Fundamentals of CFD: Project report

Jan von Rickenbach

May 15, 2014

1 Introduction

The goal of this project is to solve the lid driven cavity problem using a stream function vorticity method. The problem is introduced in the methods section. The results section consists of various code verification studies and a comparison the the experimental results of a lid-driven cavity case for two different Re numbers.

2 Methods

2.1 Problem statement

For the two dimensional lid-driven cavity problem the following non-dimensional variables are defined:

$$\bar{x} = \frac{x}{L_{ref}}, \bar{y} = \frac{y}{L_{ref}}, \quad \bar{t} = \frac{tu_{ref}}{L_{ref}}, \bar{u} = \frac{u}{u_{ref}}, \quad \bar{v} = \frac{v}{v_{ref}}, \bar{\zeta} = \frac{\zeta L_{ref}}{u_{ref}}, \bar{\psi} = \frac{\psi}{u_{ref} L_{ref}}$$

ψ is the streamfunction and ζ is the vorticity. In this report $L_{ref} = 1m$ and $u_{ref} = v_{ref} = u_{lid} = 1m/s$ (the lid velocity of the cavity) are used as reference scales. For all the cases considered a square domain with sidelength 1 m starting at the origin is used. The streamfunction equation can be written as

$$\frac{\partial^2 \bar{\psi}}{\partial \bar{x}^2} + \frac{\partial^2 \bar{\psi}}{\partial \bar{y}^2} = -\bar{\zeta} \tag{1}$$

and the voriticity equation is

$$\frac{\partial \bar{\zeta}}{\partial \bar{t}} + \bar{u}\frac{\partial \bar{\zeta}}{\partial \bar{x}} + \bar{v}\frac{\partial \bar{\zeta}}{\partial \bar{y}} = \frac{1}{Re}\left(\frac{\partial^2 \bar{\zeta}}{\partial \bar{x}^2} + \frac{\partial^2 \bar{\zeta}}{\partial \bar{y}^2}\right) \tag{2}$$

where the velocities are given as:

$$\bar{u} = \frac{\partial \bar{\psi}}{\partial \bar{y}} \tag{3}$$

and

$$\bar{v} = -\frac{\partial \bar{\psi}}{\partial \bar{x}} \tag{4}$$

The boundary conditions are

$$\bar{u} = \bar{v} = 0 \tag{5}$$

for all the walls except the top wall where:

$$\bar{u} = 1 \tag{6}$$

$$\bar{v} = 0. \tag{7}$$

2.2 Numerical discretization

2.2.1 Streamfunction equation

The equations are discretized on a regular grid with grid spacing Δx and Δy which results in $nx = \Delta x/L$ and $ny = \Delta y/L$ grids points in the two coordinate directions. A discrete point is indicated with $x_i = i\Delta x$ and $y_j = j\Delta y$ and the notation

$$\phi_{i,j} = \phi(x_i, y_i) \tag{8}$$

is used. The second derivatives in the streamfunction equation are discretized with a centered second order scheme:

$$\left.\frac{\partial^2 \bar{\psi}}{\partial \bar{x}^2}\right|_{x_i,y_j} = \frac{\psi_{i-1,j} - 2\psi_{i,j} + \psi_{i+1,j}}{\Delta x^2} + O(\Delta x^2) \tag{9}$$

$$\left.\frac{\partial^2 \bar{\psi}}{\partial \bar{y}^2}\right|_{x_i,y_j} = \frac{\psi_{i,j-1} - 2\psi_{i,j} + \psi_{i,j+1}}{\Delta y^2} + O(\Delta y^2) \tag{10}$$

A homogeneous Dirchlet boundary condition for the stream function is used since

$$\bar{v} = \frac{\partial \bar{\psi}}{\partial \bar{x}} = 0 \tag{11}$$

on the top and bottom boundary and

$$\bar{u} = -\frac{\partial \bar{\psi}}{\partial \bar{y}} = 0 \tag{12}$$

on the left and right boundary. Therfore the stream function is a constant on the boundary which is chosen to be zero. This leads to a system of equations with $(nx-2)(ny-2)$ unknowns for the internal grid points.

2.2.2 Vorticity equation

The convective terms in the vorticity equation are discretized as follows:

$$\bar{u}_{i,j} = -\frac{\bar{\psi}_{i,j+1} - \bar{\psi}_{i,j-1}}{2\Delta y} + O(\Delta y^2) \tag{13}$$

$$\bar{v}_{i,j} = \frac{\bar{\psi}_{i+1,j} - \bar{\psi}_{i-1,j}}{2\Delta x} + O(\Delta x^2) \tag{14}$$

$$\left.\frac{\partial \bar{\zeta}}{\partial \bar{x}}\right|_{i,j} = \frac{\bar{\zeta}_{i+1,j} - \bar{\zeta}_{i-1,j}}{2\Delta x} + O(\Delta x^2) \tag{15}$$

$$\left.\frac{\partial \bar{\zeta}}{\partial \bar{y}}\right|_{i,j} = \frac{\bar{\zeta}_{i,j+1} - \bar{\zeta}_{i,j-1}}{2\Delta y} + O(\Delta y^2) \tag{16}$$

The diffusive term in the vorticity equation is discretized using second order finite differences

$$\left.\frac{\partial^2\bar{\zeta}}{\partial\bar{x}^2}\right|_{x_i,y_j} = \frac{\zeta_{i-1,j} - 2\zeta_{i,j} + \zeta_{i+1,j}}{\Delta x^2} + O(\Delta x^2) \tag{17}$$

$$\left.\frac{\partial^2\bar{\zeta}}{\partial\bar{y}^2}\right|_{x_i,y_j} = \frac{\zeta_{i,j-1} - 2\zeta_{i,j} + \zeta_{i,j+1}}{\Delta y^2} + O(\Delta y^2) \tag{18}$$

The boundary condition of the vorticity equation depends on the streamfunction. Since the streamfunction on the boundary is constant:

$$\frac{\partial^2\bar{\psi}}{\partial\bar{x}^2} = -\bar{\zeta} \tag{19}$$

on the left and right boundary and

$$\frac{\partial^2\bar{\psi}}{\partial\bar{y}^2} = -\bar{\zeta} \tag{20}$$

on the top and bottom boundary. For top boundary $\bar{u} = 1$ and $\bar{v} = 0$. Therefore

$$\frac{\partial\bar{\psi}}{\partial\bar{y}} = 1 \tag{21}$$

Discretizing the first derivative this can be approximated as

$$\frac{\bar{\psi}_{i,ny+1} - \bar{\psi}_{i,ny-1}}{2\Delta y} = 1 \tag{22}$$

where $ny + 1$ indicates a ghost point outside of the domain. Solving for the ghost point results in

$$\bar{\psi}_{i,ny+1} = 2\Delta y + \bar{\psi}_{i,ny-1} \tag{23}$$

Discretizing Eq. 20, using the fact that the streamfunction is zero on the boundary together with Eq. 23 the vorticity on the boundary can be written as

$$\bar{\zeta}_{i,ny} = -\frac{2(\Delta y + \bar{\psi}_{i,ny-1})}{\Delta y^2} \tag{24}$$

The boundary condition for the remaining three boundaries can be determined analogously. Since on these boundaries both velocities are zero the Δy term in the nominator disappears: For the bottom boundary

$$\bar{\zeta}_{i,0} = -\frac{2(\bar{\psi}_{i,1})}{\Delta y^2} \tag{25}$$

the left boundary

$$\bar{\zeta}_{0,j} = -\frac{2(\bar{\psi}_{1,j})}{\Delta x^2} \tag{26}$$

and the right boundary

$$\bar{\zeta}_{nx,j} = -\frac{2(\bar{\psi}_{nx-1,j})}{\Delta x^2}. \tag{27}$$

2.3 Solution streamfunction equation

Two different methods are used to solve the streamfunction equation: Successive over-relaxation (SOR) and the Alternate Direction Implicit Method (ADI).

2.3.1 SOR method

In the SOR method one loops trough all the grid-points and updates the values with

$$\bar{\psi}_{i,j}^{n*} = \frac{(\bar{\psi}_{i+1,j}^{n} + \bar{\psi}_{i-1,j}^{n})\Delta y^2 + (\bar{\psi}_{i,j+1}^{n} - \bar{\psi}_{i,j-1}^{n})\Delta x^2 + \bar{\zeta}_{i,j}\Delta x^2 \Delta y^2}{2(\Delta x + \Delta y)} \tag{28}$$

and

$$\bar{\psi}_{i,j}^{n+1} = \alpha \bar{\psi}_{i,j}^{n*} + (1-\alpha)\bar{\psi}_{i,j}^{n}. \tag{29}$$

where α is an over-relaxation coefficient. The superscript indicates the iteration. The iteration stop if the error

$$e = max(abs(\bar{\psi}_{i,j}^{n+1} - \bar{\psi}_{i,j}^{n}) \tag{30}$$

is smaller than a prescribed iteration tolerance t. The maximum error is taken over all the grid points.

2.3.2 ADI method

In the ADI method the solution is split into two sub-steps, which are repeated until the iteration tolerance is sufficiently small. An artificial time term is added to the equation and the unsteady equation is discretized using two sub steps:

$$u_{i,j}^{n+1/2} = u_{i,j}^{n} + \quad \frac{d_x}{2}\left(u_{i+1,j}^{n+1/2} - 2u_{i,j}^{n+1/2} + u_{i-1,j}^{n+1/2}\right) \tag{31}$$

$$+ \quad \frac{d_y}{2}\left(u_{i,j+1}^{n} - 2u_{i,j}^{n} + u_{i,j-1}^{n}\right) + f_{i,j}\frac{\Delta t}{2} \tag{32}$$

and

$$u_{i,j}^{n} = u_{i,j}^{n+1/2} + \quad \frac{d_x}{2}\left(u_{i+1,j}^{n+1/2} - 2u_{i,j}^{n+1/2} + u_{i-1,j}^{n+1/2}\right) \tag{33}$$

$$+ \quad \frac{d_y}{2}\left(u_{i,j+1}^{n+1} - 2u_{i,j}^{n+1} + u_{i,j-1}^{n+1}\right) + f_{i,j}\frac{\Delta t}{2} \tag{34}$$

with $d_y = \Delta t/\Delta y^2$, $d_x = \Delta t/\Delta x^2$ and u as an arbitrary scalar. For each sub-step a tridiagonal system of equations has to be solved. The tridiagonal system is inverted using the Thomas algorithm. The ADI method combines the advantages of explicit and implicit time-stepping methods. It is unconditionally stable for linear equations as the implicit methods and cheap per time-step, since only a tridiagonal matrix has to be inverted (as explicit methods). For the streamfunction equation the time term is artificial and the goal is to reach steady state as quick as possible. It turns out that the most efficient way to achieve the steady state is using the following sequence

$$\rho = 4\cos^2(0.5\pi h)\left[\tan^2(0.5\pi h)\right]^{\frac{2p-1}{2p_{max}}} \text{for} p = 1, 2, 3, ..., p_{max} \tag{35}$$

with $d_x = d_y = 1/\rho$, $\Delta x = \Delta y = h$ and

$$p_{max} = \frac{\log \tan^2(0.5\pi h)}{2\log(\sqrt{2}-1)} \tag{36}$$

2.4 Solution of the vorticity equation

The vorticity equation is integrated in time using a forward euler method. The discretized convective (C) and the diffusive terms (D) are pushed to the right hand side.

$$\frac{\partial \bar{\zeta}}{\partial \bar{t}} = D - C \tag{37}$$

The forward euler method is then written as

$$\bar{\zeta}_{i,j}^{n+1} = \bar{\zeta}_{i,j}^{n} + (D - C)_{i,j}\Delta \bar{t} \tag{38}$$

2.5 Combined solution of the streamfunction and the vorticity equation

The system of equations is integrated in time as follows

1. Initialize solution
2. Set boundary conditions for the vorticity equation
3. Integrate vorticity equation in time using the forward Euler method
4. Solve stream function equation with ADI or SOR
5. Go to 2 until endtime is reached

3 Results

3.1 Code verification

3.1.1 Stream function

To verfiy the correct implementation of the stream function equation the method of manufactured solutions is used. The following manufactured solution is used:

$$\bar{\psi}_{m1}(x, y) = \sin(\pi x)\sin(\pi y). \tag{39}$$

By substituting Eq. 39 into the stream-function equation we obtain the corresponding function for the vorticity that satisfies the differential equation:

$$\bar{\zeta}_{m1}(x, y) = 2\pi^2 \sin(\pi x)\sin(\pi y) \tag{40}$$

As an initial condition the stream function is set to one everywhere except the bounndaries, where it is set to zero. The solution of the stream-function is obtained using the SOR and the ADI method as described in the methods section. The solution is obtained with different iteration tolerances, in order to study the impact of the iteration tolerance on the numerical solution. In Fig. 1 the maximum error defined as

$$e_{max} = max(abs(\bar{\psi}_{m1} - \bar{\psi})) \tag{41}$$

is shown for the two methods with different iteration tolerances.

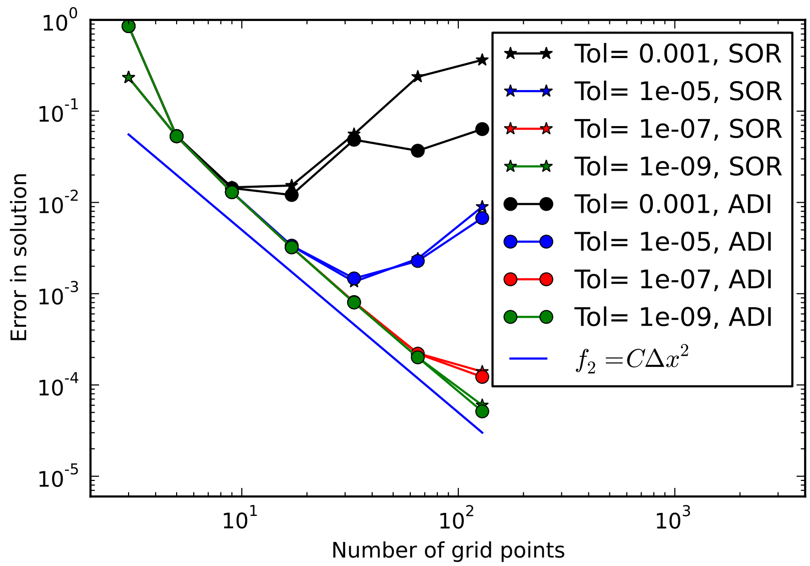

Figure 1: Error of numerical solution compared to the manufactured solution SOR and ADI with different iteration tolerances

The function

$$f_2 = C\Delta x^2 \tag{42}$$

is shown as reference. The constant C is chosen such that f_2 lies close to the obtained solutions and can be used to check second order decrease of the error. Figure 1 shows that the leading error term is second order for sufficently fine grids. If a large iteration tolerance is chosen, the iteration error becomes dominant for the finer grids with low discretization errors. Behaviour of the two methods studied (ADI and SOR) is similar. The error in the two methods is different for large iteration errors since the iteration error depends on the method. In the range where the iteration error is small the error in the two methods is essentially the same. This is expected since the numerical method used to solve the linear system is only reflected in the iteration error. For very fine grids it is expected, that truncation errors become significant. This was not observed for the current verification study, probably because the grids are not fine enough and the disretization error is still much larger than the truncation error.

3.1.2 Vorticity

In order to verify the implementation of the vorticity equation the same function for the vorticity as above for the stream function is chosen.

$$\bar{\zeta}_{m2}(x,y) = \sin(\pi x)\sin(\pi y) \tag{43}$$

and for the stream-function

$$\bar{\psi}_{m2}(x,y) = y - x \tag{44}$$

is used which results in a velocity of one for the two velocity components. The Re number is set to 1 and all the term except the time derivate are pushed to the right hand side. With the manufactured solution we obtain the right hand side:

$$R(x,y) = -\pi(\cos(\pi x)\sin(\pi y) + \sin(\pi x)\cos(\pi y) - 2\pi^2\sin(\pi x)\sin(\pi y) \tag{45}$$

We add a source term S to the right hand side of the vorticity equation to cancel the contribution of the convective and the diffusive term:

$$S = -R \tag{46}$$

If the vorticity equation is integrated for a single time step with $\Delta\bar{t} = 1$ with the manufactured solution as the initial condition on should obtain the unchanged initial condition. Figure 2 shows the error of the solution at $\Delta\bar{t} = 1$ compared to the manufactured solution $\bar{\psi}_{m2}$. No iteration tolerance is involved and the time disretization error is zero since the solution remains constant in the time interval considered. Therefore the dominating error term is the discretization error. The leading error of the discretization error is second order which can be seen in the plot. Again truncation errors are not observed even for the finest grids considered.

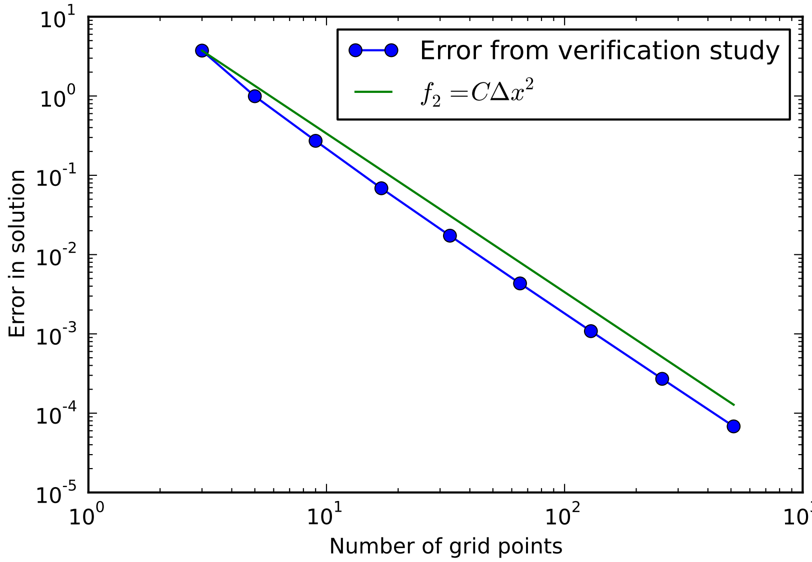

Figure 2: Error in the advection diffusion terms

3.2 Efficiency of the ADI and the SOR method

Here the efficiency of the two methods to solve the stream-function equation is compared. In order to determine the optimal relaxation parameters the same problem as for the stream function verification is run on an grid with 65x65 points. Figure 3 shows the time to solution for different relaxation factors with the SOR method. For relaxation factors larger than roughly 1.9 the iteration process becomes unstable and the solver does not converge anymore. To stay on the save side a relaxation factor or 1.8 is used for all the simulations using the SOR-method.

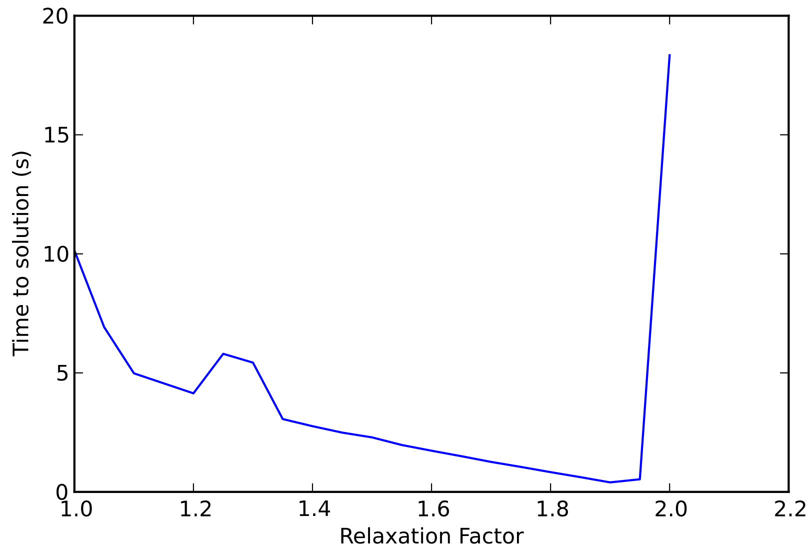

Figure 3: Time to solution for different relaxation parameters (SOR-method)

Figure 4 shows runtimes for the the SOR and the ADI method on the verification problem for the stream-function equation. The grid and the iteration tolerance were varied independently. The solution of the stream-function equation was repeated ten times and the total time is shown. Repeated tests are necessary to improve the accuracy of the timings since the timings become very inaccurate for short runtimes below 0.01 seconds. The plot shows that for small grids the the runtimes for the two methods are not very different and no clear trend is visible as of which method is faster. However for finer grids the ADI method is clearly more efficient. Therefore the ADI method is the better choice for demanding simulations.

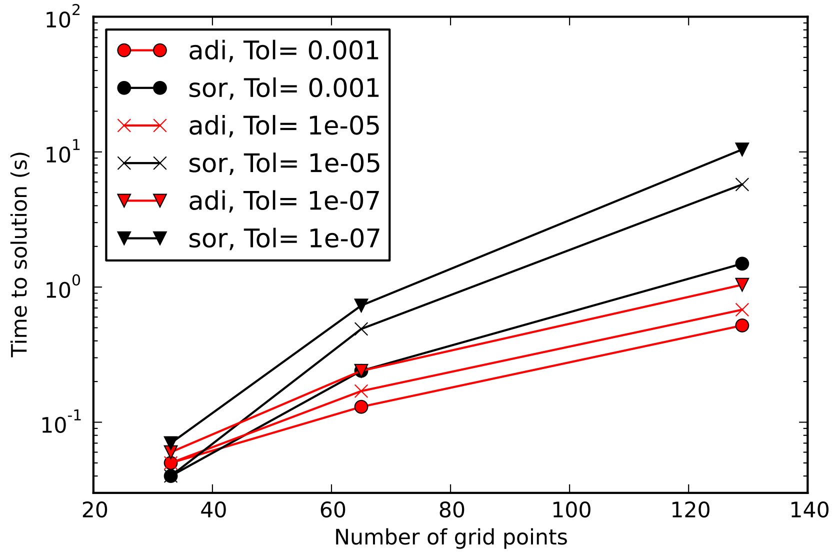

Figure 4: Time to solution for different grids and tolerances (ADI and SOR method)

3.3 The lid driven cavity

The lid-driven cavity problem as introduced in the methods section is solved using the ADI method. The problem is solved for two different Re numbers: 100 and 1000. To estimate the timestep the two relevant non-dimensional numbers are considered: The CFL and the Diffusion number

$$CFL = \frac{\Delta \bar{t} \bar{u}_{ref}}{\Delta x} \tag{47}$$

$$DIF = \frac{\Delta \bar{t} \nu}{\Delta x^2} \tag{48}$$

We can rewrite the diffusion number as

$$DIF = \frac{\Delta t \bar{u}_{ref} nx}{Re \Delta x} \tag{49}$$

A conservative way to choose a time step could be

$$\Delta t = \frac{\Delta x}{\bar{u}_{ref}} min \left(CFL, \frac{DIF Re}{nx} \right) \tag{50}$$

Due to the non-linearity of the Navier-Stokes equations, it is difficult to determine the exact stability region for the numerical disretization. Numerical experiments have shown that a DIF=CFL=0.05 leads to a stable solution process for the Re numbers considered. The simulations were run for 100000 timesteps in order to reach a steady state. Comparison with shorter runs showed no change in the solution and therefore a steady state was confirmed. Figures 5 - 8 show the u-velocity at $\bar{y} = 0.5$ and the v-velocity along $\bar{x} = 0.5$ for the two Re numbers considered. The figures show the velocity profiles obtained with different grid resolutions. The experimental data of Ghia et al. is shown as dots. The plot clearly shows that discretization errors and iteration errors are very small since it is hardly possible to distinguish the solution for the two finest grids. A finer grid is needed to obtain small errors for the higher Re number case compared to the lower Re number case. This is most probably due to the thinner boundary layers in the high Re number case, which have to be resolved with the numerical grid. The overall match with the experimental result is very good. Deviations are very small and probably lie within the experimental errors. Modelling errors are expected to be small since the Navier Stokes equations are a very accurate description of this laminar flow.

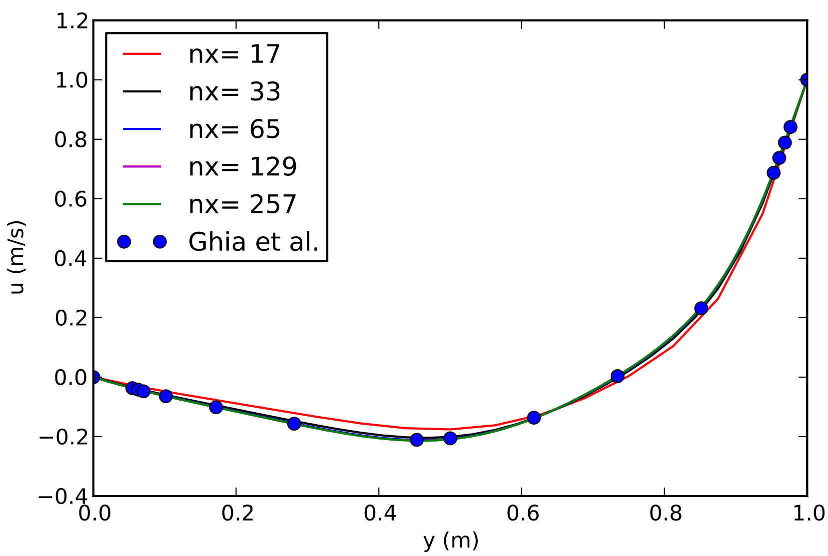

Figure 5: Comparison of the u-velocity along the vertical centerline at x=0.5 with the reference solution of Ghia et al.. Re=100

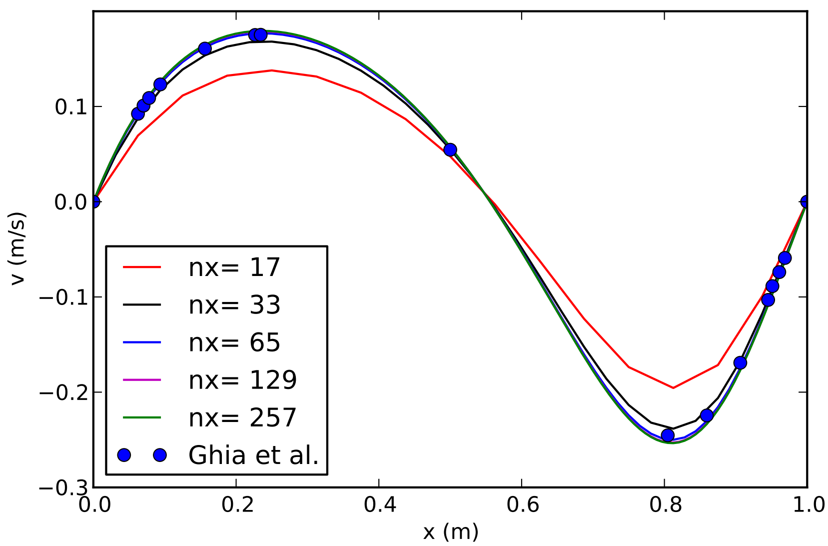

Figure 6: Comparison of the v-velocity along the horizontal centerline at y=0.5 with the reference solution of Ghia et al.. Re=100

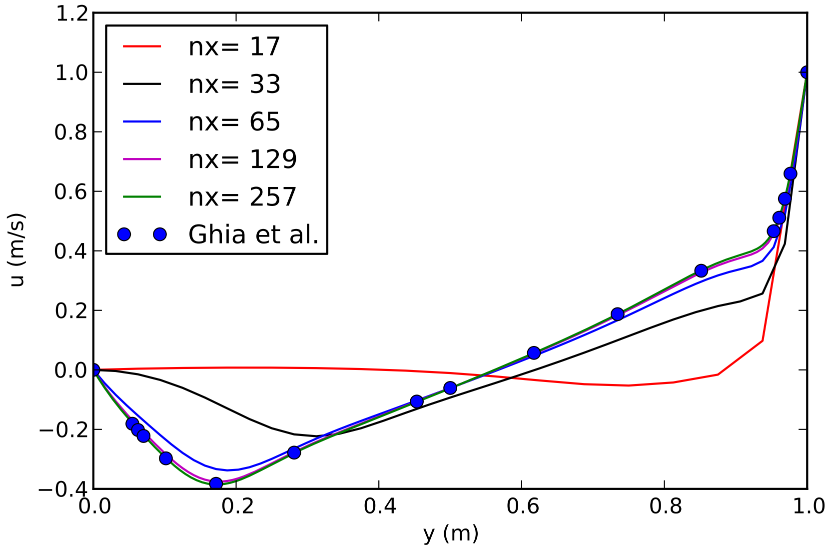

Figure 7: Comparison of the u-velocity along the vertical centerline at x=0.5 with the reference solution of Ghia et al.. Re=1000

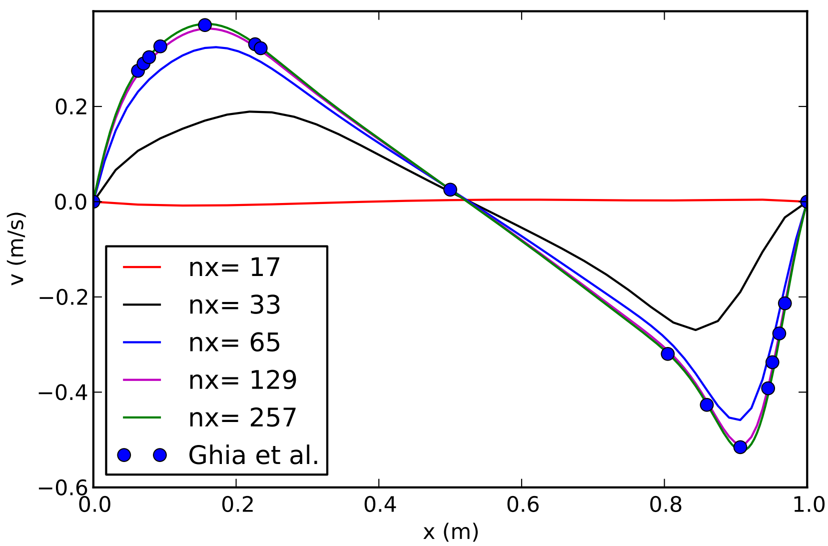

Figure 8: Comparison of the v-velocity along the horizontal centerline at y=0.5 with the reference solution of Ghia et al.. Re=1000

4 Discussion of the project

The project clearly highlighted the points that are important in developing a CFD code:

- To have a clear idea of the overall structure of the code before starting to write code. Write down difficult algorithms on paper before starting to code them.
- Split a task into sub-tasks and write function/classes to achieve the tasks.
- Test the functions/classes for a simple case where input and output are clearly defined.
- Use the simplest possible case to debug a problem
- Document code when while writing it

If I had to do the project again I would use a something like a lab notbook to write the ideas and derivations down while coding. Writing the report I realized that I had to look up some equations in the code, because I lost the hand-written documentation. I would also use a task list. I often started to code and forgot some parts that I had planned initially which caused to code to fail.