Assignment CFD II

Navier-Stokes Solver using Incidence & Hodge matrices

AE4136: CFD II

Prajwal Deval & Emmanouil Vretoudakis

Delft University of Technology

Assignment CFD II

Navier-Stokes Solver using Incidence & Hodge matrices

by

Prajwal Deval & Emmanouil Vretoudakis

Student Name	Student Number
Prajwal Deval	4811526
Emmanouil Vretoudakis	4799216

Instructor: M. Gerritsma
Faculty: Faculty of Aerospace Engineering, Delft

Contents

1 Introduction **1**

2 Method **2**
2.1 Equations & Variables . . . 2
2.2 Incidence Matrices . . . 3
2.2.1 Primal matrix 21 . . . 3
2.2.2 Dual matrix 10 . . . 6
2.2.3 Primal Matrix 10 . . . 6
2.2.4 Dual matrix 21 . . . 7
2.3 Hodge Matrices . . . 7
2.4 Tangential Velocity Boundary Condition . . . 8

3 Results **12**
3.1 Streamfunction . . . 12
3.2 Vorticity . . . 14
3.3 Pressure . . . 16
3.4 Flow center-lines . . . 18

4 Discussions **27**
4.1 Timestep analysis . . . 27
4.2 Tolerance analysis . . . 30
4.3 Pressure matrix singularity . . . 32
4.4 Integrated vorticity . . . 33

1 Introduction

The goal of this assignment is to develop a numerical solver for the Navier-Stokes equations, which describe fluid motion on the so called ”lid-driven cavity flow”. Lid-driven cavity flow refers to a type of fluid flow in a rectangular cavity where the motion of the fluid is driven by the movement of a solid boundary, typically the lid of the cavity. In this flow configuration, the fluid is confined within the cavity, and the lid is used to impose a velocity on the fluid, either through continuous motion or by applying a constant velocity boundary condition. The solver will utilize incidence matrices and hodge matrices, mathematical tools that offer a more efficient and flexible approach for solving the Navier-Stokes equations. Note that for this assignment, the flow will be solved on a unit square.

The primary objectives of this assignment are as follows:

- Understand the Navier-Stokes equations and their relevance in the context of this assignment.
- Investigate the use of incidence matrices as representations of vector operations.
- Utilize Hodge matrices for transitioning between primal and dual grids, enabling representation of unknowns on both grids.
- Formulate the Navier-Stokes equations using the incidence and Hodge matrices, incorporating boundary conditions.
- Implement a numerical solver based on the developed formulation.
- Visualise the results of the solver and compare it with reference solutions found in literature.
- Discuss the results and reach conclusions.

The structure of the report is as follows. Firstly the method used for setting up the solver is presented in chapter 2 based on the information given in the assignment description. Then the results of the solver are presented in chapter 3, mostly in the form of figures. Finally, in chapter 4 some discussions are presented concerning certain material involved in the solver.

2

Method

2.1. Equations & Variables

The resulting system of equations to be solved as presented in the assignment description presented below.

$$\nabla \cdot \vec{u} = 0 \tag{2.1}$$

$$\xi = \nabla \times \vec{u} \tag{2.2}$$

$$\frac{\partial \vec{u}}{\partial t} - \vec{u} \times \vec{\xi} + \nabla P + \frac{1}{Re} \nabla \times \vec{\xi} = 0 \tag{2.3}$$

The vector convention used throughout the entire calculation process is shown below, with u and v representing the horizontal and vertical velocities respectively, both using the left-to-right and then bottom-to-top single increasing indexing.

$$\vec{u} = \begin{pmatrix} \tilde{u}_1 \\ \tilde{u}_2 \\ \tilde{u}_3 \\ \tilde{u}_4 \\ \vdots \\ \tilde{v}_1 \\ \tilde{v}_2 \\ \tilde{v}_3 \\ \vdots \end{pmatrix} \tag{2.4}$$

This velocity vector, the entries of which lie on the dual grid, can be translated to the vorticity describing the surfaces on the dual grid using the incidence matrix which represents the curl operator on the dual, inner-oriented grid. This can be mathematically interpreted using Stokes theorem which relates the circulation of a vector field around a closed curve to the flux of its curl through a surface bounded by that curve. It establishes a connection between line integrals and surface integrals, providing a way to evaluate the circulation of a vector field using the curl of the field over a surface.

$$\oint_{\partial S} \mathbf{F} \cdot d\mathbf{r} = \iint_S (\nabla \times \mathbf{F}) \cdot d\mathbf{S} \tag{2.5}$$

We see that replacing the placeholder F with the velocity field u, we get a direct connection between the velocity on the edges and the vorticity over the surfaces of the dual grid.

Physically, this curl of the velocity called vorticity (denoted by the symbol ξ or ω) represents the local rotation of fluid particles in the flow. It is a measure of the tendency of the fluid elements to rotate about their respective axes and the magnitude of the vorticity indicates the strength of the rotation at a given

point in the fluid. By analyzing the vorticity distribution in the flow field, we can identify regions of high turbulence or low stability, and predict the formation of vortices, separation of flow, or the occurrence of turbulence.

In the context of this assignment, the below representaion of the curl is used to practically calculate the vorticity given the velocity field and the results will be shown in later sections.

$$\xi = \mathbb{E}^{(2,1)} \times \vec{u} \tag{2.6}$$

More information on the incidence matrices is given in the next section.

The full discretized system is presented below.

$$\tilde{\mathbb{E}}^{(2,1)} H^{\tilde{1},1} \vec{u}^{n+1} + \vec{u}_{norm} = 0 \tag{2.7}$$

$$\xi^n = \mathbb{E}^{(2,1)} \vec{u}^n \tag{2.8}$$

$$\frac{\vec{u}^{n+1} - \vec{u}^n}{\Delta t} + \text{ convective }^n + \mathbb{E}^{(1,0)} P^{n+1} + \frac{1}{Re} H^{1,\bar{I}} \tilde{\mathbb{E}}^{(1,0)} H^{0,2} \left(\xi^n + \xi_{\text{prescr}}\right) = 0 \tag{2.9}$$

which represents a system that solves for the time instant n+1.
Now, in order to solve the system ensuring that the newly calculated velocity fits the conservation of mass too, the following system for the pressure calculation is derived (basic principle being the application of continuity through matrices $\tilde{\mathbb{E}}^{(2,1)} H^{\tilde{1},1}$ inside the momentum equation).

$$A\vec{P} = f \tag{2.10}$$

with

$$A = -\tilde{\mathbb{E}}^{(2,1)} H^{\tilde{1},1} \mathbb{E}^{(1,0)} \tag{2.11}$$

and

$$f = \tilde{\mathbb{E}}^{(2,1)} H^{\tilde{1},1} \left[\frac{\vec{u}^n}{dt} - \text{ convective }^n - \frac{1}{\text{Re}} H^{1,\tilde{1}} \tilde{\mathbb{E}}^{(1,0)} H^{\tilde{0},2} \left(\xi^n + \xi_{\text{pres}}\right) \right] + \frac{\vec{u}_{norm}}{\Delta t} \tag{2.12}$$

2.2. Incidence Matrices

The incidence matrices that were used in the solver are the following: $\tilde{\mathbb{E}}^{(2,1)}$, $\tilde{\mathbb{E}}^{(1,0)}$, $\mathbb{E}^{(2,1)}$ and $\mathbb{E}^{(1,0)}$. In this section, the programmatic construction of these matrices is shown (for any given value for N). This is done through the use of examples on a grid corresponding to N=3 which is shown with the numbering of the edges, points and volumes (dual and primal) in Figure 2.1.

- For the primal edges, a rightward flux or an upward flux is positive.
- For the dual edges, a rightward or upward tangential velocity is positive.
- The primal volumes are sources and dual points are sinks.
- The dual volumes and primal points are clockwise positive.

2.2.1. Primal matrix 21

This subsection presents the construction of the matrix that maps the primal volumes to the primal edges. This represents an exact calculation of the divergence of the quantity represented on the primal edges. This was done once manually for the N=3 grid in Figure 2.1, and on this basis, an algorithm was generated by dividing the matrix into sub-blocks that repeat.

The size of the matrix is first calculated. The number of columns correspond to the number of primal edges and the number of columns is the number of primal volumes. Note that the augmented grid is used here so the yellow points are counted along with the red points in Figure 2.1. Similarly, the outer augmented boundary edges are also counted.

- Number of primal edges = $2 \cdot N(N+1) + 4 \cdot N$

- Number of primal volumes = $N^2 + 4 \cdot N$

For the N=3 grid, the size of the matrix is $(21, 36)$. As is seen in the numbering of the edges, the first half of the edges correspond to the horizontal fluxes and the second half to the vertical fluxes. Therefore the matrix was divided into two halves of size $(21, 18)$ and then put together.

Horizontal Fluxes

The left half of the matrix is shown in Table 2.1 in the form of a table. The algorithm for its generation is as follows:

- The first and last N rows are full of zeros because the top and bottom augmented volumes do not have vertical edges associated to them (yellow boxes)
- The rest of the rows in the middle are formed by a repeating matrix of $N + 2$ rows and $N + 3$ columns with -1 on the lower main diagonal and +1 on the upper main diagonal. These represent all the horizontal fluxes present.
- These repeating matrices(green blocks) are placed in the diagonal of a larger matrix N times.

Table 2.1: Left hand side of the tE21 matrix.

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
-1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	-1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	-1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	-1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	-1	1	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	-1	1	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	-1	1	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	-1	1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	-1	1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	-1	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	-1	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	-1	1	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	-1	1	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-1	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-1	1
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Vertical Fluxes

The right half of the matrix for N=3 is shown in Table 2.2 in the form of a table. The explanation for its generation is as follows:

- The augmented volumes on the left and right of the grid do not have vertical flux associated with them. This is shown in the blue and green block which are all zeros.
- the $N + 1^{th}$ row and the $N + 1^{th}$ row from the last are the blue all-zero rows. The green blocks repeat after every N rows after the first blue block.
- The populated block is the yellow block of size $(N, 2 * N)$. The top left entry is -1 and there is a +1 entry N columns to the right. These correspond to the inward flux below and outward above the volume. This shifts one column right every row as *for the same y-coordinate* the counter of the volumes and edges increment together.
- Each yellow block corresponds to a row of primal volumes in the mesh. Each yellow block is shifted N columns right relative to the one above it.

Table 2.2: Right hand side of the tE21 matrix.

-1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	-1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	-1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	-1	0	0	1	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	-1	0	0	1	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	-1	0	0	1	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	-1	0	0	1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	-1	0	0	1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	-1	0	0	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	-1	0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	-1	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	-1	0	0	1	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	-1	0	0	1	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	-1	0	0	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	-1	0	0	1

Removal of boundary columns

For the flux boundary condition to be set, the columns of the tE21 matrix that correspond to the edges that were added as a result of augmenting the mesh are extracted into another matrix. This is then multiplied with the prescribed flux here (all zero in this case) and returned as Equation (2.13). For this, all the columns of the tE21 matrix are iterated over and the corresponding columns are removed.

The algorithm used to extract the columns is shown below:

- i = 1
- Remove column i
- For i=1:N-1
  - i = i + N + 2
  - Remove column i
  - i = i+1
  - Remove column i
- i = i + N + 2
- Remove column i
- i = i+1
- Remove columns i to i + N - 1
- Remove last N columns

This leaves the final tE21 matrix with

- Number of columns: $2 \cdot N(N+1)$
- Number of rows: $N^2 + 4N$

With this, the the flux on the boundary of the domain is included into the mass conservation of the NS equations in the form of $\vec{u}_{norm}$ (Equation (2.7)).

$$\vec{u}_{norm} = \begin{pmatrix} \tilde{e}_{1,i+1} & \dots & \tilde{e}_{1,j+1} \\ \vdots & & \vdots \\ \tilde{e}_{k,i+1} & \dots & \tilde{e}_{k,j+1} \\ \vdots & & \vdots \\ \tilde{e}_{K,i+1} & \dots & \tilde{e}_{K,j+1} \end{pmatrix} \begin{pmatrix} \tilde{u}_{i+1} \\ \vdots \\ \tilde{u}_{j+1} \end{pmatrix} \tag{2.13}$$

2.2.2. Dual matrix 10

The matrix $\mathbb{E}^{(1,0)}$ represents a mapping on the dual mesh from the points to the edges. As shown in Figure 2.1b, the dual points and the primal volumes share the same number (red text). Furthermore, the dual and primal edges show share the same numbering and are positive rightwards and upwards. The only difference being that the dual points are sinks and the primal volumes are source-like.

The corresponding numbering means that for any primal edges (say *a, b, c* and *d*) that border a given primal volume *V*, it is guaranteed that on the dual mesh edges *a, b, c* and *d* will be connected to point *P* that is in the same position as the center of primal volume *V*. This fact along with the fact that the primal volumes and dual points have opposing orientations, means that the directional physical mapping between the dual points & edges and the primal edges&volumes is identical but with opposite directionality.

In the matrix $\mathbb{E}^{(1,0)}$, each row represents a dual edge and each column a dual point. In the matrix $\tilde{\mathbb{E}}^{(2,1)}$, each row represents a primal volume and each column a primal edge. This being said, along with the now known cognateness of primal volumes to the dual volumes and the primal and dual edges, it is said that the two incidence matrices are transposes of each other with opposing signs due to the opposing orientation of the dual points and the primal volumes.

$$\mathbb{E}^{(1,0)} = -(\tilde{\mathbb{E}}^{(2,1)})^T \tag{2.14}$$

2.2.3. Primal Matrix 10

The matrix $\tilde{\mathbb{E}}^{(1,0)}$ represents a mapping on the primal mesh from the points to the edges. If the flux of an edge that is connected to a point has a clockwise flux, it is given a +1 value, -1 if anti-clockwise and 0 if the edge is not connected to the point. For the generation of this matrix, the augmented mesh was not considered, but just the original core mesh.

As a result:

- Number of rows = $2 \cdot N(N+1)$
- Number of columns = $(N+1)^2$

For N = 3, the size of the matrix becomes $(24, 16)$. The matrix, constructed manually for N = 3 is shown in Table 2.3 in a tabular format. It is vertically divided into two halves and constructed. The method used to construct this matrix for arbitrary N is as follows:

- The top half of the matrix corresponds to the primal edges that have horizontal flux (these edges themselves are vertical).
- This top matrix is of size $(N(N+1), (N+1)^2$, it simply contains two diagonals, the main diagonal with all -1 entries and the $(N+1)^{th}$ diagonal with +1 entries. This gap of $N+1$ is because each edge is connected to two points that are one row apart in the grid.
- The green blocks correspond to the edges with vertical flux. These edges connect points that are adjacently numbered due to the numbering system used. Each green block represents one of the rows of points in the grid. Therefore there are $N+1$ green blocks. They are each of size $(N, N+1)$.

Table 2.3: The matrix tE10 for N = 3.

-1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
0	-1	0	0	0	1	0	0	0	0	0	0	0	0	0	0
0	0	-1	0	0	0	1	0	0	0	0	0	0	0	0	0
0	0	0	-1	0	0	0	1	0	0	0	0	0	0	0	0
0	0	0	0	-1	0	0	0	1	0	0	0	0	0	0	0
0	0	0	0	0	-1	0	0	0	1	0	0	0	0	0	0
0	0	0	0	0	0	-1	0	0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	-1	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	-1	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0	0	-1	0	0	0	1	0	0
0	0	0	0	0	0	0	0	0	0	-1	0	0	0	1	0
0	0	0	0	0	0	0	0	0	0	0	-1	0	0	0	1
1	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	1	-1	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	-1	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	-1	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	-1	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	1	-1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1	-1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1	-1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	1	-1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1	-1	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	1	-1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	-1

2.2.4. Dual matrix 21

As presented in Section 2.2.2, the correspondence in features of the dual and primal meshes allowed the conclusion that the matrix on the dual mesh can be derived from the matrix on the primal mesh. The matrix $\mathbb{E}^{(2,1)}$ maps the dual edges onto the dual volumes. The dual edges corresponds directly to the primal edges, and the dual volumes to the primal points. Matrix $\tilde{\mathbb{E}}^{(1,0)}$ maps the primal points onto the primal edges. Therefore, analogous to the justification presented in Section 2.2.2, the matrix $\mathbb{E}^{(2,1)}$ is the transpose of the matrix $\tilde{\mathbb{E}}^{(1,0)}$. Here there is no minus sign as the dual volumes and primal points have the same clockwise orientation.

$$\mathbb{E}^{(2,1)} = (\tilde{\mathbb{E}}^{(1,0)})^T \tag{2.15}$$

2.3. Hodge Matrices

While the incidence matrices map the geometry of the mesh between the dimensions *on the same mesh* (either the primal or dual mesh), the Hodge matrices bridge the dual and primal meshes. This Hodge matrices used in this assignment are shown in Figure 2.2: namely: $\mathbb{H}^{(\tilde{1}1)}$, $\mathbb{H}^{(1\tilde{1})}$, $\mathbb{H}^{(\tilde{1}\tilde{1})}$. The use in the solver of each of these is presented specifically in the equations in Section 2.1.

The Hodge matrices encode the approximation in the system. Typically, the constitutive relations involved in a model are represented here while the exact operations are represented by the incidence matrices.

In this assignment, the Hodge matrices are used for these specific reasons:

- $\mathbb{H}^{(\tilde{1}1)}$ and $\mathbb{H}^{(1\tilde{1})}$ to approximate the velocities on the dual dual from the flux on the primal edges and vice versa respectively.
- $\mathbb{H}^{(\tilde{0}2)}$ to approximate quantities on the primal points from the dual volumes.

Hodge Matrices 11
These are both square matrices as they express the quantities on one of the edge sets (either primal or dual) as a function of the quantity on each of the other edge sets. In this specific case, it is a diagonal matrix as the approximation used makes the dual edge quantity a funciton of only the corresponding primal edge and vice versa.

For $\mathbb{H}^{(\tilde{1}1)}$ the quantity is simply divided by the edge length in the dual mesh and multiplied with those of the primal mesh. This is presented in Equation (2.16). The converse is true for the matrix $\mathbb{H}^{(1\tilde{1})}$ as shown in Equation (2.17).

$$\mathbb{H}^{(\tilde{1}1)}_{i,i} = \frac{\tilde{h}_i}{h_i} \tag{2.16}$$

$$\mathbb{H}^{(1\tilde{1})}_{i,i} = \frac{h_i}{\tilde{h}_i} \tag{2.17}$$

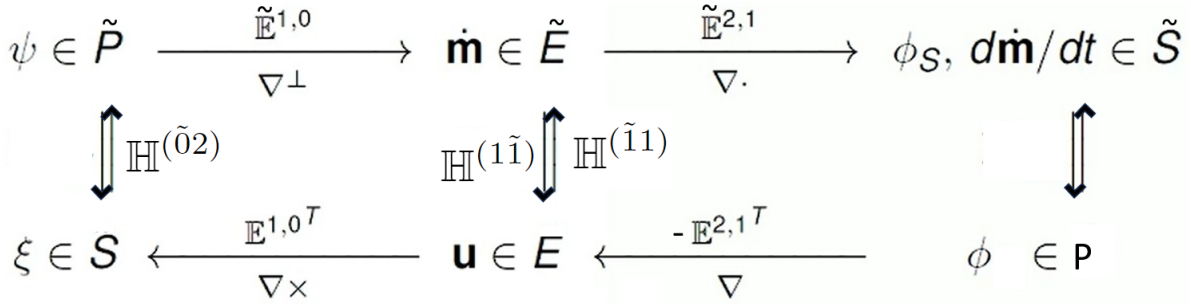


Figure 2.2: The double de Rham sequence for this 2D mesh system. Only the Hodge matrices used in the assignment are shown.

2.4. Tangential Velocity Boundary Condition

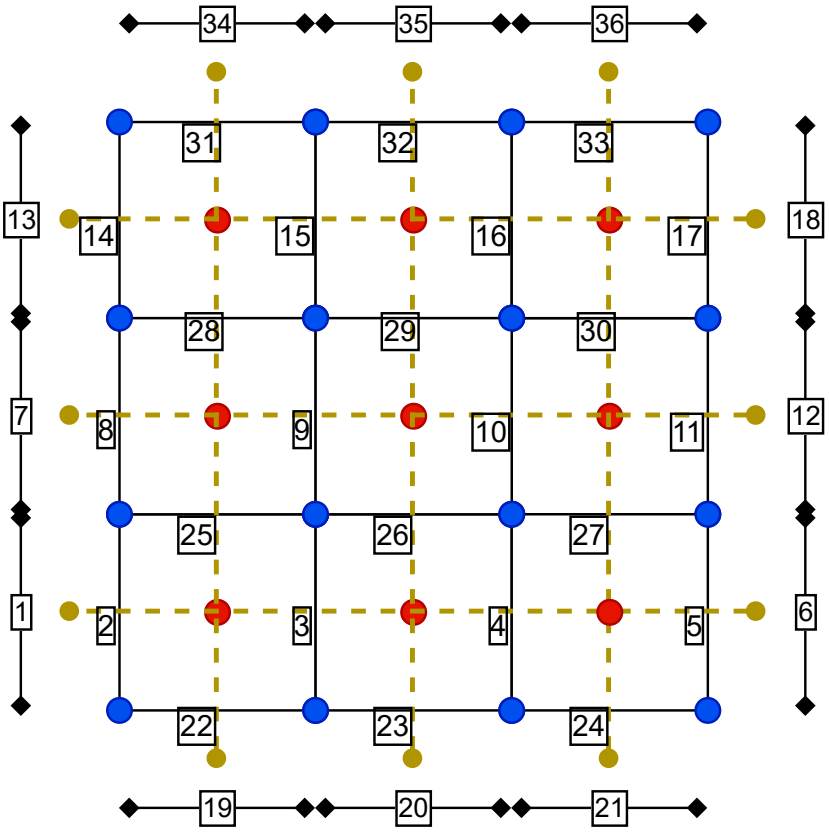
At the boundary, along with the flux (which was treated in Section 2.2.1), the tangential velocity is also prescribed. This must be handled on the dual edges, since that is the only location where the tangential velocity can be defined. In the solver itself, this prescribed tangential velocity is used in the form of the prescribed vorticity as seen in Equation (2.9) as ξ_{prescr}.

In order to calculate ξ_{prescr}, an incidence matrix is needed, mapping only the velocities on the augmented edges to the vorticity induced from them on the dual volumes. Again, for N=3, the corresponding matrix is presented in Table 2.4.

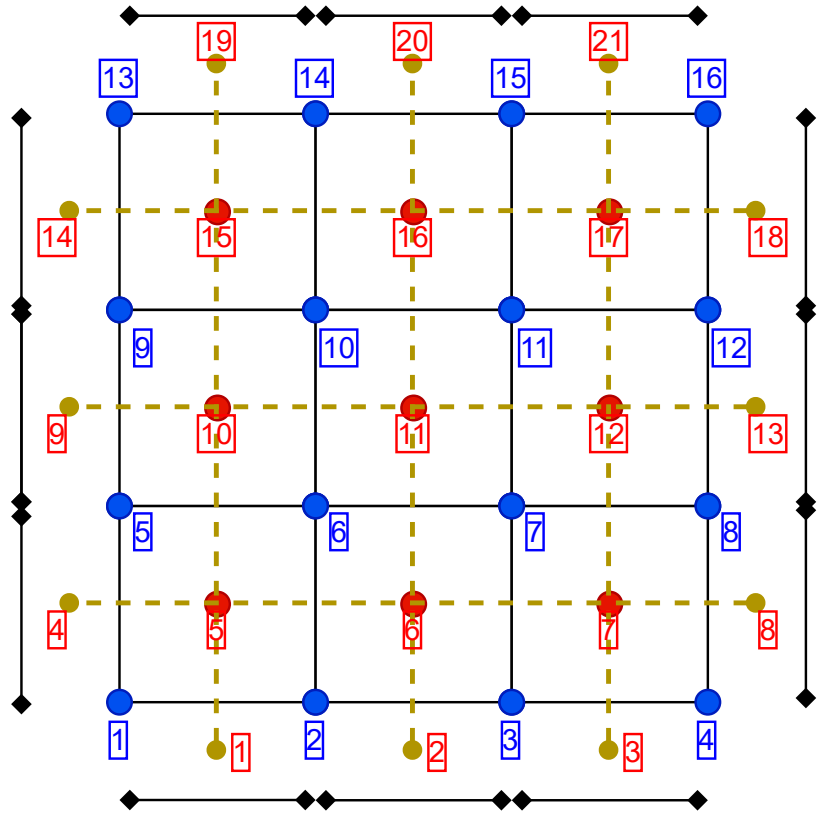
Each row in this matrix represents one of the dual volumes. The columns represent the *boundary* edges (dotted red edges) in the augmented dual mesh shown in Figure 2.3. The internal edges are *not* included. As is seen in Figure 2.3, there are only 4 internal dual volumes that are not bounded by the boundary edges (for N=3), and therefore, there are 4 rows in the matrix that are all zeros as the prescribed tangential velocity at the boundary has no effect on the vorticity in these volumes. This matrix is only used to translate the tangential velocity boundary condition onto the calculated vorticity field.

Table 2.4: Matrix that maps boundary tangential velocity onto vorticity defined on dual volumes.

1	0	0	0	0	0	0	0	-1	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	-1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	-1	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
0	0	0	0	-1	0	0	0	0	0	0	0	0	0	-1	0
0	0	0	0	0	-1	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	-1	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	-1	0	0	0	0	0	0	0	1



(a) Here the numbering of the dual and primal edges are shown in black numbers.



(b) Here the numbering of the primal points (dual volumes) is shown in blue text and the numbering of the primal volumes (dual points) is shown in red text.

Figure 2.1: The grid for N=3 shown (with the augmented volumes). The blue dots represent the primal points. The black lines represent the primal edges. The yellow dotted lines represent the dual edges. The red and yellow dots represent the dual points.

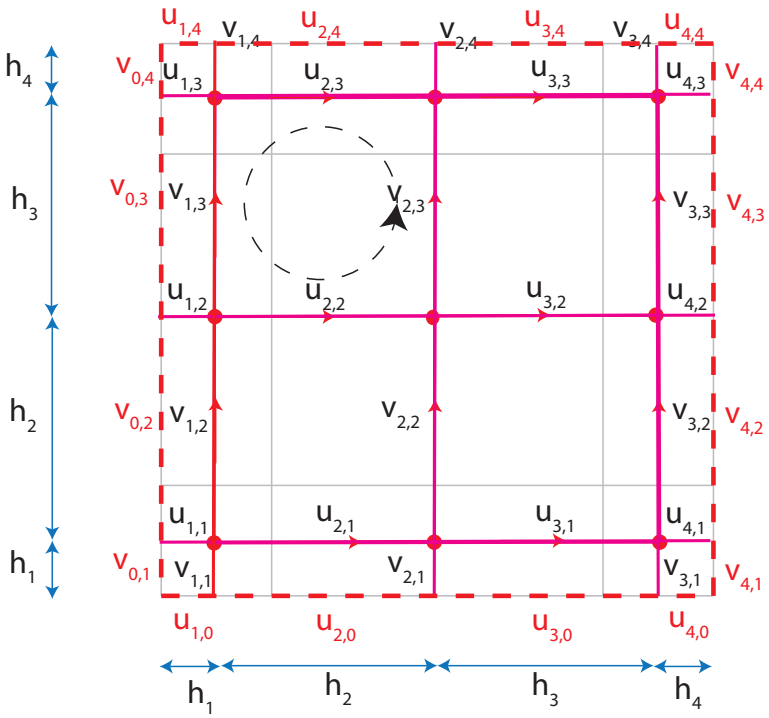


Figure 2.3: The augmented dual mesh used to implement the tangential velocity boundary condition.

3

Results

This chapter presents the visual results as requested in the assignment description. Results include the streamfunction, pressure and vorticity field inside the entire 1x1 flow domain as well as multiple variables plotted on the horizontal and vertical centerlines of the square domain. For comparison purposes, the flowfield illustrations use the same isolines as the ones presented in the reference literature. Likewise, for the centerlines reference values are plotted also in order to visualize the accuracy and convergence of the field as the discretization N increases. Note that for all quantities, the field is resolved and presented for the numbers N of 15, 31, 47, 55, and 63.

3.1. Streamfunction

The stream function is a mathematical function that describes the flow pattern in a two-dimensional, incompressible fluid flow. It is a scalar function defined such that its partial derivatives yield the velocity components of the flow.

In two dimensions, the velocity components can be expressed as follows:

$$u = \frac{\partial \psi}{\partial y} \tag{3.1}$$

$$v = -\frac{\partial \psi}{\partial x} \tag{3.2}$$

However, this system of equation can easily be interpreted as a curl operation (refer to the primal mesh side of the double De Rham sequence going from streamfunction to velocity) on the velocity vector u, as defined in the solver including the horizontal and vertical velocities on the dual mesh. It is then apparent that the incidence matrix that corresponds to the curl operation on the primal mesh would be sufficient to move from streamfunction ψ to velocity on the primal mesh as follows:

$$u = \tilde{\mathbb{E}}^{(1,0)}\psi \tag{3.3}$$

Given this relation, in order to solve for the streamfunction one needs to transfer the velocity to the primal mesh (1) and then let the in-built functions in matlab solve the linear system (2):

1) $u_{prim} = Ht11 * u$;
2) $psi = linsolve(full(tE10), u_{prim})$;

Note that since so far we have been working with single vectors containing the values of the 2D flow field, mesh staggering is done through reshaping the vector and the appropriate rotation/flipping brings it to the conventional form of a cavity flow.

The results are shown below, with a clear convergence to a smooth solution visible with increasing N.

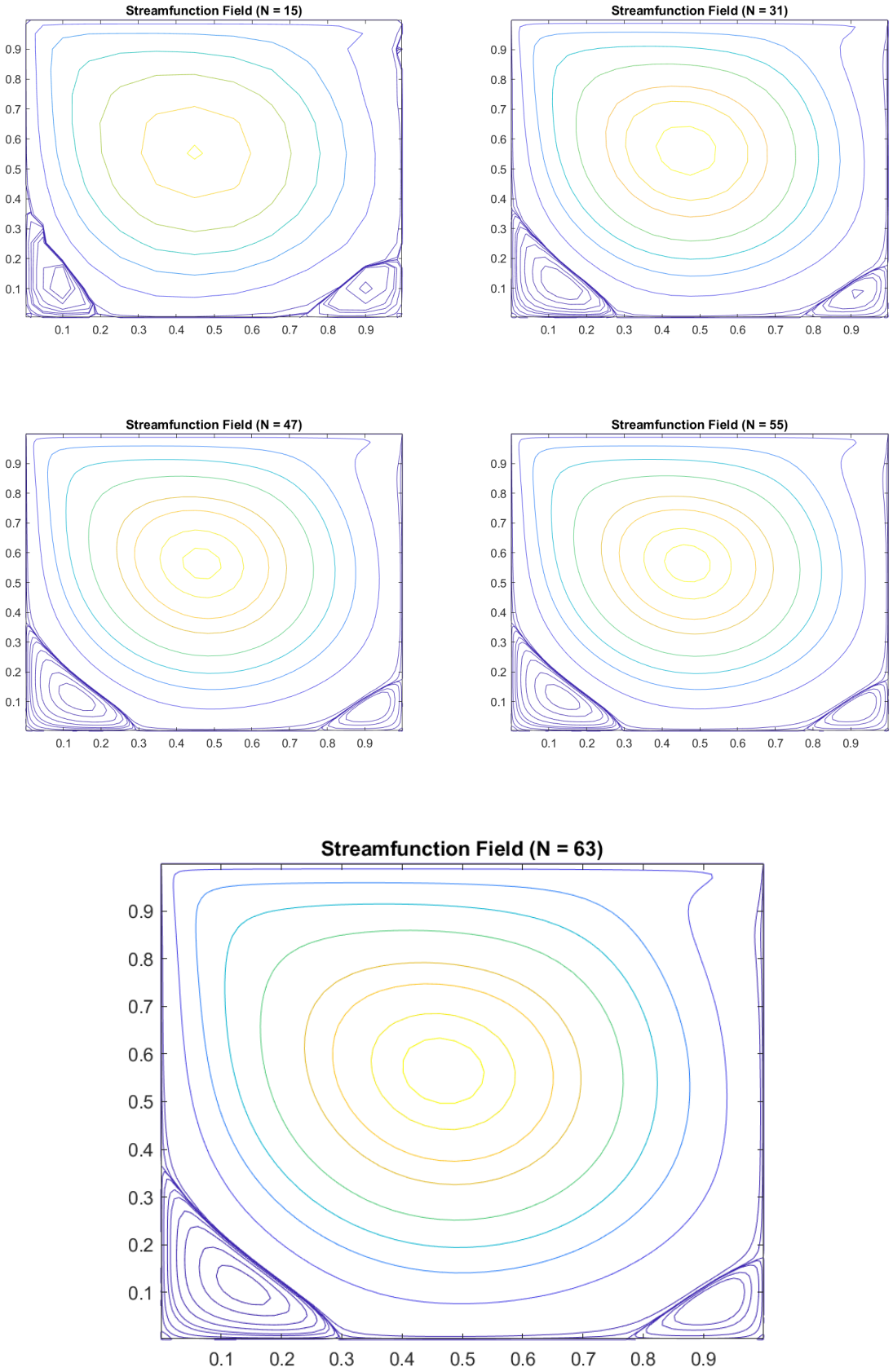


Figure 3.1: Streamfunction field for the different Ns.

3.2. Vorticity

The vorticity is taken directly from the solver, reshaped, rotated to fit cavity flow standards and presented below.

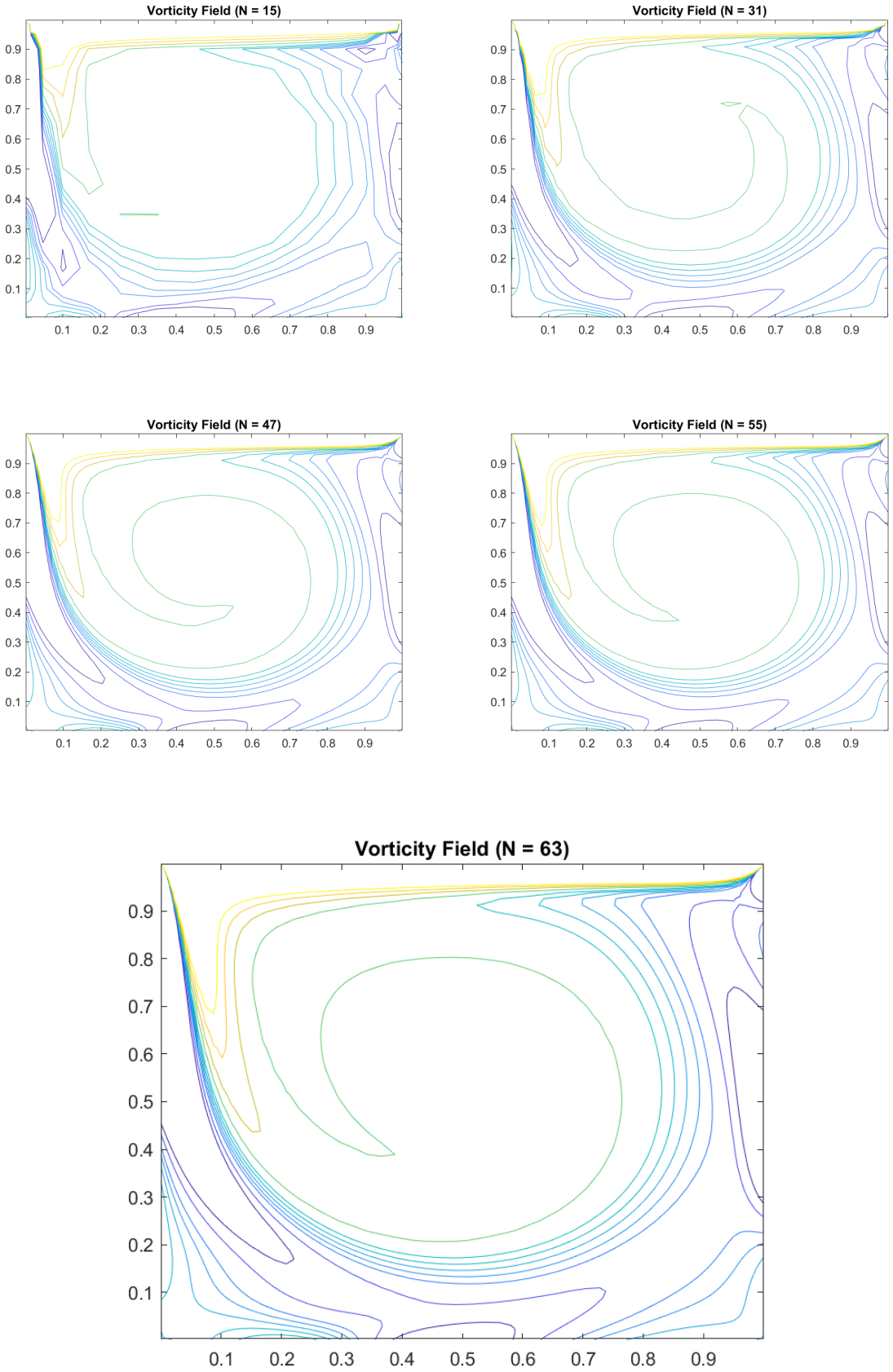


Figure 3.2: Vorticity field for the different Ns.

3.3. Pressure

For the calculation and the visualization of the pressure, the following steps are taken:

- Obtain pressure vector p on the dual mesh through the solver
- Remove boundary points
- Divide velocity vector u by the corresponding lengths to obtain the actual velocity
- Linearly interpolate velocity (horizontal and vertical) on the dual points
- Subtract the dynamic pressure $\frac{1}{2}(u^2_{horizontal} + u^2_{vertical})$ given the interpolated velocities from before
- Subtract the value of the pressure in the middle of the flow from the entire pressure vector so as to have a normalized zero pressure point in the middle (explanation for this degree of freedom given in the next chapter)
- Reshape and rotate/flip

The results again are shown below for the various N.

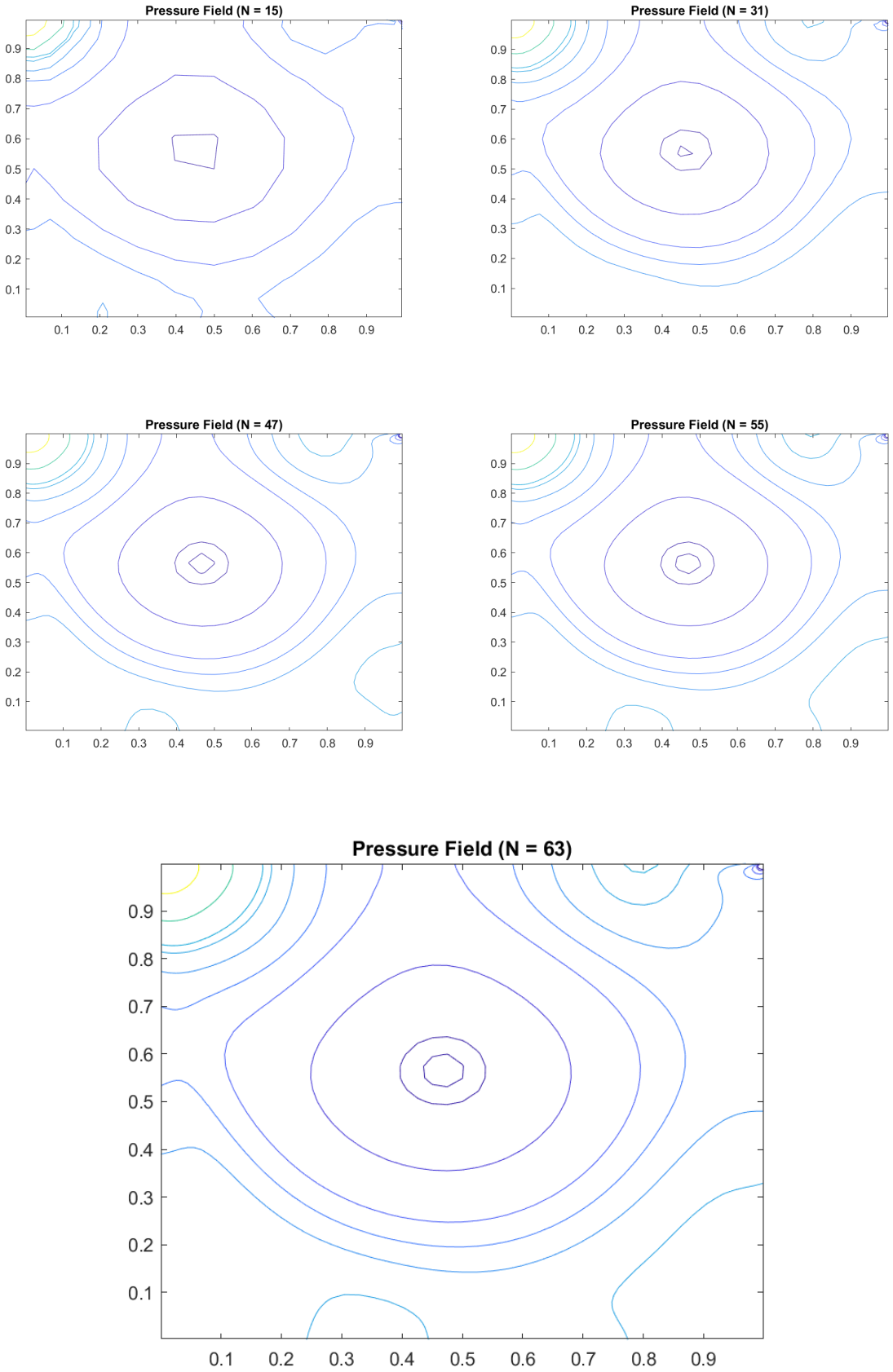


Figure 3.3: Pressure field for the different Ns.

3.4. Flow center-lines

In this section all the horizontal and vertical flow centerline comparisons with the reference values (red points) are illustrated. In all cases the convergence for higher N's is visible although in some cases less than others (for instance in the sudden vorticity drop on the left of the horizontal centerline). Note that there are no reference values for the vertical velocity on the vertical centerline and for the horizontal velocity on the horizontal centerline.

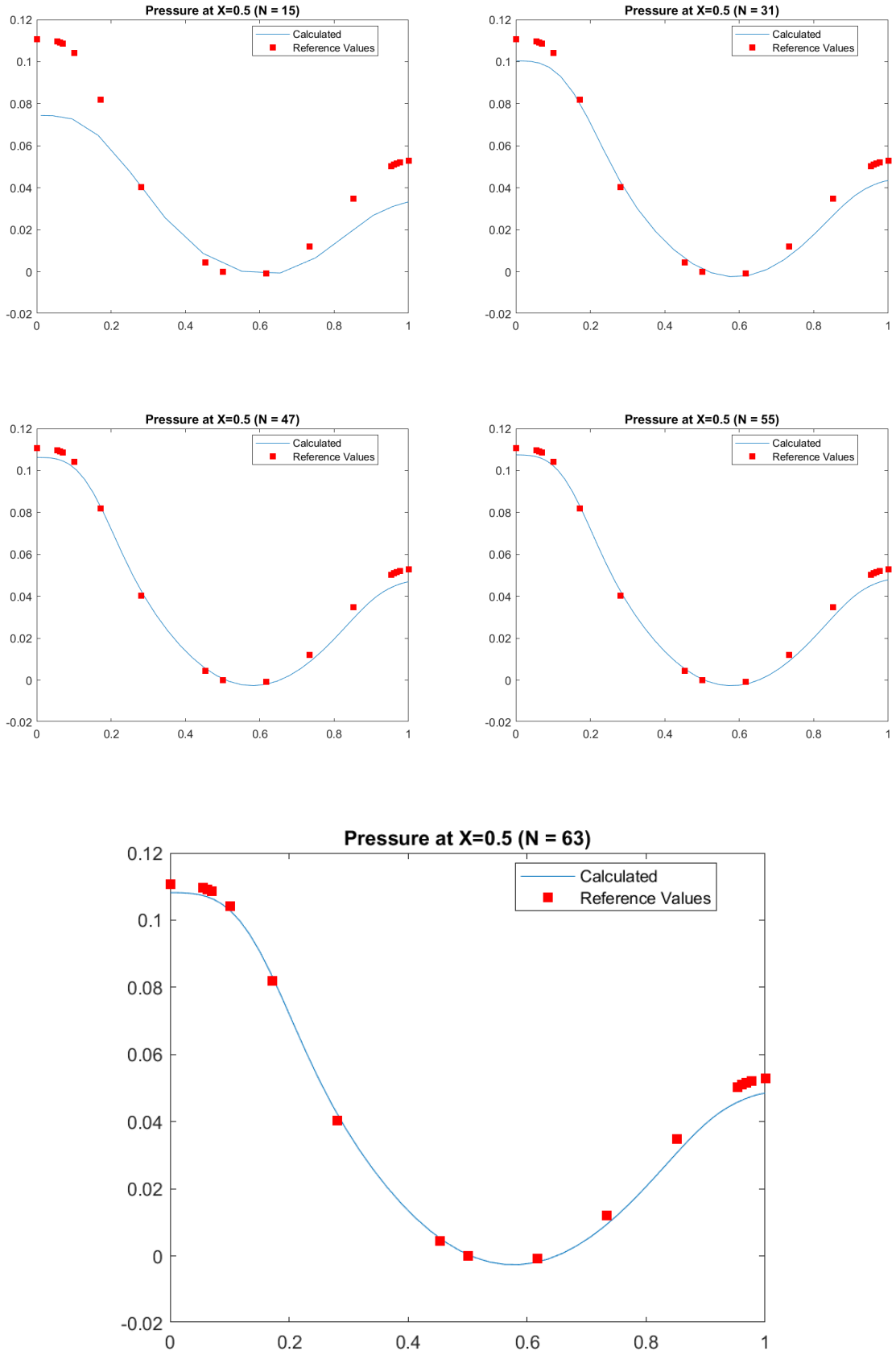


Figure 3.4: Pressure at vertical flow centerline.

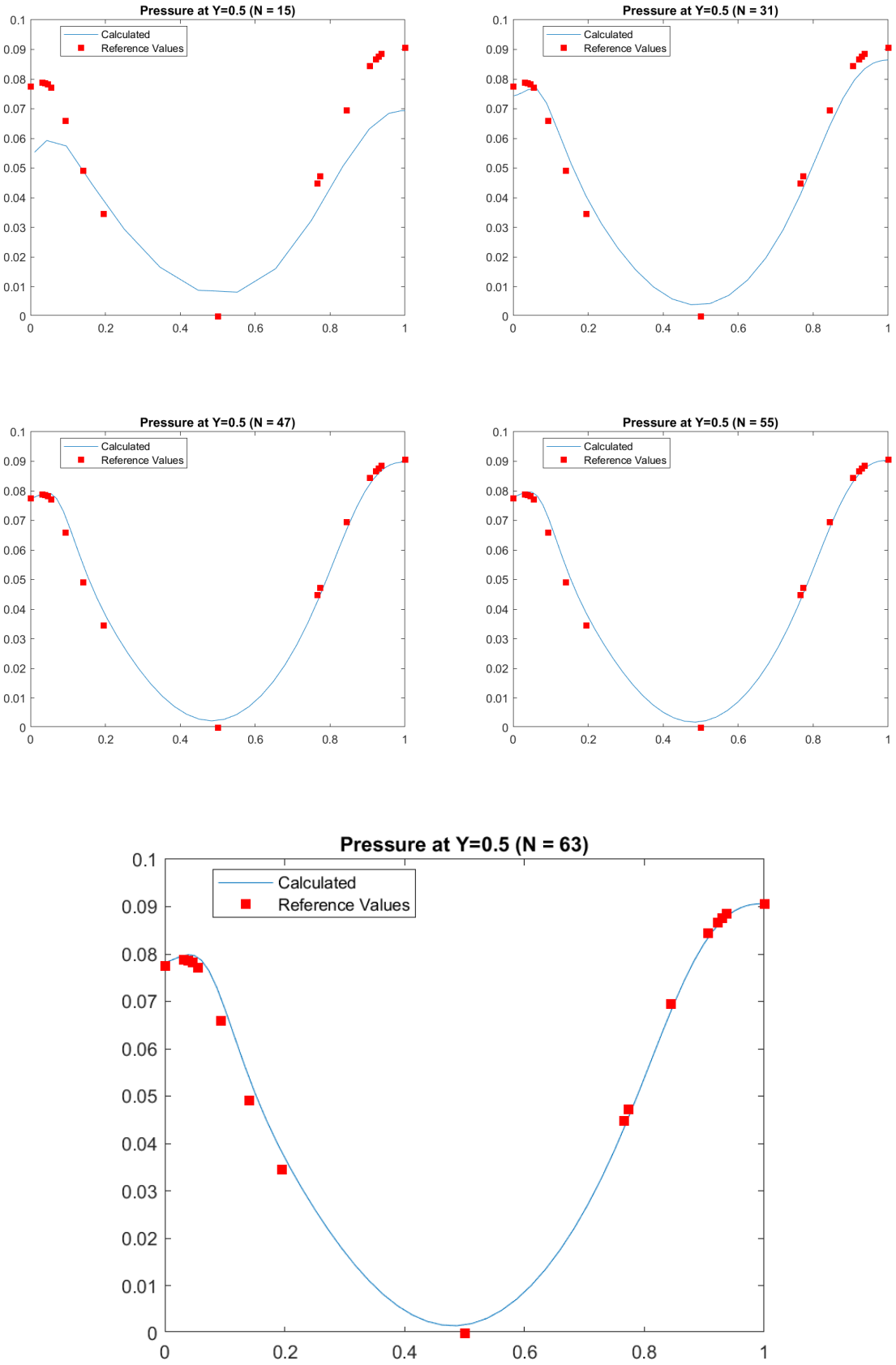


Figure 3.5: Pressure at horizontal flow centerline.

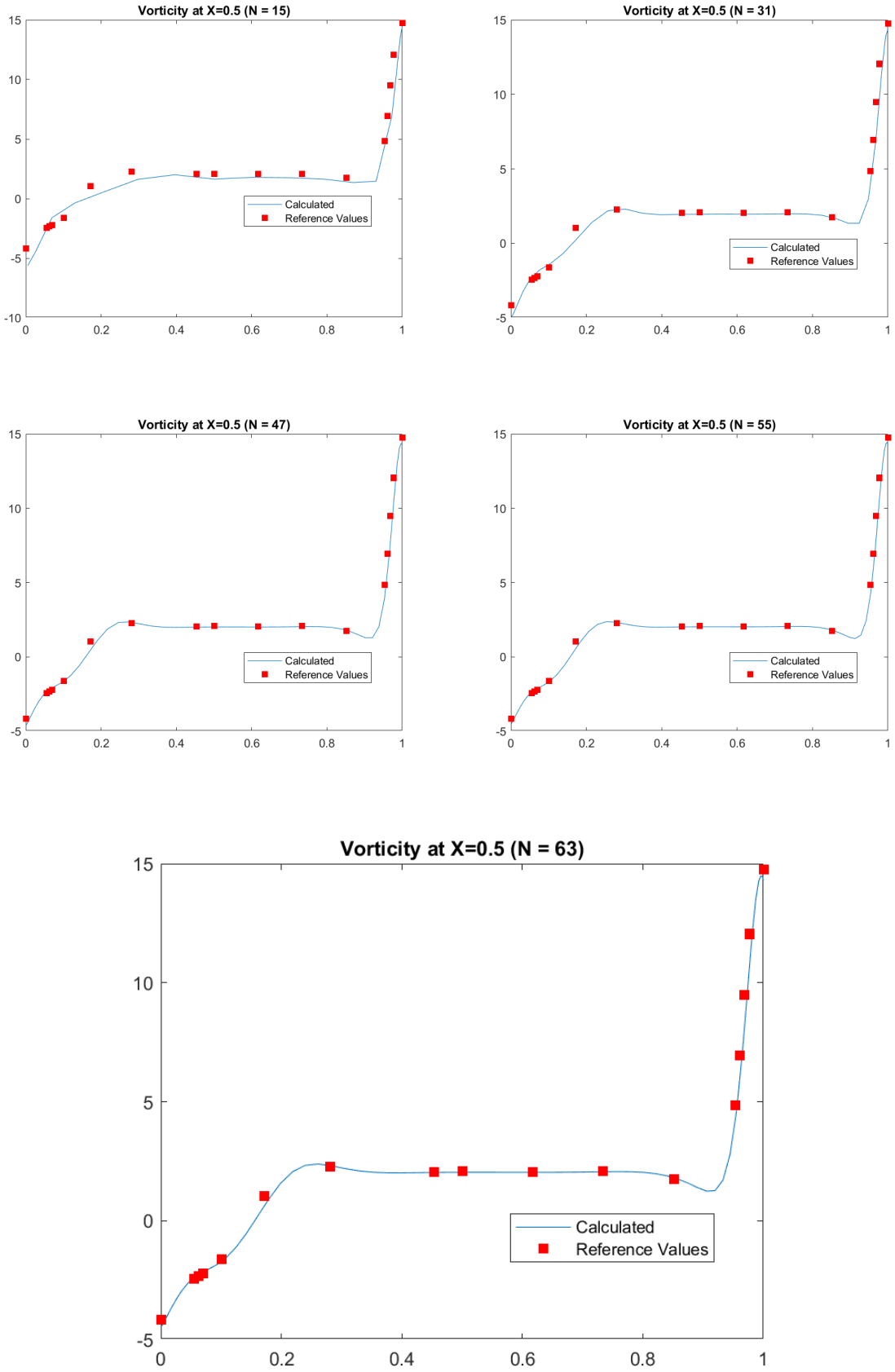


Figure 3.6: Vorticity at vertical flow centerline.

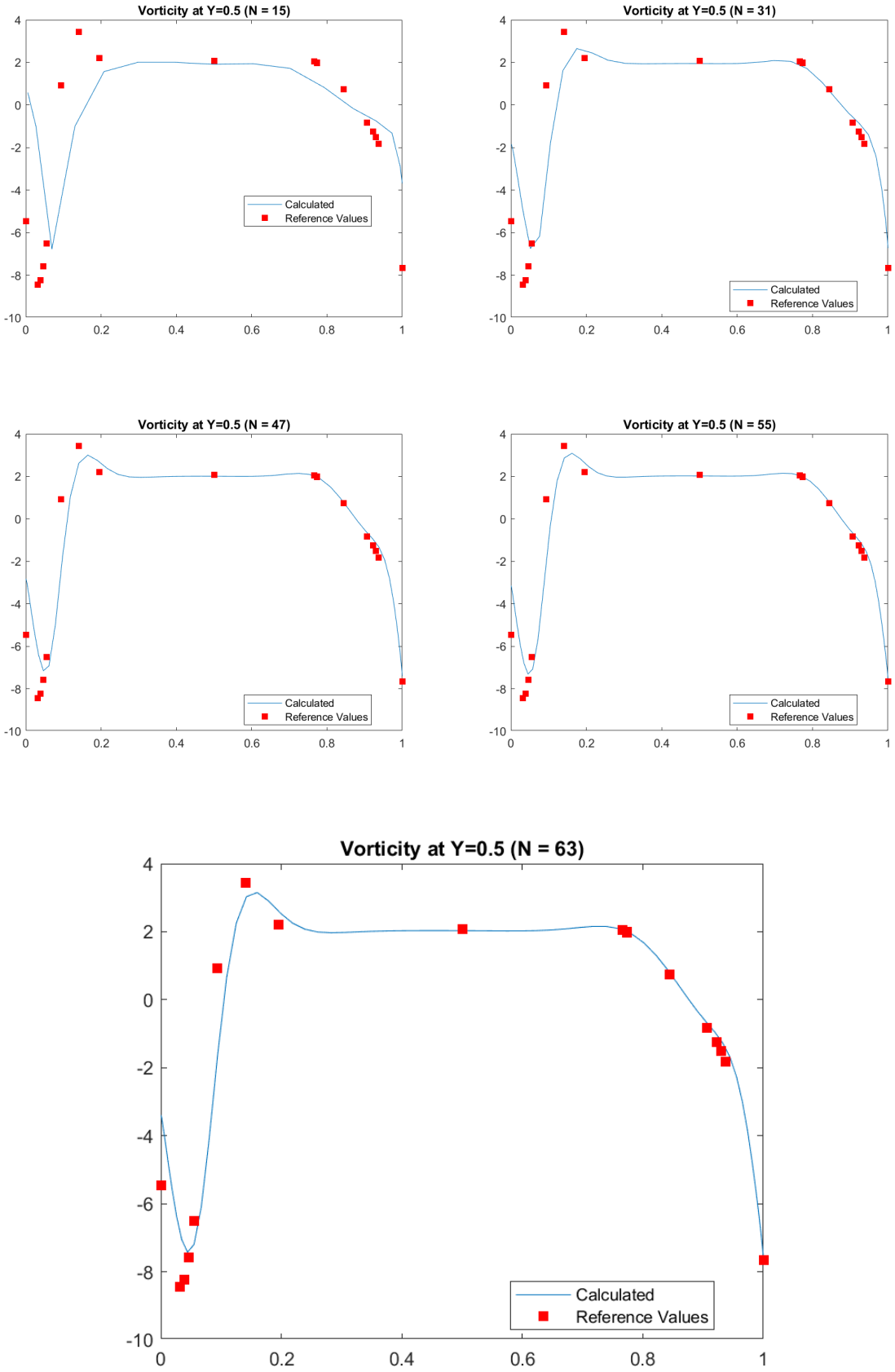


Figure 3.7: Vorticity at horizontal flow centerline.

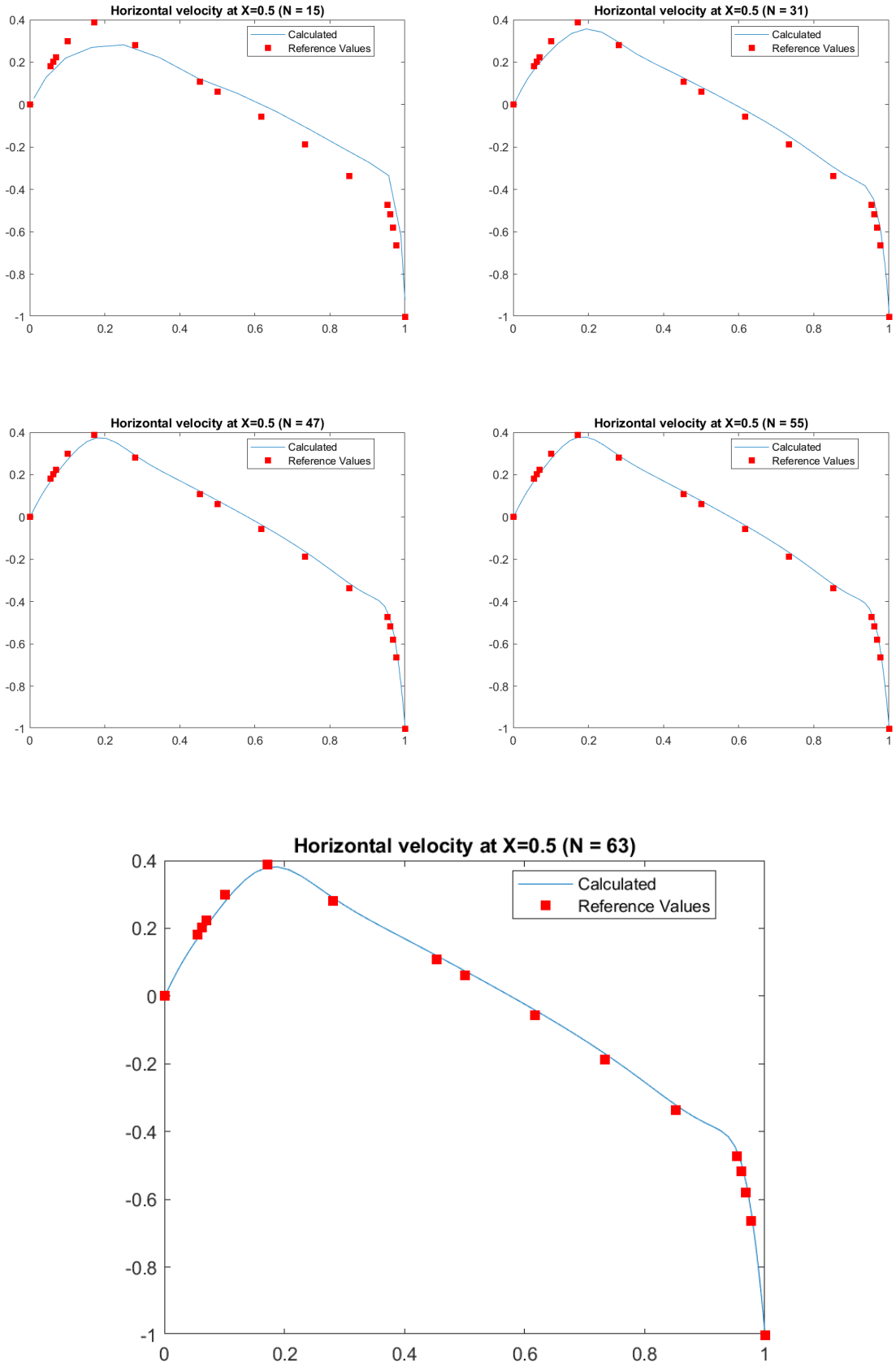
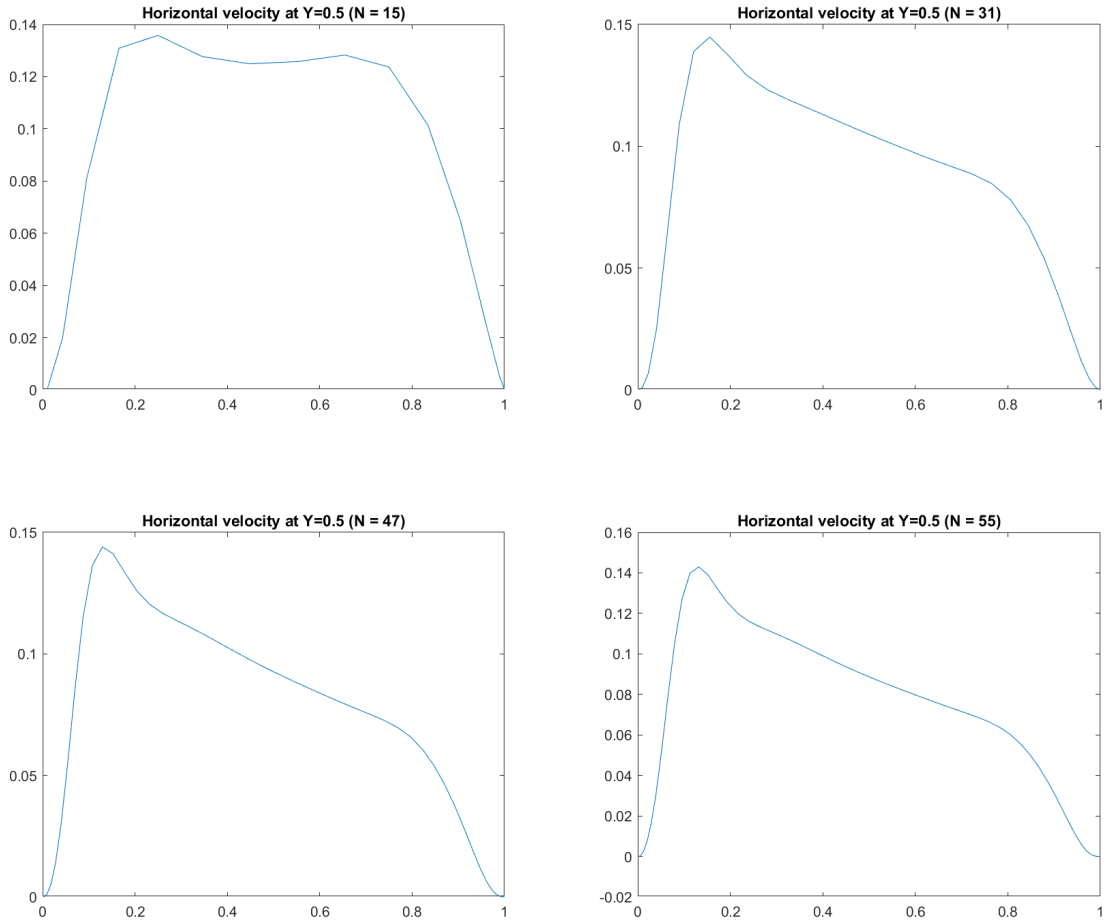


Figure 3.8: Horizontal velocity at vertical flow centerline.



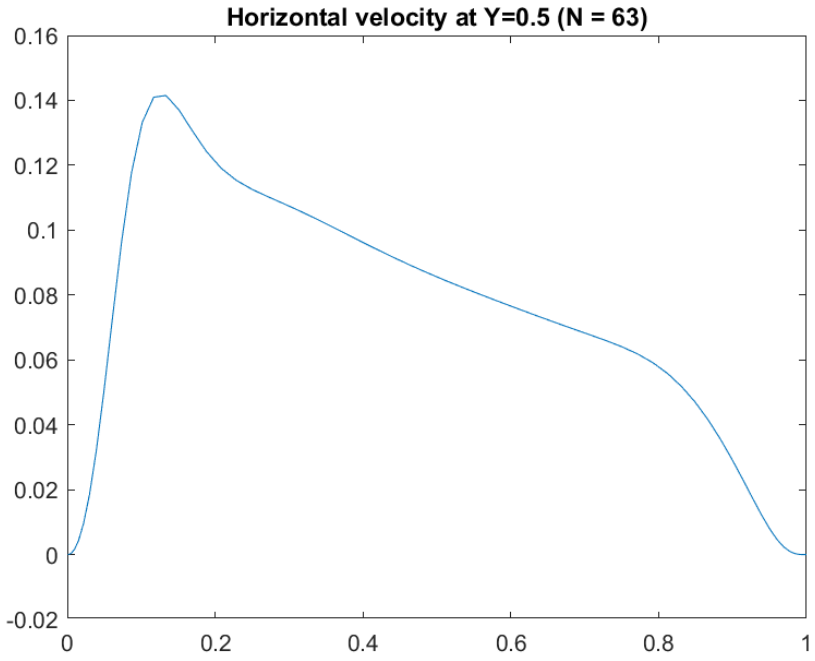


Figure 3.9: Horizontal velocity at horizontal flow centerline.

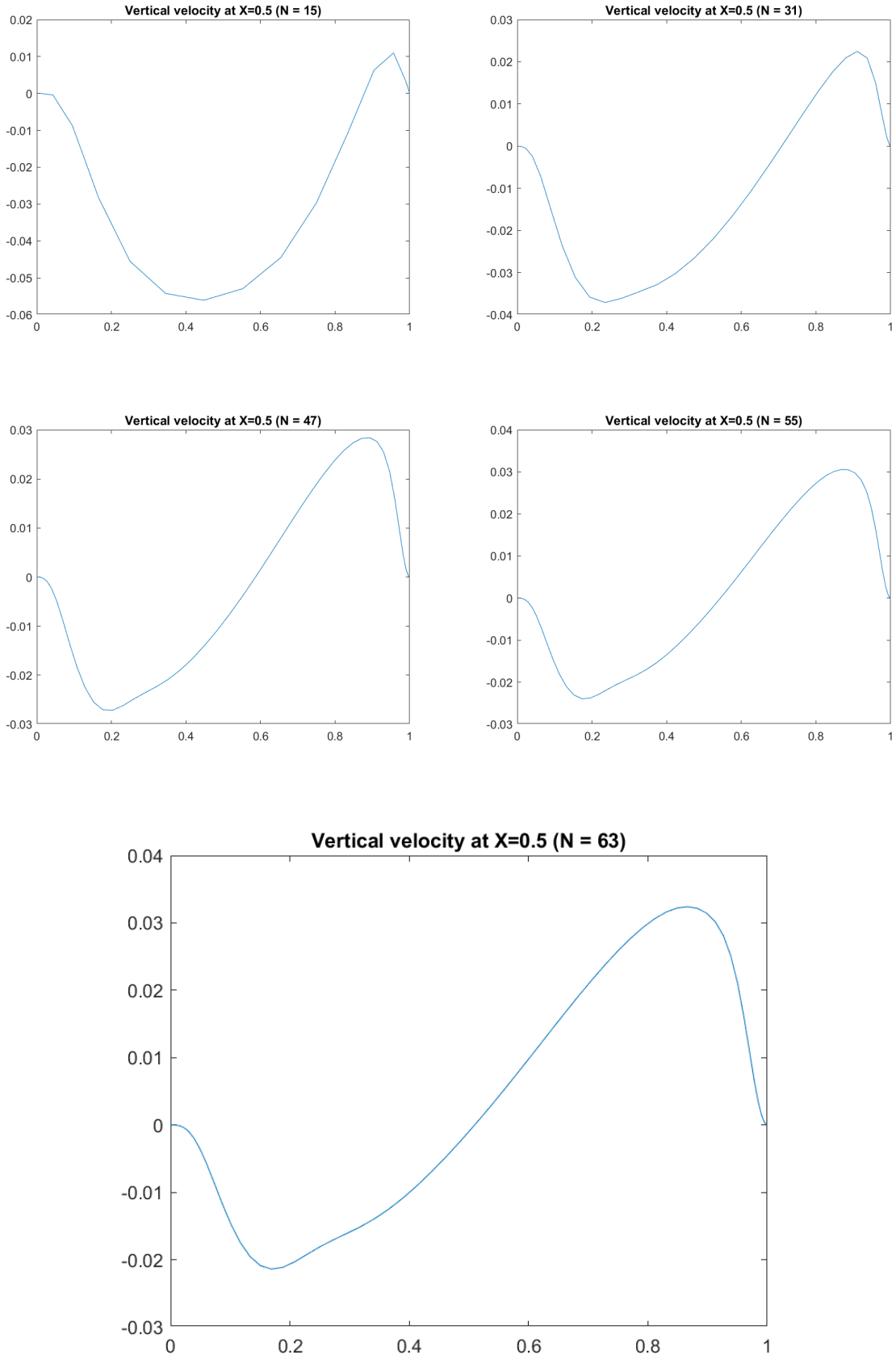


Figure 3.10: Vertical velocity at vertical flow centerline.

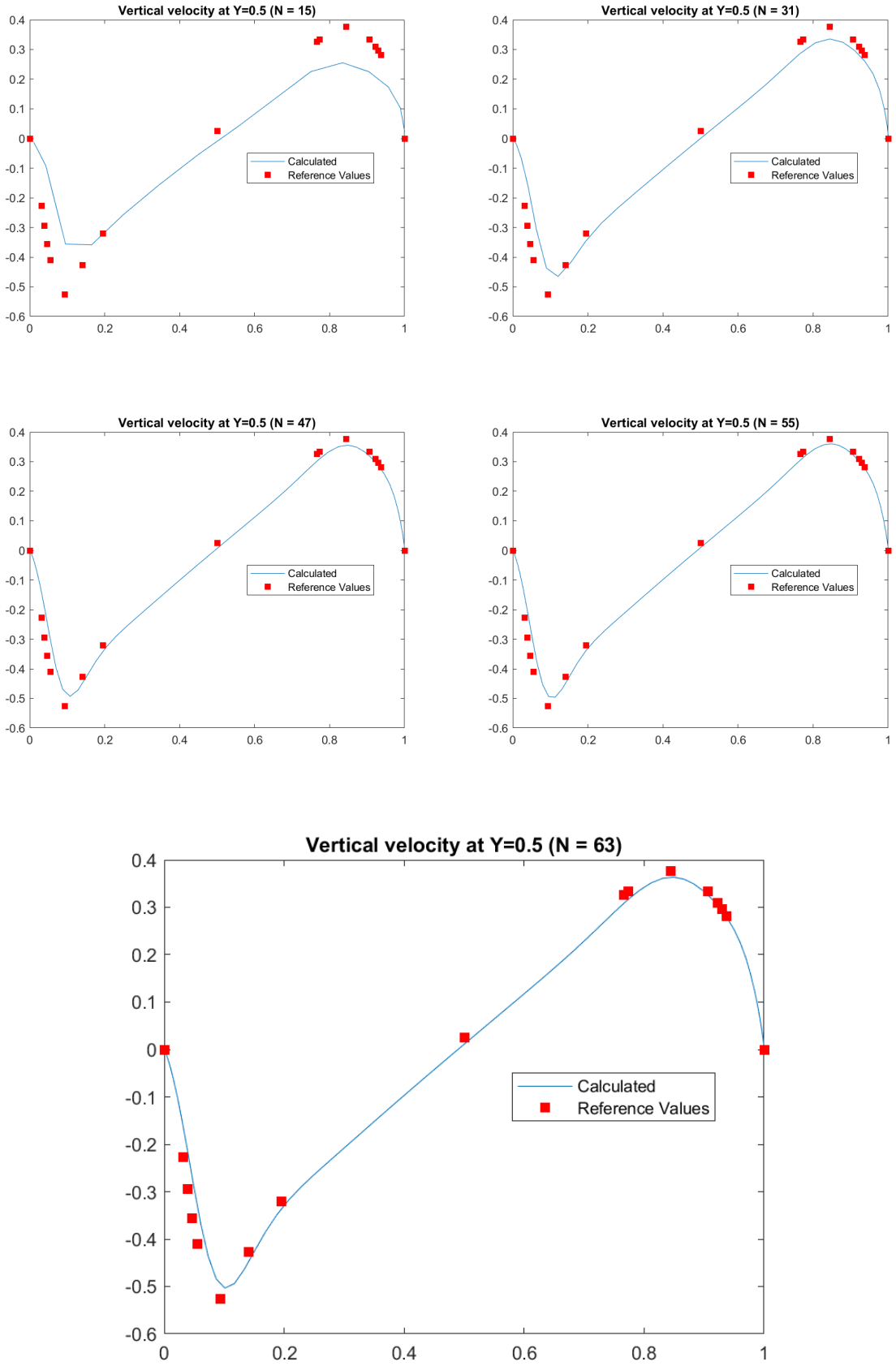


Figure 3.11: Vertical velocity at horizontal flow centerline.

4

Discussions

4.1. Timestep analysis

The default timestep as suggested in the assignment description is shown below:

$$\Delta t = \min\left(h_{\min}, \frac{1}{2}\,\mathrm{Re}\,h_{\min}^2\right) \tag{4.1}$$

In order, however, to optimize the timestep used, the concept of the Courant time step is investigated.

The Courant time step (CFL) condition, is a criterion used in numerical simulations to ensure stability and accuracy in solving time-dependent equations, particularly in the context of finite difference, finite volume, and finite element methods. It is derived from the concept of the wave propagation speed within a numerical domain and is based on the idea that the time step should be small enough to capture the fastest wave or signal propagating through the system without causing numerical instability.

The Courant time step is defined as the ratio of the spatial grid size (Δx) to the velocity of the wave (u), multiplied by a stability factor (CFL):

$$dt = CFL * \Delta x/u \tag{4.2}$$

In order to capture the field accurately, the CFL has to take values from 0 to 1. A smaller CFL number leads to a smaller time step, providing more accurate results but increasing the computational cost. Conversely, a larger CFL number can lead to instability or inaccurate solutions. In practice, to determine an appropriate CFL number testing is required and adjusting it until a stable and accurate solution is obtained.

In this assignment this number is tested with varying CFL and the convergence of the solver is monitored. Note that the dt changes with each iteration taking into account the smallest length scale (constant based on the initial h) and the highest speed spotted in the flow at the previous iteration (max value from vector u). The results are shown below tested for N=35 and using a tolerance value of 10^{-4}:

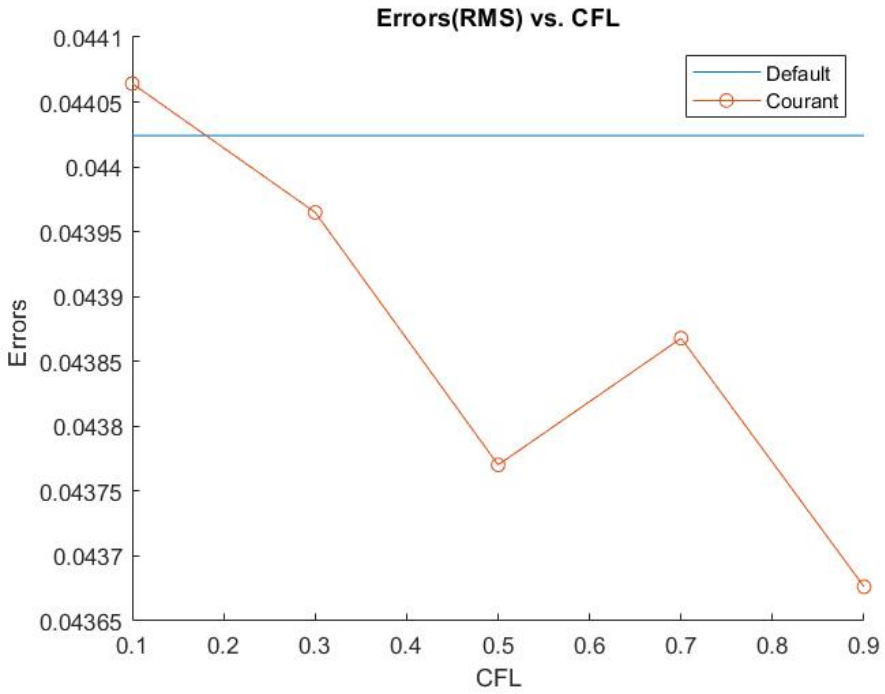


Figure 4.1: Error for various CFLs compared to default timestep (N=35).

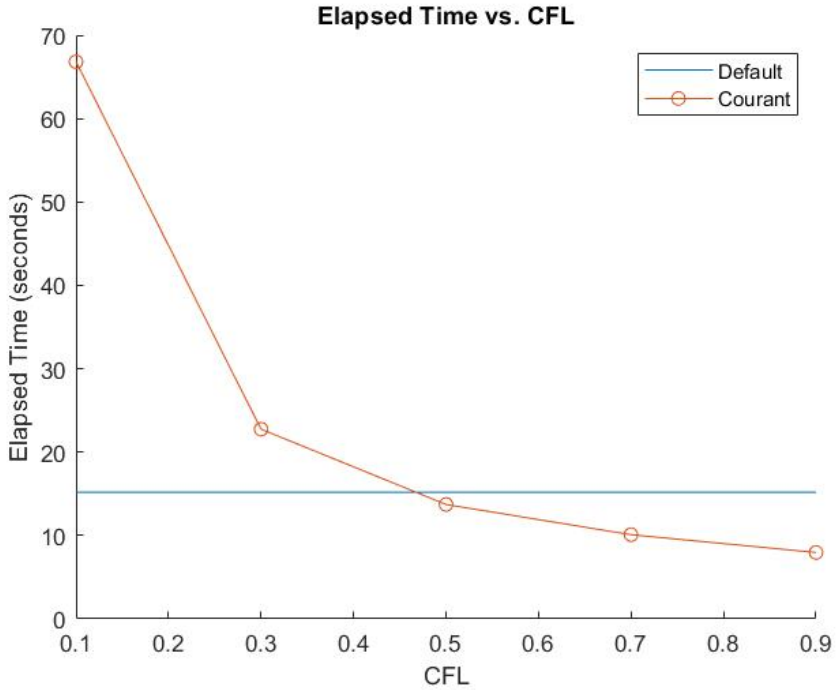


Figure 4.2: Runtime for various CFLs compared to default timestep (N=35).

Note that the error is calculated as the RMS of the velocity values at the horizontal centerline compared to the values found in literature for very high N's. Surprisingly, for lower CFL numbers the error is seen to have a slight increase which is counter-intuitive. In order to validate that behaviour, the same graphs are shown below but now for N=45:

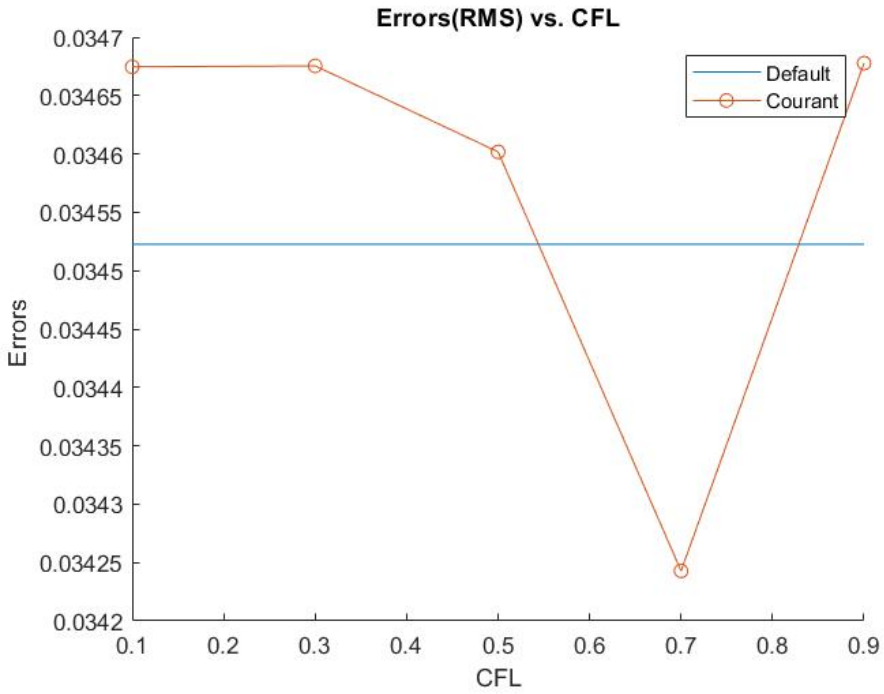


Figure 4.3: Error for various CFLs compared to default timestep (N=45).

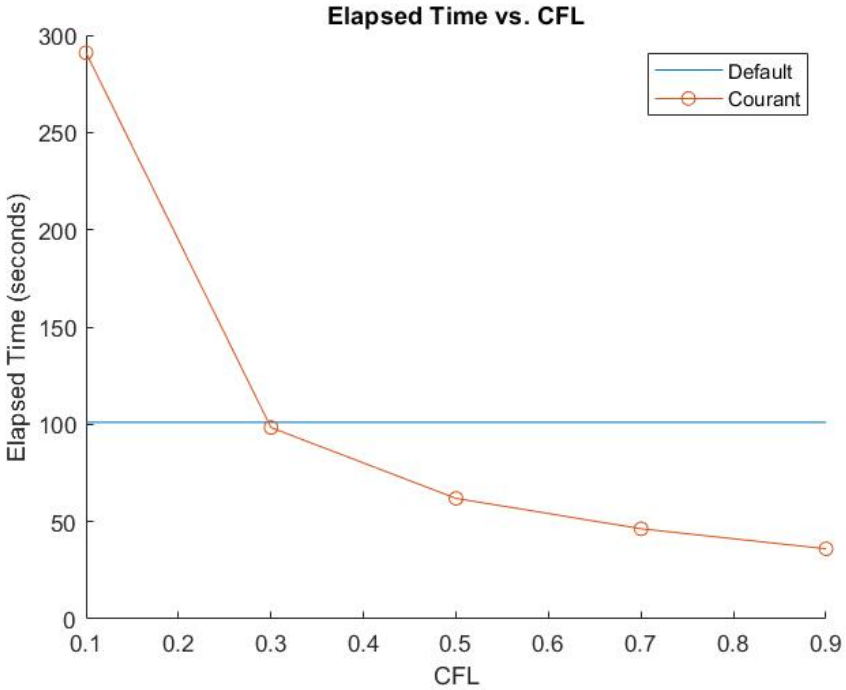


Figure 4.4: Runtime for various CFLs compared to default timestep (N=45).

For N=45, although the error behavior is inconsistent, one can see a maximum difference in error between the default and the Courant timestep of less than 1% which indicates no significant loss or gain of convergence accuracy. This leads to the conclusion that there is benefit from using the Courant number in terms of runtime, given that there is no significant decrease in error associated with it.

With that in mind, it was decided to try multiplying the default, constant timestep by certain integer

factors K and see the behaviour of the runtime as well as the error. The goal would be to identify at what factor K would the error start deviating significantly (more than 2%) from the error at the initial timestep. The formula for the timestep inside the solver would then be:

$$dt = K * \min\left(h_{\min}, \frac{1}{2}\,\mathrm{Re}\,h_{\min}^2\right) \tag{4.3}$$

After running for N=45, the results are shown below. Note that for K>5 the solver becomes unstable and is unable to give results. Also note that for all cases the error remains within a 2% range and no decrease is spotted for larger Ks.

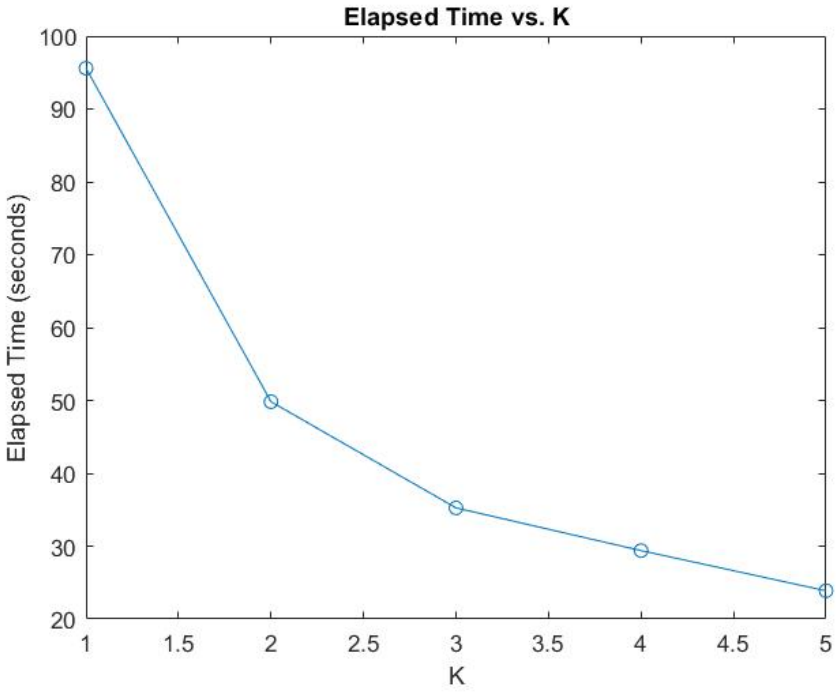


Figure 4.5: Runtime for the factors K compared to the default timestep (N=45).

To conclude this section, a decrease in runtime was successful through both the Courant timestep as well as a simple factor to the default timestep without a significant compromise in error. Values of the CFL of 0.5-0.9 as well as values K<5 were found to benefit the solver's runtime. Note that this analysis gives an overview of the effect of the timestep in both error and runtime of the solver however it takes into account only a specific number N and a specific tolerance. A larger scale analysis with various tolerances and Ns would be required to fully understand the effect of the timestep.

4.2. Tolerance analysis

We saw previously that there was room for improvement in terms of convergence time when it comes to the timestep. However it is equally important to understand how the different stopping criteria (tolerances) influence the convergence and runtime of the solver. For that reason, similar analyses as before are done (centerline error and runtime), now with varying tolerance, keeping N=45 and using the default timestep.

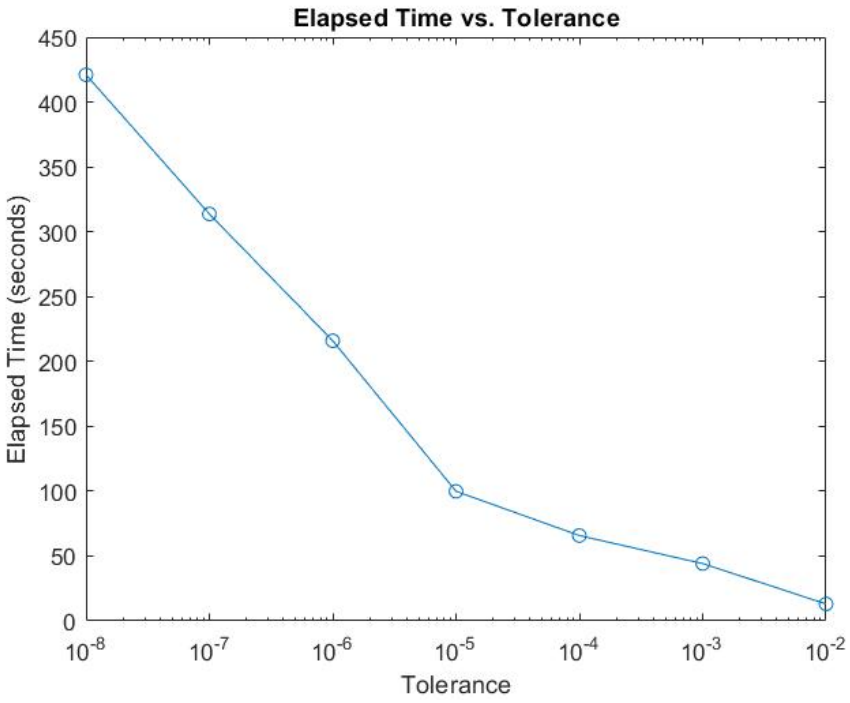


Figure 4.6: Runtime for different tolerance levels (N=45).

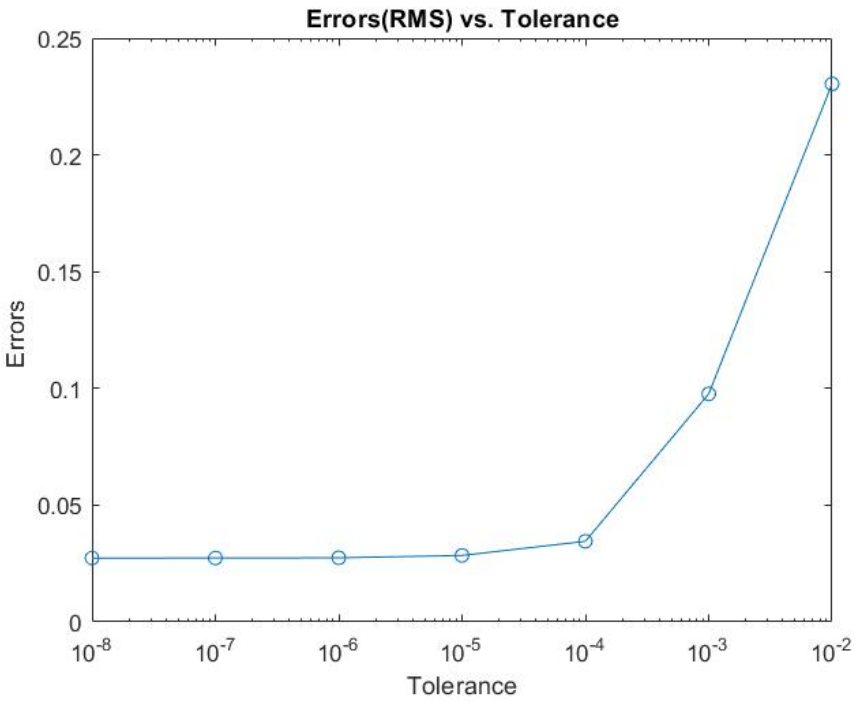


Figure 4.7: Error for the different tolerance levels (N=45).

As seen in the figures, the runtime of the solver linearly decreases for increasing orders of magnitude of the tolerance. The error however seems to exponentially decrease and stagnates approximatelly after 10^{-5}. Of course, when it comes to finding a suitable time stopping criterion one needs to be careful of two things:

- **Stopping too early**: It can cause the solver to terminate without having shown signs of conver-

gence and without having resolved the flow enough.

- **Stopping too late**: It can cause the runtime to unsustainably increase while also reaching accuracy levels of machine precision which could in turn cause very high frequency vibrations on the smallest scales within the flow.

Both cases indicate that there is a ”sweet spot” for the stopping criterion to be chosen and for the sake of this analysis, the choice will be made based on the tolerance level that has the least runtime while concurrently maintaining an error level close to the ”stagnation” error seen in lower orders of magnitude for N=45. Given the previous, sufficient flow resolution and convergence is expected at a tolerance level of 10^{-5}, where the error is marginally higher than, for instance, the one of 10^{-8} but with a runtime more than 4 times less.

Finally, to make sure convergence is sufficiently reached, a visual verification is done for the chosen tolerance point ($N = 45$, $tol = 10^{-5}$) as shown below. Evidently, the flow is sufficiently resolved and converged.

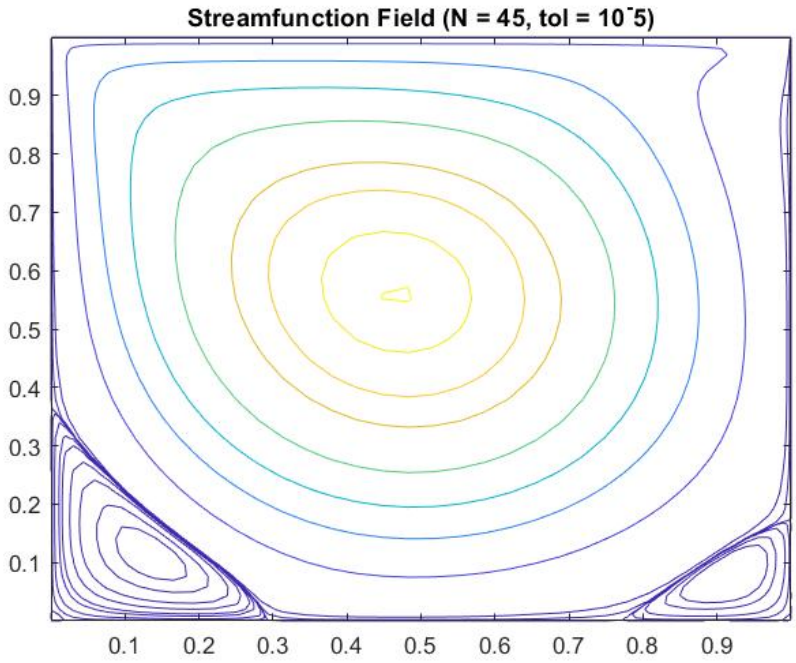


Figure 4.8: Streamfunction contours for the purpose of convergence validation of chosen tolerance 10^{-5} (N=45).

4.3. Pressure matrix singularity

In this section, discussion is made regarding the pressure matrix A appearing in the solver. To give more context, the two relations below show the appearance of the A matrix in the solving procedure and the numerical calculation through Incidence and Hodge matrices respectively.

$$A\vec{P} = f \tag{4.4}$$

$$A = -\tilde{\mathbb{E}}^{(2,1)} H^{\tilde{1},1} \mathbb{E}^{(1,0)} \tag{4.5}$$

One of the posed questions was regarding investigating the pressure matrix A in terms of its properties. For that reason, specific properties of the A matrix are tested and then some discussion is made about the physical meaning behind it.

- `all(sum(A, 2) == 0)`: Checks if the row sum of the pressure matrix is zero.
- `isequal(A, A')`: Checks if the pressure matrix is symmetric.
- `(rank(full(A)) < min(size(A)))`: Checks if the pressure matrix is singular.

The evident singularity and dependence of the dimensions within the A matrix indicate the existence of a degree of freedom in the pressure. This would mean that there is freedom to change the pressure while maintaining all other important quantities in the field.

This coincides with what is known in theory, where the pressure field is expected to be invariant to a global change by a simple constant. This constant which can be added or subtracted from the entire field is therefore represented by that one degree of freedom within the A matrix.

4.4. Integrated vorticity

The integrated vorticity inside a certain flow field represents the double integral over the entire square domain of the vorticity ξ as calculated previously in the solver. Given the discretization, this value would simply be the sum of the values in the ξ matrix after taking the inverse of the diagonal Hodge matrix Ht02 which effectively contains the areas of the cells to be integrated. After realizing this, the calculations are performed for various numbers N and the integrated vorticity is found to constantly be precisely equal to 1. Below, the mathematical intuition behind this fact will be given and a small analysis of the vorticity intensity will be shown against the effect of the Reynolds number.

$$\oint_C \mathbf{V} \cdot d\mathbf{r} = \iint_S (\nabla \times \mathbf{V}) \cdot d\mathbf{S} \tag{4.6}$$

Above is the Stoke's theorem which states that the line integral over a boundary of a vector field is equal to the surface integral of the curl of the vector field over the domain bounded by the line boundary. In context, the formula would transform as follows, relating velocity field and vorticity:

$$\iint_S \omega \, dS = \oint_C \mathbf{V} \cdot d\mathbf{r} \tag{4.7}$$

However, remember that by definition of the circulation:

$$\Gamma = \oint_C \mathbf{V} \cdot d\mathbf{r} \tag{4.8}$$

This indicates that:

$$\Gamma = \iint_S \omega \, dS \tag{4.9}$$

meaning that the circulation is precisely equal to the integrated vorticity over the domain and which, in turn, is equal to the line integral of the velocity field which is equal to 1 since the only contribution from the boundaries is given by the velocity imposed at the ”lid” with magnitude 1.

Finally, the integrated vorticity is plotted as a function of Re. Note that as an indication of the overall vorticity intensity, the absolute values are summed up resulting in a varying positive number.

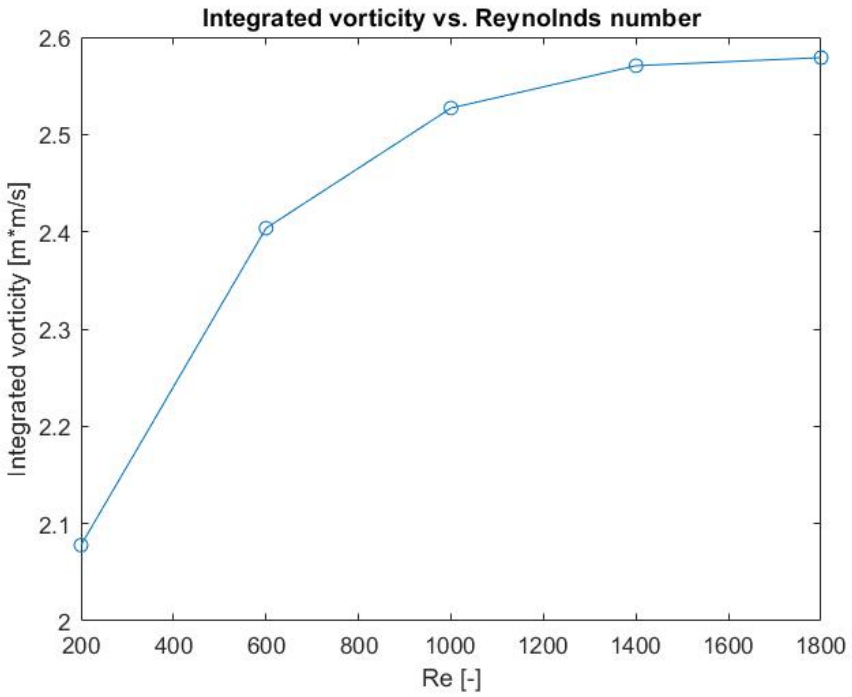


Figure 4.9: Effect of Reynolds number on the integrated vorticity intensity over the flow field (N=35).

It is clear how the Reynolds number affect the generation of vorticity within the flow. At lower Reynolds numbers, the flow is typically more laminar and characterized by smooth, ordered motion with less turbulence. In such flow it is then expected to have lower levels of vorticity as indicated in the figure too. As the Reynolds number increases and the flow transitions to higher values, it becomes more turbulent. That flow will then generally exhibit higher mixing and therefore higher levels of vorticity.