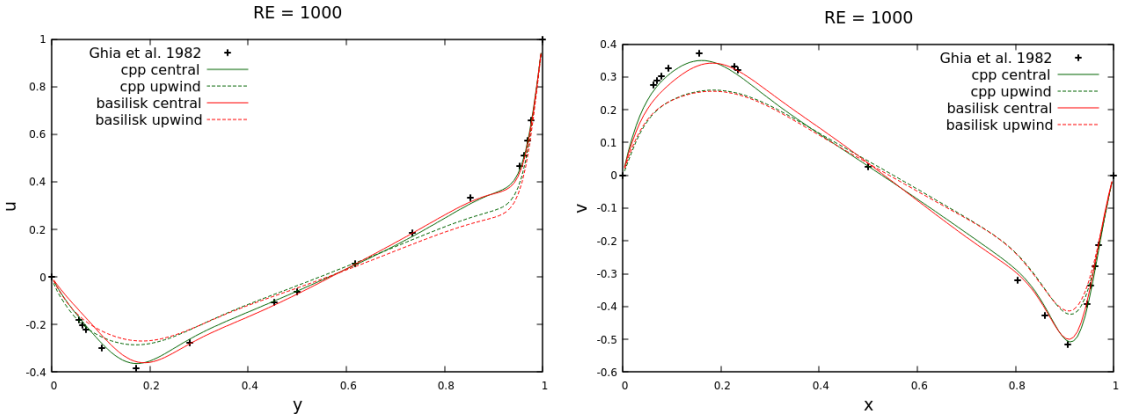

Figure 11: u velocity profile along a vertical line (left) and the v velocity profile along a horizontal line (right) for each program and each method (central —, upwind − − −) compared with results from Ghia et al. 1982 (black points).

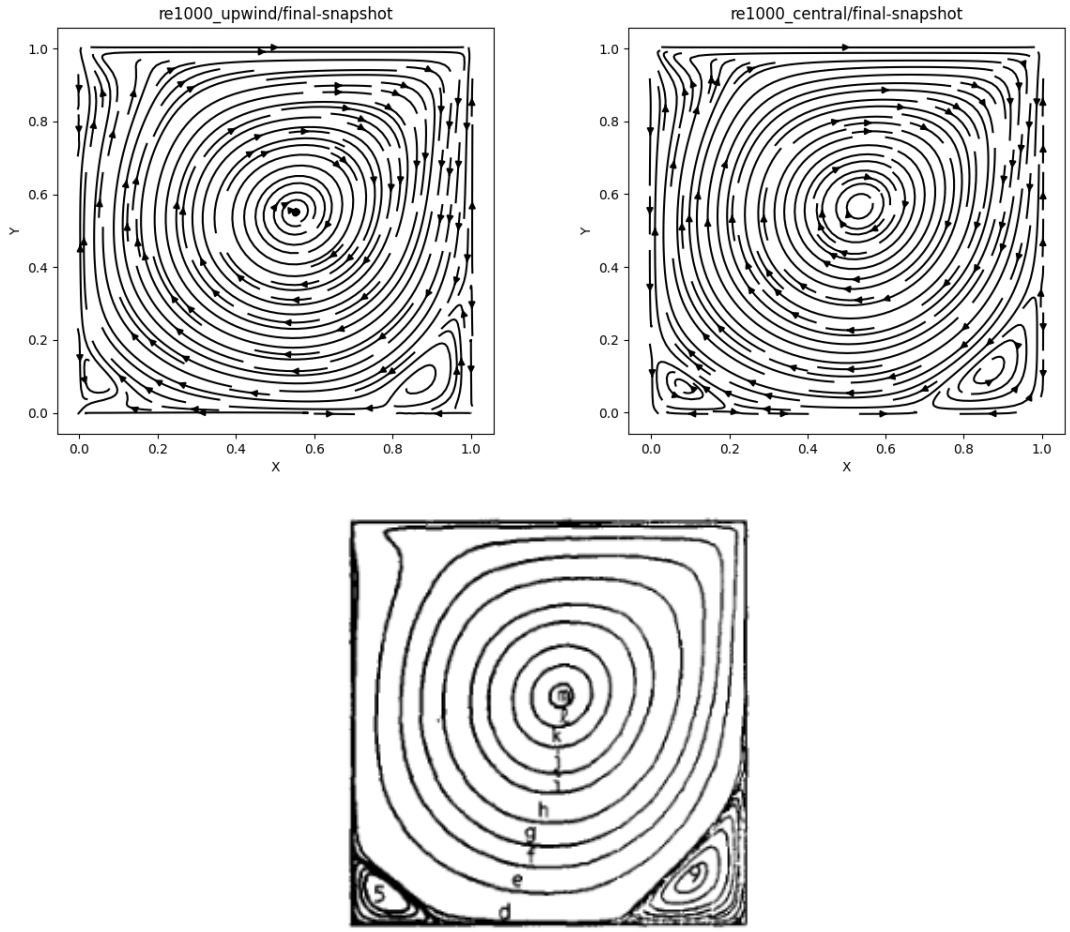

Figure 12: Velocity stream lines for the C++ program, upwind (left) and central (right), compared with that from Ghia's paper.

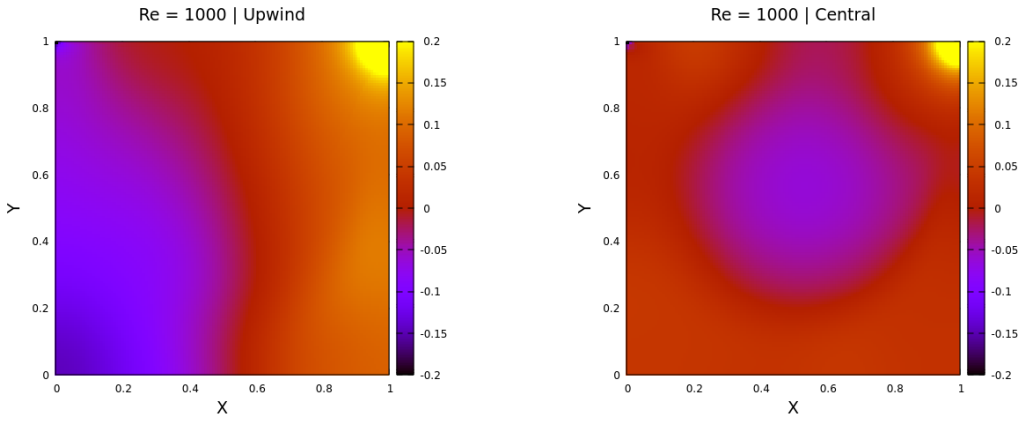

Figure 13: Pressure field for the C++ program, upwind (left) and central (right) at Re = 1000.

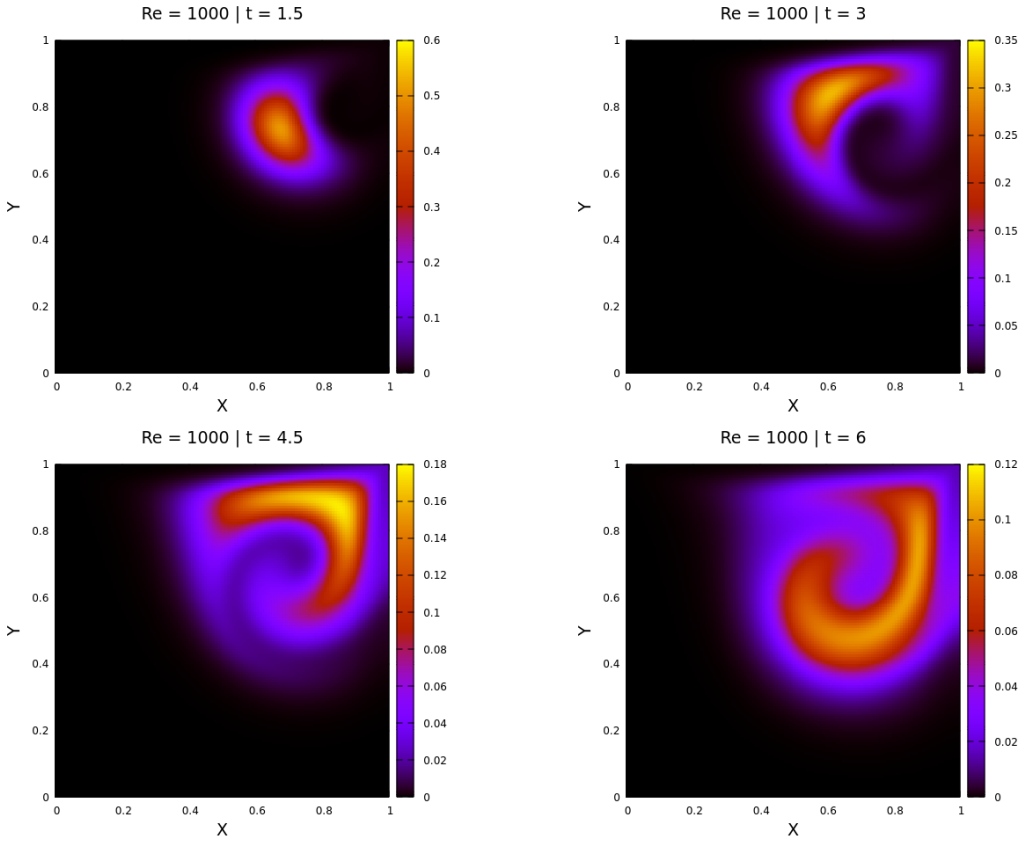

Figure 14: Tracer field at different time snapshots, t = 1, 2, 3, and 5. The color scale changes in each snapshot to account for the diffusing tracer field.