Computational Fluid Dynamics

Ex1

Almog Dobrescu

214254252

Table of Contents

Problem Definition

The Airfoil

The program is designed to generate a "C" type mesh for a NACA 00XX airfoil defined as follows:

$$y(x) = \pm 5 \cdot t \cdot \left[0.2969\sqrt{x_{int} \cdot x} - 0.1260(x_{int} \cdot x) - 0.3516(x_{int} \cdot x)^2 + 0.2843(x_{int} \cdot x)^3 - 0.1015(x_{int} \cdot x)^4\right] \quad (1)$$

where the plus sign corresponds to the upper surface and the minus sign corresponds to the lower surface, t is the maximum airfoil thickness, and $x_{int} = 1.0$ for the classical NACA 00XX airfoil and $x_{int} = 1.008930411365$ for the modified NACA 00XX airfoil.

The classical NACA 00XX airfoil has a non-closed or blunt trailing edge. The modified NACA 00XX is constructed such that the airfoil exactly closes at the trailing edge. The x and y coordinates and t are normalized by the chord length.

Description of the Physical and Computational Domains

Figure 1 contains a description of the physical and computational domains while Figures 2 and 3 contain schematics of the physical and computational domains, respectively.

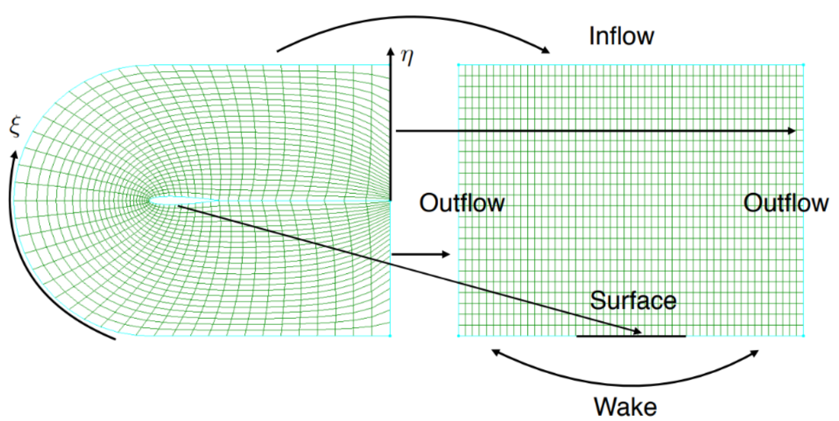

Figure 2: Physical and computational domains

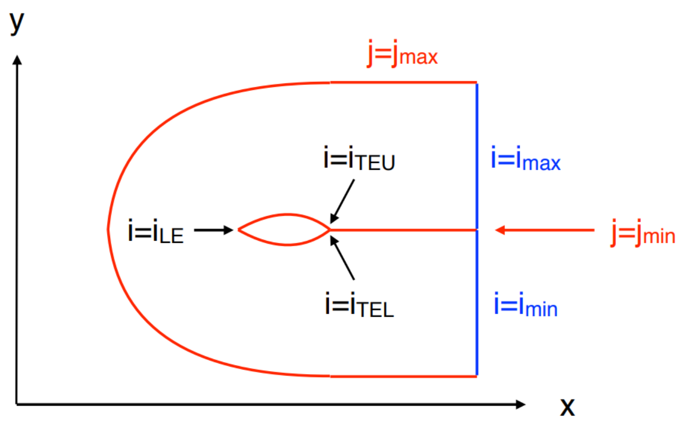

Figure 1: Schematic of the physical domain

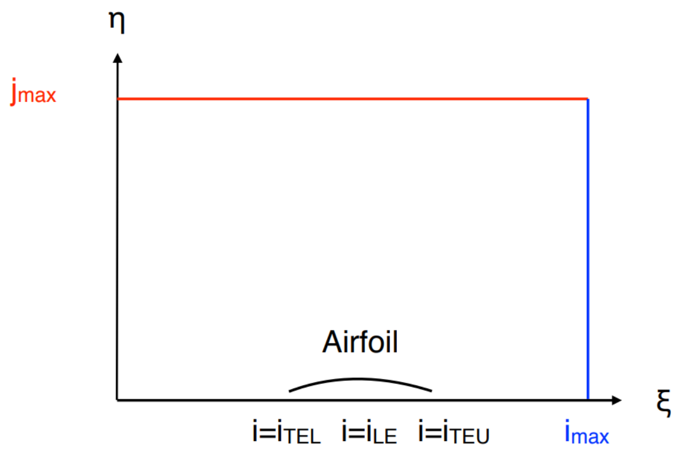

Figure 3: Schematic of the comutational domain

The Governing Equations – Thompson, Thames and Mastin (TTM)

Based on the following Poisson's equation

$$g_{xx} + g_{yy} = S \tag{2}$$

Solving twice, once for each coordinate $(2 - D)$

$$\xi_{xx} + \xi_{yy} = P(\xi, \eta) \tag{3}$$

$$\eta_{xx} + \eta_{yy} = Q(\xi, \eta) \tag{4}$$

Where P and Q are the control functions.

After applying the mapping and the transformation to the curvilinear coordinates system, the transformed equations are

$$\begin{aligned} \alpha x_{\xi\xi} - 2\beta x_{\xi\eta} + \gamma x_{\eta\eta} &= -J^2\left(P x_\xi + Q x_\eta\right) \\ \alpha y_{\xi\xi} - 2\beta y_{\xi\eta} + \gamma y_{\eta\eta} &= -J^2\left(P y_\xi + Q y_\eta\right) \end{aligned} \tag{5}$$

$$\begin{aligned} \alpha &= x_\eta^2 + y_\eta^2 \\ \beta &= x_\xi x_\eta + y_\xi y_\eta \\ \gamma &= x_\xi^2 + y_\xi^2 \\ J &= x_\xi y_\eta - y_\xi x_\eta \end{aligned} \tag{6}$$

The functions P and Q allow to control the local mesh resolution.

TTM uses the following formulation:

$$P = \phi(\xi, \eta) \cdot \left(\xi_x^2 + \xi_y^2\right) \tag{7}$$

$$Q = \psi(\xi, \eta) \cdot \left(\eta_x^2 + \eta_y^2\right) \tag{8}$$

The resulting equations take the form:

$$\begin{aligned}\alpha\left(x_{\xi\xi}+\phi x_{\xi}\right)-2\beta x_{\xi\eta}+\gamma\left(x_{\eta\eta}+\psi x_{\eta}\right)=0\\ \alpha\left(y_{\xi\xi}+\phi y_{\xi}\right)-2\beta y_{\xi\eta}+\gamma\left(y_{\eta\eta}+\psi y_{\eta}\right)=0\end{aligned} \tag{9}$$

Setting the control functions is by setting the terms in parentheses to be zero on the boundaries as follows

$$\xi=const\begin{cases}x_{\eta\eta}+\psi x_{\eta}=0\\ y_{\eta\eta}+\psi y_{\eta}=0\end{cases} \tag{10}$$

$$\eta=const\begin{cases}x_{\xi\xi}+\psi x_{\xi}=0\\ y_{\xi\xi}+\psi y_{\xi}=0\end{cases} \tag{11}$$

Evaluating the control functions inside the computational domain through linear interpolations

- $\xi = const$

$$\begin{aligned}|y_\eta|>|x_\eta|\rightarrow\psi(\xi_{min},\eta)&=-\left.\frac{y_{\eta\eta}}{y_\eta}\right|_{\xi=\xi_{min}}\\ |y_\eta|<|x_\eta|\rightarrow\psi(\xi_{min},\eta)&=-\left.\frac{x_{\eta\eta}}{x_\eta}\right|_{\xi=\xi_{min}}\\ |y_\eta|>|x_\eta|\rightarrow\psi(\xi_{max},\eta)&=-\left.\frac{y_{\eta\eta}}{y_\eta}\right|_{\xi=\xi_{max}}\\ |y_\eta|<|x_\eta|\rightarrow\psi(\xi_{max},\eta)&=-\left.\frac{x_{\eta\eta}}{x_\eta}\right|_{\xi=\xi_{max}}\end{aligned} \tag{12}$$

- $\eta = const$

$$\begin{aligned}|x_\xi|>|y_\xi|\rightarrow\phi(\xi,\eta_{min})&=-\left.\frac{x_{\xi\xi}}{x_\xi}\right|_{\eta=\eta_{min}}\\ |x_\xi|<|y_\xi|\rightarrow\phi(\xi,\eta_{min})&=-\left.\frac{y_{\xi\xi}}{y_\xi}\right|_{\eta=\eta_{min}}\\ |x_\xi|>|y_\xi|\rightarrow\phi(\xi,\eta_{max})&=-\left.\frac{x_{\xi\xi}}{x_\xi}\right|_{\eta=\eta_{max}}\\ |x_\xi|<|y_\xi|\rightarrow\phi(\xi,\eta_{max})&=-\left.\frac{y_{\xi\xi}}{y_\xi}\right|_{\eta=\eta_{max}}\end{aligned} \tag{13}$$

Indexes

From here, it is assumed that the programming language is "C" and therefore it is assumed that all arrays start from zero. For example, if the number of grid points in the ξ coordinate direction is ni then $\xi_{min} = 0$ and $\xi_{max} = i_{max} = ni - 1$. Similarly, $\eta_{min} = 0$ and $\eta_{max} = j_{max} = nj - 1$ (nj being the number of grid points in the η coordinate direction).

Boundary and Initial Conditions

Around the Airfoil

Assuming that the leading edge is placed at the origin of the coordinate system, grid points may be distributed along the airfoil surface using:

$$\left.\begin{aligned} \Delta x &= \frac{1.0}{i_{LE} - i_{TEL}} \\ x_{i,j_{min}} &= 1 - \cos\left(\frac{\pi(i_{LE} - i)\Delta x}{2}\right) \\ y_{i,j_{min}} &= y\left(x_{i,j_{min}}\right) \end{aligned}\right\} i = i_{TEL}, \dots, i_{LE} \tag{14}$$

The stretching function utilized above allows to stretch the grid points toward the leading-edge to provide sufficient points to capture the leading-edge suction phenomenon.

Since the airfoil in question is symmetric, one may find the upper surface grid points, $i = i_{LE}, \dots, i_{TEU}$, using a mirror transformation (use the " $-$ " sign in Equation 1 for y values of the lower surface and the " $+$ " sign for the upper surface).

Airfoil Wake

The rest of the grid point coordinates for the boundary $j = j_{min}$ may be found using:

$$\begin{aligned} &\left.\begin{aligned} x_{i,j_{min}} &= x_{i-1,j_{min}} + \left(x_{i-1,j_{min}} - x_{i-2,j_{min}}\right) * XSF \\ y_{i,j_{min}} &= 0 \end{aligned}\right\} i = i_{TEU} + 1, \dots, i_{max} \\ &\left.\begin{aligned} x_{i,j_{min}} &= x_{i_{max}-i,j_{min}} \\ y_{i,j_{min}} &= 0 \end{aligned}\right\} i = i_{min}, \dots, i_{TEL} - 1 \end{aligned} \tag{15}$$

Where $XSF > 1$ allows stretching of the grid points.

Exit-Boundary

The exit-boundary grid points are given by the following boundary conditions:

$$\begin{aligned} y_{i_{max},j_{max}+1} &= \Delta y \\ y_{i_{max},j} &= y_{i_{max},j-1} + \left(y_{i_{max},j-1} - y_{i_{max},j-2}\right) * YSF; \quad j = j_{min} + 2, \dots, j_{max} \\ x_{i_{max},j} &= x_{i_{max},j_{min}}; \quad j = j_{min} + 1, \dots, j_{max} \\ y_{i_{min},j} &= -y_{i_{max},j}; \quad j = j_{min} + 1, \dots, j_{max} \\ x_{i_{min},j} &= x_{i_{min},j_{min}}; \quad j = j_{min} + 1, \dots, j_{max} \end{aligned} \tag{16}$$

The quantities $\Delta y, XSF, YSF, i_{max}, j_{max}$ are supplied by the user. The value of Δy is set based on the flow model (inviscid, laminar, or turbulent), XSF and YSF should be less than 1.3 and the dimensions i_{max} and j_{max} are chose to have a sufficiently long wake and a free stream boundary far enough from the airfoil (at least 5 chords).

Outer Boundary

The number of points on the outer boundary is known (ni). Using equal spacing to determine the distance between points:

$$\begin{aligned} L &= 2x_{i_{max},j_{max}} + \pi y_{i_{max},j_{max}} \\ d &= \frac{L}{i_{max}} \end{aligned} \tag{17}$$

Determining the x and y values along the outer boundary will happen in two steps for the upper side and mirrored for the lower side:

Horizontal Section

It isn't known how much points are in the horizontal section. So a marching method is used:

$$\left.\begin{aligned} x_{i,j_{max}} &= x_{i+1,j_{max}} - d \\ y_{i,j_{max}} &= y_{i_{max},j_{max}} \end{aligned}\right\} i = i_{max} - 1, \dots, i_{x0} \tag{18}$$

Where i_{x0} is the index for it:

$$x_{i_{x0}-1,j_{max}} < 0 \tag{19}$$

Arc Section

Because $x_{i0} \neq 0$, the angle of the arc is not exactly $\frac{\pi}{2}$ (90°). The addition to the angle is ϵ:

$$\begin{aligned} \epsilon &= \operatorname{atan}\left(\frac{x_{i_{x0},j_{max}} - 0}{y_{i_{max},j_{max}}}\right) \\ \theta &= \frac{\pi}{2} + \epsilon \\ \Delta\theta &= \frac{\theta}{i_{x0} - \frac{i_{max}}{2} + 1} \end{aligned} \tag{20}$$

$$\left.\begin{aligned} x_{i+\frac{i_{max}}{2},j_{max}} &= -y_{i_{max},j_{max}} \cdot \cos(\Delta\theta \cdot i) \\ y_{i+\frac{i_{max}}{2},j_{max}} &= y_{i_{max},j_{max}} \cdot \sin(\Delta\theta \cdot i) \end{aligned}\right\} i = 0, \dots, i_{x0} - \frac{i_{max}}{2} \tag{21}$$

X, Y Matrices

Using interpolation to determine the x and y values inside the domain.

$$\left.\begin{aligned} x_{i,j} &= \frac{x_{i,j_{max}} - x_{i,j_{min}}}{j_{max}} \cdot j + x_{i,j_{min}} \\ y_{i,j} &= \frac{y_{i,j_{max}} - y_{i,j_{min}}}{j_{max}} \cdot j + y_{i,j_{min}} \end{aligned}\right\} j = 1, \dots, j_{max} - 1 \tag{22}$$

The Numerical Scheme

Poiison's equations for the x and y coordinate in curvilinear coordinates are given by:

$$\begin{aligned} \alpha\left(x_{\xi\xi} + \phi x_{\xi}\right) - 2\beta x_{\xi\eta} + \gamma\left(x_{\eta\eta} + \psi x_{\eta}\right) = 0 \\ \alpha\left(y_{\xi\xi} + \phi y_{\xi}\right) - 2\beta y_{\xi\eta} + \gamma\left(y_{\eta\eta} + \psi y_{\eta}\right) = 0 \end{aligned} \tag{23}$$

$$\begin{aligned} \alpha &= x_{\eta}^{2} + y_{\eta}^{2} \\ \beta &= x_{\xi}x_{\eta} + y_{\xi}y_{\eta} \\ \gamma &= x_{\xi}^{2} + y_{\xi}^{2} \end{aligned} \tag{24}$$

The derivatives

Calculating the second order first and second derivatives of x using a combination of operators, when $\Delta i, \Delta j = 1$:

$$\begin{aligned} x_{\xi} &= \frac{\mu_{\xi}\delta_{\xi}x_{i,j}}{\Delta i} + O(\Delta i^{2}) = \\ &= \frac{\mu\left(x_{i+\frac{1}{2},j} - x_{i-\frac{1}{2},j}\right)}{\Delta i} + O(\Delta i^{2}) = \\ &= \frac{\left(x_{i+1,j} + x_{i,j} - x_{i,j} - x_{i-1,j}\right)}{2 \cdot \Delta i} + O(\Delta i^{2}) = \\ &= \frac{\left(x_{i+1,j} - x_{i-1,j}\right)}{2 \cdot \Delta i} + O(\Delta i^{2}) \approx \frac{\left(x_{i+1,j} - x_{i-1,j}\right)}{2 \cdot \Delta i} \end{aligned} \tag{25}$$

$$\begin{aligned} x_{\xi\xi} &= \frac{\delta_{\xi}^{2}x_{i,j}}{\Delta i^{2}} + O(\Delta i^{2}) = \\ &= \frac{\delta_{\xi}\left(x_{i+\frac{1}{2},j} - x_{i-\frac{1}{2},j}\right)}{\Delta i^{2}} + O(\Delta i^{2}) = \\ &= \frac{x_{i+1,j} - x_{i,j} - x_{i,j} + x_{i-1,j}}{\Delta i^{2}} + O(\Delta i^{2}) = \\ &= \frac{x_{i+1,j} - 2 \cdot x_{i,j} + x_{i-1,j}}{\Delta i^{2}} + O(\Delta i^{2}) \approx \frac{x_{i+1,j} - 2 \cdot x_{i,j} + x_{i-1,j}}{\Delta i^{2}} \end{aligned} \tag{26}$$

The derivatives in the η direction are calculated ate the exact same way:

$$x_{\eta} = \frac{\mu_{\eta}\delta_{\eta}x_{i,j}}{\Delta j} + O(\Delta j^{2}) \approx \frac{\left(x_{i,j+1} - x_{i,j-1}\right)}{2 \cdot \Delta j} \tag{27}$$

$$x_{\eta\eta} = \frac{\delta_{\eta}^{2}x_{i,j}}{\Delta j^{2}} + O(\Delta j^{2}) \approx \frac{x_{i,j+1} - 2 \cdot x_{i,j} + x_{i,j-1}}{\Delta j^{2}} \tag{28}$$

The derivatives at the y direction are calculated at the exact same way:

$$
\begin{aligned}
y_\xi &= \frac{\mu_\xi \delta_\xi y_{i,j}}{\Delta i} + O(\Delta i^2) \approx \frac{(y_{i+1,j} - y_{i-1,j})}{2 \cdot \Delta i} \\
y_{\xi\xi} &= \frac{\delta_\xi^2 y_{i,j}}{\Delta i^2} + O(\Delta i^2) \approx \frac{y_{i+1,j} - 2 \cdot y_{i,j} + y_{i-1,j}}{\Delta i^2} \\
y_\eta &= \frac{\mu_\eta \delta_\eta y_{i,j}}{\Delta j} + O(\Delta j^2) \approx \frac{(y_{i,j+1} - y_{i,j-1})}{2 \cdot \Delta j} \\
y_{\eta\eta} &= \frac{\delta_\eta^2 y_{i,j}}{\Delta j^2} + O(\Delta j^2) \approx \frac{y_{i,j+1} - 2 \cdot y_{i,j} + y_{i,j-1}}{\Delta j^2}
\end{aligned}
\tag{29}
$$

The Mixed derivatives are calculated as follows:

$$
\begin{aligned}
x_{\xi\eta} &= \frac{\mu_\eta \delta_\eta \left(\frac{\mu_\xi \delta_\xi (x_{i,j})}{\Delta i} \right)}{\Delta j} + O(\Delta i^2 + \Delta j^2) = \\
&= \frac{\mu_\eta \delta_\eta \left(\frac{(x_{i+1,j} - x_{i-1,j})}{2 \cdot \Delta i} \right)}{\Delta j} + O(\Delta i^2 + \Delta j^2) = \\
&\frac{x_{i+1,j+1} - x_{i-1,j+1} - (x_{i+1,j-1} - x_{i-1,j-1})}{4 \cdot \Delta i \cdot \Delta j} + O(\Delta i^2 + \Delta j^2) \approx \\
&\frac{x_{i+1,j+1} - x_{i-1,j+1} - x_{i+1,j-1} + x_{i-1,j-1}}{4 \Delta i \Delta j}
\end{aligned}
\tag{30}
$$

$$
y_{\xi\eta} = \frac{y_{i+1,j+1} - y_{i-1,j+1} - y_{i+1,j-1} + y_{i-1,j-1}}{4 \Delta i \Delta j} \tag{31}
$$

Approximate Factorization for Laplace Equation

Adding to the Laplace equation a time derivative. The new equation takes the form:

$$
-f_t + f_{xx} + f_{yy} = 0 \tag{32}
$$

The difference equation takes the form:

$$
\begin{aligned}
-\frac{-f_{i,j}^{n+1} - f_{i,j}^n}{\Delta t} + \frac{1}{\Delta x^2} \underbrace{\left(f_{i+1,j}^{n+1} - 2 f_{i,j}^{n+1} + f_{i-1,j}^{n+1} \right)}_{\delta_{xx} f_{i,j}^{n+1}} \\
+ \frac{1}{\Delta y^2} \underbrace{\left(f_{i,j+1}^{n+1} - 2 f_{i,j}^{n+1} + f_{i,j-1}^{n+1} \right)}_{\delta_{yy} f_{i,j}^{n+1}} = 0
\end{aligned}
\tag{33}
$$

The difference equation in delta form:

$$
-\Delta f_{i,j}^n + \frac{\Delta t}{\Delta x^2} \delta_{xx} \Delta f_{i,j}^n + \frac{\Delta t}{\Delta y^2} \delta_{yy} \Delta f_{i,j}^n + \frac{\Delta t}{\Delta x^2} \delta_{xx} f_{i,j}^n + \frac{\Delta t}{\Delta y^2} \delta_{yy} f_{i,j}^n = 0 \tag{34}
$$

The difference equation in operator form:

$$\left(-I+\frac{\Delta t}{\Delta x^2}\delta_{xx}+\frac{\Delta t}{\Delta y^2}\delta_{yy}\right)\Delta f_{i,j}^n+\Delta t L f_{i,j}^n=0$$
$$L=\left(\frac{\delta_{xx}}{\Delta x^2}+\frac{\delta_{yy}}{\Delta y^2}\right) \tag{35}$$

The operator N is:

$$N=\left(-I+\frac{\Delta t}{\Delta x^2}\delta_{xx}+\frac{\Delta t}{\Delta y^2}\delta_{yy}\right) \tag{36}$$

The term Δt acts as a relaxation parameter (ω).

Suggested approximate factorization:

$$\underbrace{\left(-I+\frac{\Delta t}{\Delta x^2}\delta_{xx}+\frac{\Delta t}{\Delta y^2}\delta_{yy}\right)}_{N}\approx-\underbrace{\left(I-\frac{\Delta t}{\Delta x^2}\delta_{xx}\right)}_{N1}\cdot\underbrace{\left(I-\frac{\Delta t}{\Delta y^2}\delta_{yy}\right)}_{N2} \tag{37}$$

The factorization error:

$$N1\cdot N2=\underbrace{-I+\frac{\Delta t}{\Delta x^2}\delta_{xx}+\frac{\Delta t}{\Delta y^2}\delta_{yy}}_{N}+\underbrace{-\frac{\Delta t^2}{\Delta x^2\Delta y^2}\delta_{xx}\delta_{yy}}_{Error} \tag{38}$$

As we can see, the error is of a second order. So, it can be neglected.

ADI Solution Procedure

AF scheme:

$$\left(I-\frac{\Delta t}{\Delta x^2}\delta_{xx}\right)\left(I-\frac{\Delta t}{\Delta y^2}\delta_{yy}\right)\Delta f_{i,j}^n=\Delta t L f_{i,j}^n \tag{39}$$

Define:

$$\left(I-\frac{\Delta t}{\Delta y^2}\delta_{yy}\right)\Delta f_{i,j}^n\triangleq\Delta f_{i,j}^{n\,*} \tag{40}$$

Solve:

$$\left(I-\frac{\Delta t}{\Delta x^2}\delta_{xx}\right)\Delta f_{i,j}^{n\,*}=\Delta t L f_{i,j}^n=RHS$$
$$\left(I-\frac{\Delta t}{\Delta y^2}\delta_{yy}\right)\Delta f_{i,j}^n=\Delta f_{i,j}^{n\,*}=RHS \tag{41}$$

Update:

$$f_{i,j}^{n+1}=f_{i,j}^n+\Delta f_{i,j}^n \tag{42}$$

Transformig to curvilinear coordinate system

In general:

$$\left(I - \frac{\Delta t}{\Delta x^2}\delta_{xx}\right)\left(I - \frac{\Delta t}{\Delta y^2}\delta_{yy}\right)\Delta f_{i,j}^n = \Delta t L f_{i,j}^n$$
$$\downarrow$$
$$\begin{gathered}(r - \alpha\delta_{\xi\xi})(r - \gamma\delta_{\eta\eta})C_{x_{i,j}}^n = r\omega L x_{i,j}^n \\ C_{x_{i,j}}^n = x_{i,j}^{n+1} - x_{i,j}^n \\ (r - \alpha\delta_{\xi\xi})(r - \gamma\delta_{\eta\eta})C_{y_{i,j}}^n = r\omega L y_{i,j}^n \\ C_{y_{i,j}}^n = y_{i,j}^{n+1} - y_{i,j}^n\end{gathered} \tag{43}$$

The parameters r and ω are used to accelerate the convergence and sometimes even prevent divergence.

The operators $Lx_{i,j}^n$ and $Ly_{i,j}^n$ are defined by:

$$\begin{aligned}Lx_{i,j}^n = \alpha_{i,j}^n & \left[\left(x_{i+1,j}^n - 2x_{i,j}^n + x_{i-1,j}^n\right) + \frac{\phi_{i,j}}{2}\left(x_{i+1,j}^n - x_{i-1,j}^n\right)\right] \\ & - \frac{1}{2}\beta_{i,j}^n\left[x_{i+1,j+1}^n - x_{i+1,j-1}^n - x_{i-1,j+1}^n + x_{i-1,j-1}^n\right] \\ & + \gamma_{i,j}^n\left[\left(x_{i,j+1}^n - 2x_{i,j}^n + x_{i,j-1}^n\right) + \frac{\psi_{i,j}}{2}\left(x_{i,j+1}^n - x_{i,j-1}^n\right)\right]\end{aligned} \tag{44}$$

$$\begin{aligned}Ly_{i,j}^n = \alpha_{i,j}^n & \left[\left(y_{i+1,j}^n - 2y_{i,j}^n + y_{i-1,j}^n\right) + \frac{\phi_{i,j}}{2}\left(y_{i+1,j}^n - y_{i-1,j}^n\right)\right] \\ & - \frac{1}{2}\beta_{i,j}^n\left[y_{i+1,j+1}^n - y_{i+1,j-1}^n - y_{i-1,j+1}^n + y_{i-1,j-1}^n\right] \\ & + \gamma_{i,j}^n\left[\left(y_{i,j+1}^n - 2y_{i,j}^n + y_{i,j-1}^n\right) + \frac{\psi_{i,j}}{2}\left(y_{i,j+1}^n - y_{i,j-1}^n\right)\right]\end{aligned} \tag{45}$$

The scheme is carried out in two sweeps:

- Sweep 1:

$$\begin{cases}(r - \alpha\delta_{\xi\xi})f_{x_{i,j}}^n = r\omega L x_{i,j}^n \\ (r - \alpha\delta_{\xi\xi})f_{y_{i,j}}^n = r\omega L y_{i,j}^n\end{cases} \tag{46}$$

- Sweep 2:

$$\begin{cases}(r - \alpha\delta_{\xi\xi})C_{x_{i,j}}^n = f_{x_{i,j}}^n \\ (r - \alpha\delta_{\xi\xi})C_{y_{i,j}}^n = f_{y_{i,j}}^n\end{cases} \tag{47}$$

Line Jacobi

The boundary conditions for f_x, f_y, C_x, C_y are zero (the boundary does not move), thus, the LHS and RHS boundary conditions are:

$$\begin{gathered} A_0 = 0, A_{max} = 0 \\ B_0 = 1, B_{max} = 1 \\ C_0 = 0, C_{max} = 0 \\ D_0 = 0, D_{max} = 0 \end{gathered} \tag{48}$$

The tri-diagonal system that is solved looks as follows:

$$\begin{pmatrix} B_0 & 0 & 0 & \dots & \dots & 0 & 0 \\ A_1 & B_1 & C_1 & 0 & \dots & \dots & 0 \\ 0 & \ddots & \ddots & \ddots & 0 & \dots & 0 \\ 0 & 0 & A_{i\ or\ j} & B_{i\ or\ j} & C_{i\ or\ j} & 0 & 0 \\ 0 & \dots & 0 & \ddots & \ddots & \ddots & 0 \\ 0 & \dots & \dots & 0 & A_{max-1} & B_{max-1} & C_{max-1} \\ 0 & 0 & \dots & \dots & 0 & 0 & B_{max} \end{pmatrix} \cdot \begin{pmatrix} f_{x_0} \\ f_{x_1} \\ \dots \\ \dots \\ \dots \\ f_{x_{max-1}} \\ f_{x_{max}} \end{pmatrix} = \begin{pmatrix} 0 \\ D_1 \\ \dots \\ \dots \\ \dots \\ D_{max-1} \\ 0 \end{pmatrix} \tag{49}$$

The equation for x of sweep 1 is set up and inverted as follows:

$$(r - \alpha\delta_{\xi\xi}) f_{x_{i,j}}^n = r\omega L x_{i,j}^n \tag{50}$$

$$\underbrace{-\alpha_{i,j}^n}_{A} f_{x_{i-1,j}}^n + \underbrace{(r + 2\alpha_{i,j}^n)}_{B} f_{x_{i,j}}^n \underbrace{-\alpha_{i,j}^n}_{C} f_{x_{i+1,j}}^n = \underbrace{r\omega L x_{i,j}^n}_{D} \tag{51}$$

The equation for y of sweep 1 is set up and inverted as follows:

$$(r - \alpha\delta_{\xi\xi}) f_{y_{i,j}}^n = r\omega L y_{i,j}^n \tag{52}$$

$$\underbrace{-\alpha_{i,j}^n}_{A} f_{y_{i-1,j}}^n + \underbrace{(r + 2\alpha_{i,j}^n)}_{B} f_{y_{i,j}}^n \underbrace{-\alpha_{i,j}^n}_{C} f_{y_{i+1,j}}^n = \underbrace{r\omega L y_{i,j}^n}_{D} \tag{53}$$

The equation for x of sweep 2 is set up and inverted as follows:

$$(r - \alpha\delta_{\xi\xi}) C_{x_{i,j}}^n = f_{x_{i,j}}^n \tag{54}$$

$$\underbrace{-\alpha_{i,j}^n}_{A} C_{x_{i-1,j}}^n + \underbrace{(r + 2\alpha_{i,j}^n)}_{B} C_{x_{i,j}}^n \underbrace{-\alpha_{i,j}^n}_{C} C_{x_{i+1,j}}^n = \underbrace{f_{x_{i,j}}^n}_{D} \tag{55}$$

The equation for y of sweep 22 is set up and inverted as follows:

$$(r - \alpha\delta_{\xi\xi}) C_{y_{i,j}}^n = f_{y_{i,j}}^n \tag{56}$$

$$\underbrace{-\alpha_{i,j}^n}_{A} C_{y_{i-1,j}}^n + \underbrace{(r + 2\alpha_{i,j}^n)}_{B} C_{y_{i,j}}^n \underbrace{-\alpha_{i,j}^n}_{C} C_{y_{i+1,j}}^n = \underbrace{f_{y_{i,j}}^n}_{D} \tag{57}$$

Functional Description

Flow Chart

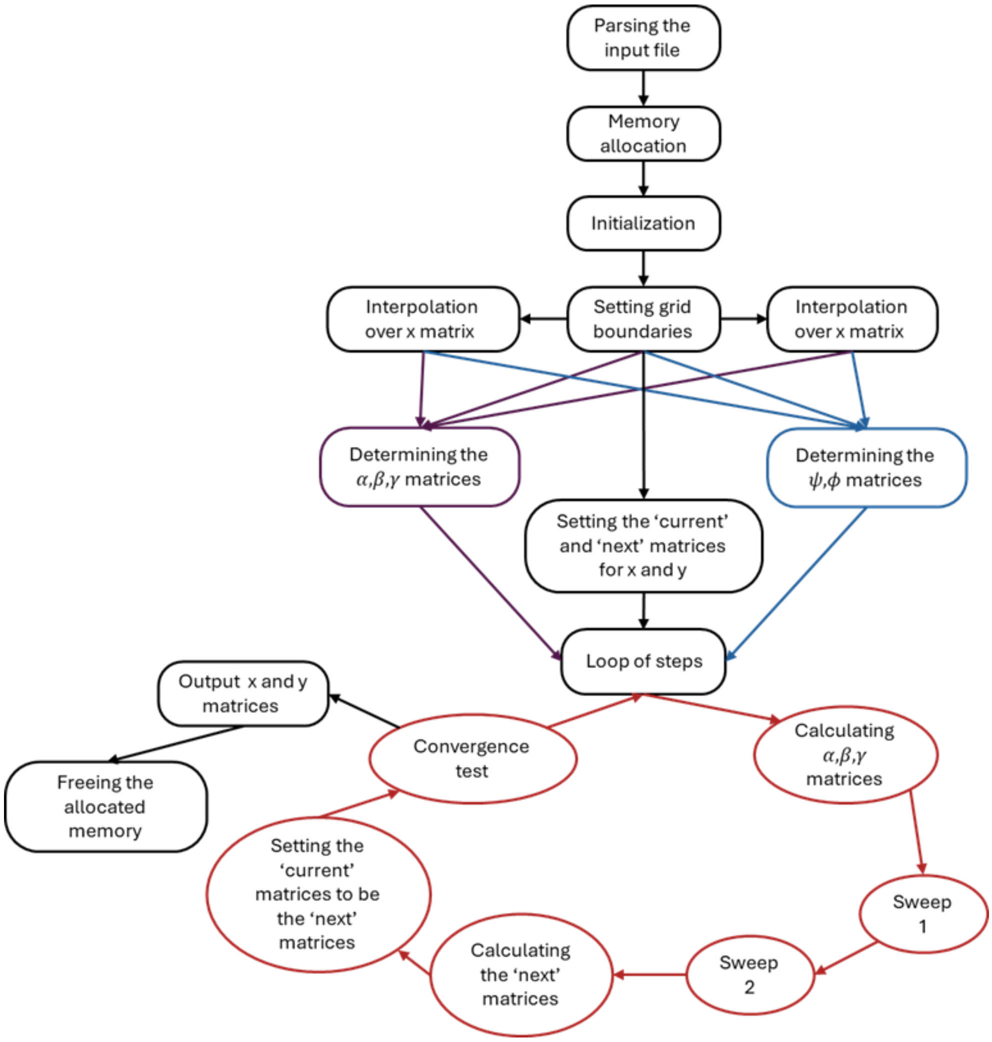

Figure 4: Flow Chart

The functions in the program

The functions that are included in the program and explanation about them are included in the code itself.

Results

Different ϕ and ψ Cases

$\phi = \psi = 0$ (without control)

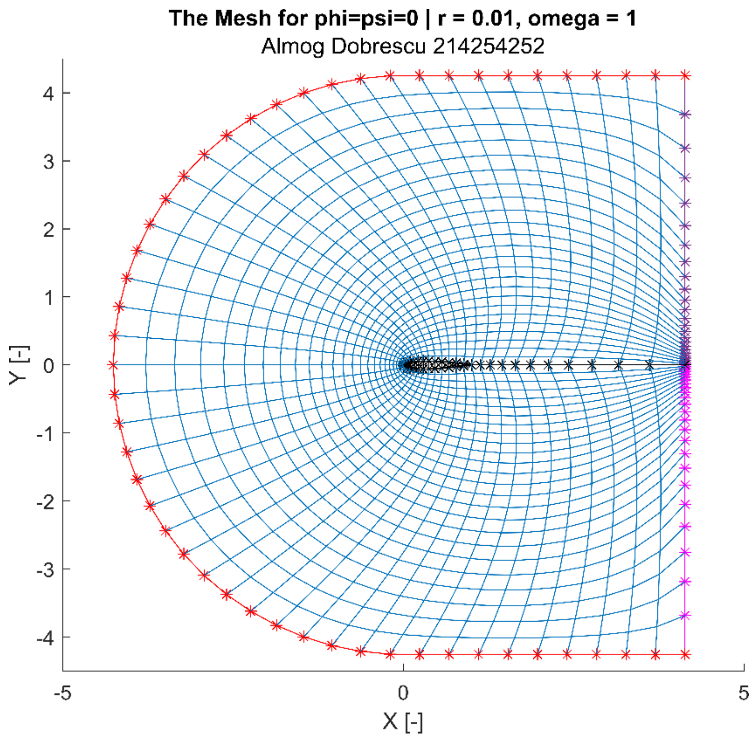

Figure 5: The mesh of $\phi = 0, \psi = 0$

As we can see the boundary conditions did not change. The lines are pretty much equally spaced.

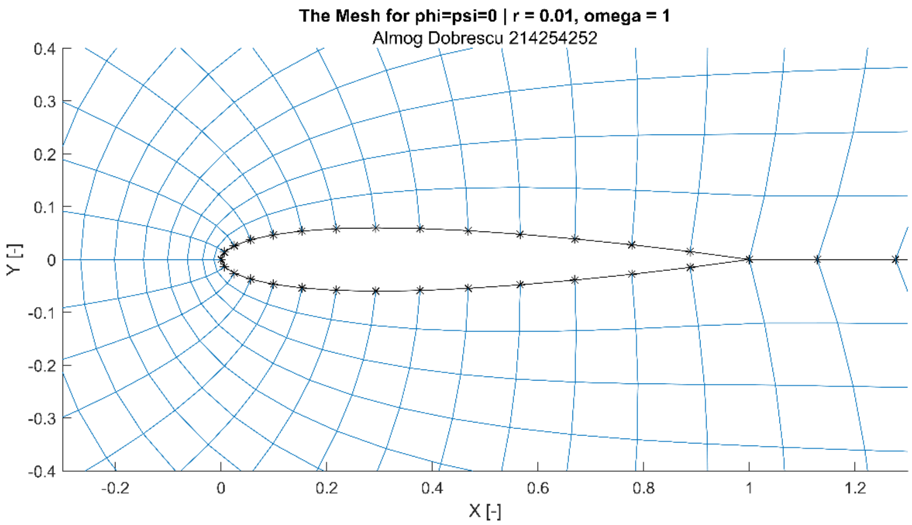

Figure 6: The mesh of $\phi = 0, \psi = 0$ - close up

We can see that there aren't a lot of points at the trailing edge and the wake of the airfoil.

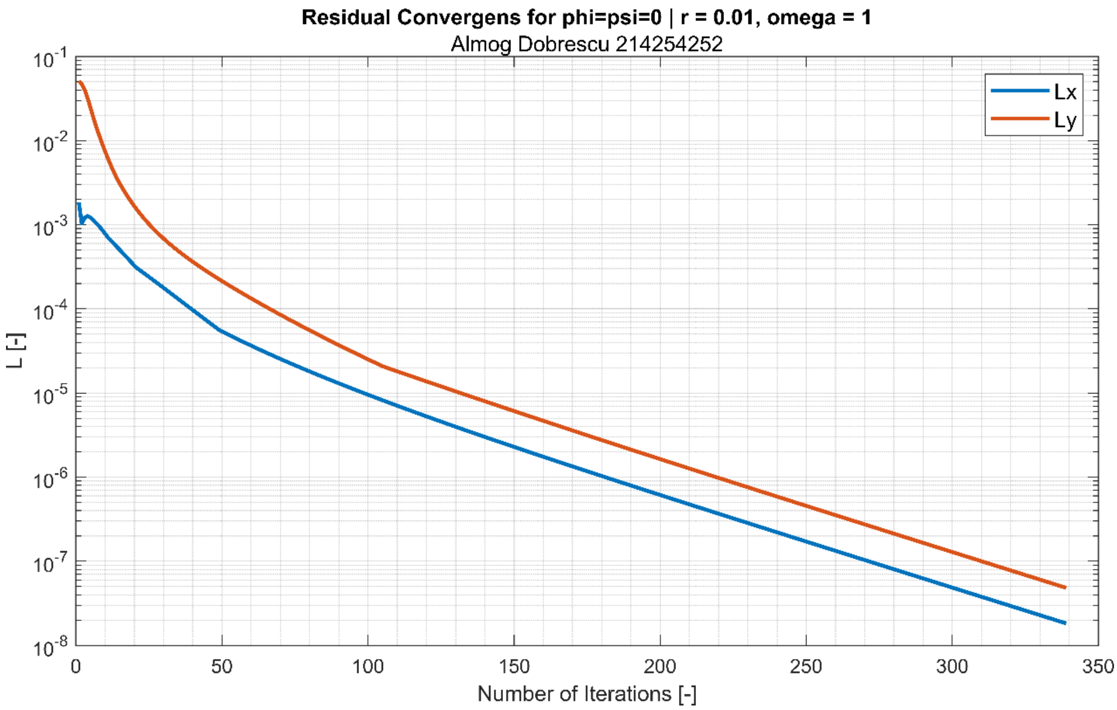

Figure 7: The residual convergens for $\phi = 0, \psi = 0$

We can see that after about 100 iterations the number of iterations decays exponential.

$\phi = 0, \psi$ is Calculated From the BC

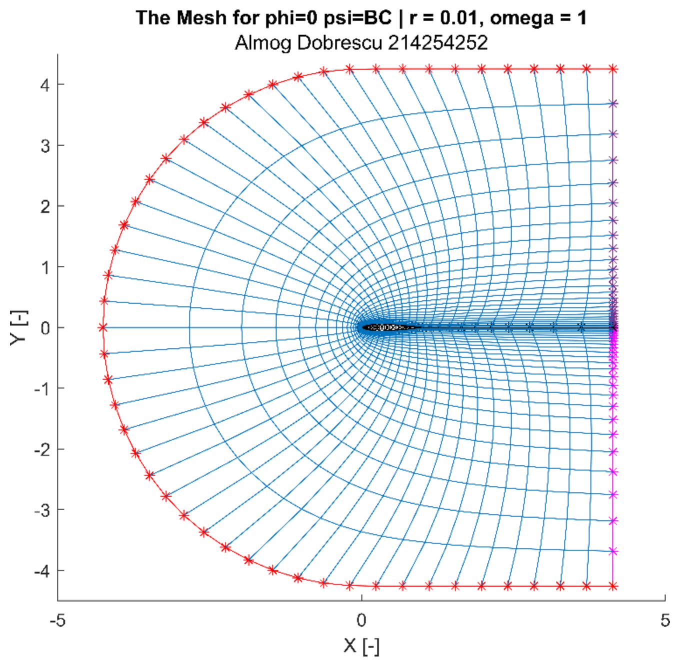

Figure 8: The mesh of $\phi = 0, \psi = BC$

We can see that only the lines in the ξ direction are affected since the ψ function is calculated in that direction. The lines are dens near the airfoil and the wake and spacious at the free flow area.

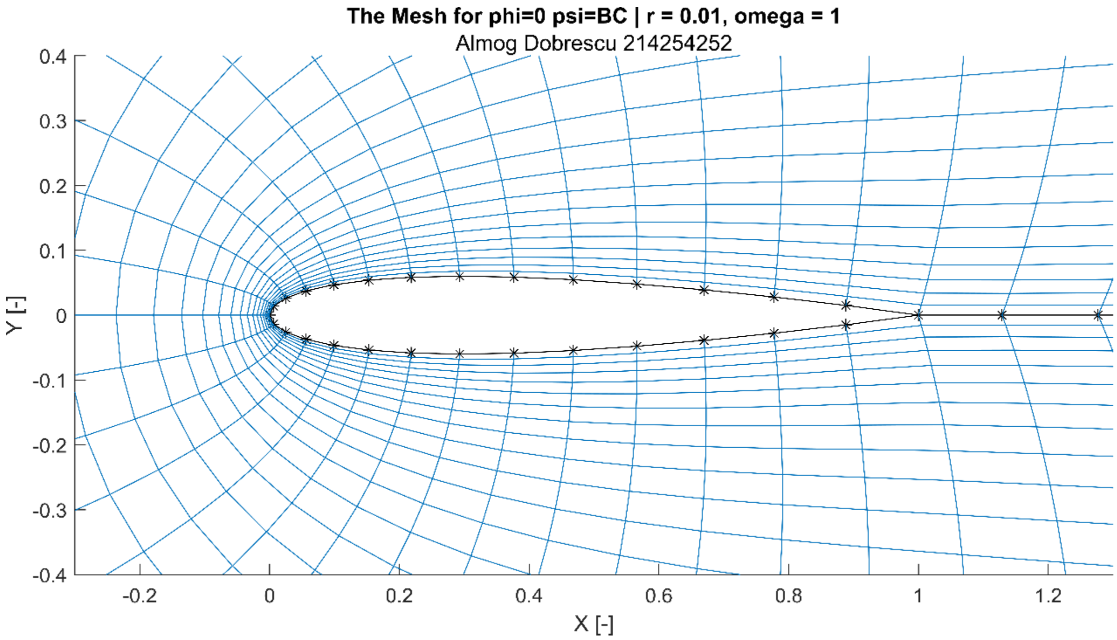

Figure 9: The mesh of $\phi = 0, \psi = BC$ - close up

We can see that there are a lot of points near the leading edge and more point at the wake compared the $\phi = \psi = 0$ case.

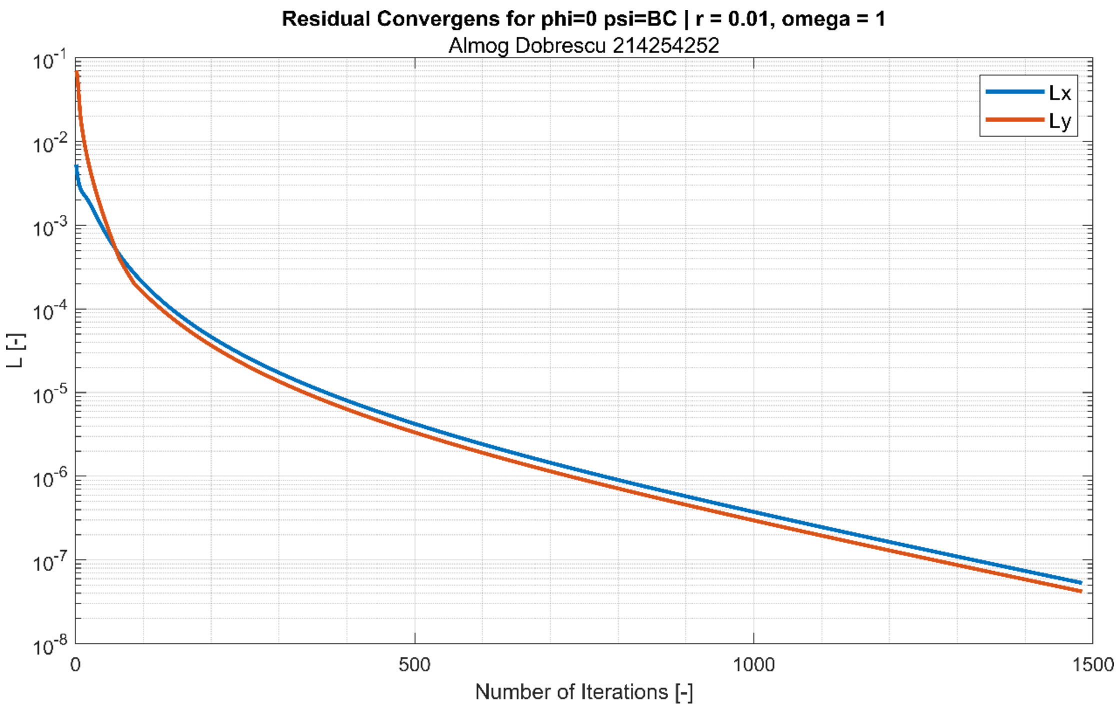

Figure 10: The residual convergens for $\phi = 0, \psi = BC$

We can see that after about 100 iterations the number of iterations decays exponential. Moreover, activating the control function in the ξ direction increased the iteration number by five times compared the $\phi = 0, \psi = 0$ case.

$\psi = 0, \phi$ is Calculated From the BC

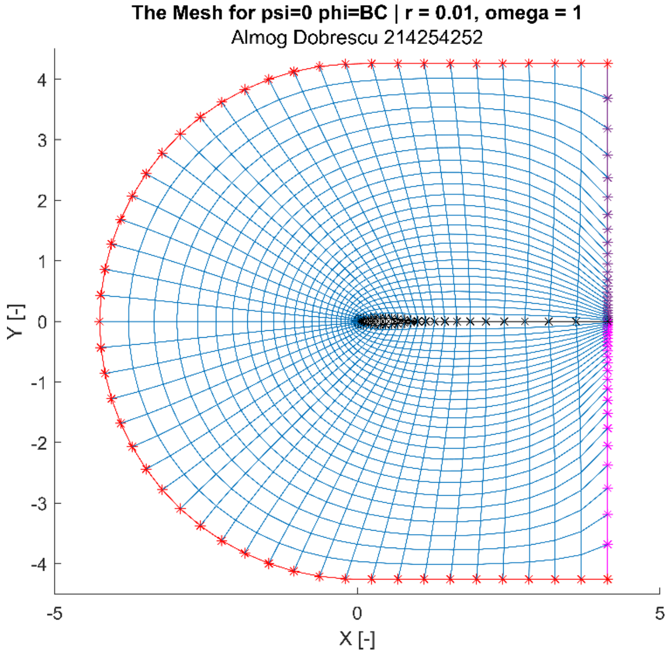

Figure 11: The mesh of $\psi = 0,\ \phi = BC$

We can see that only the lines in the η direction are affected since the ϕ function is calculated in that direction. The lines are a bit denser near the airfoil and at the end of the wake and more spacious away from the center.

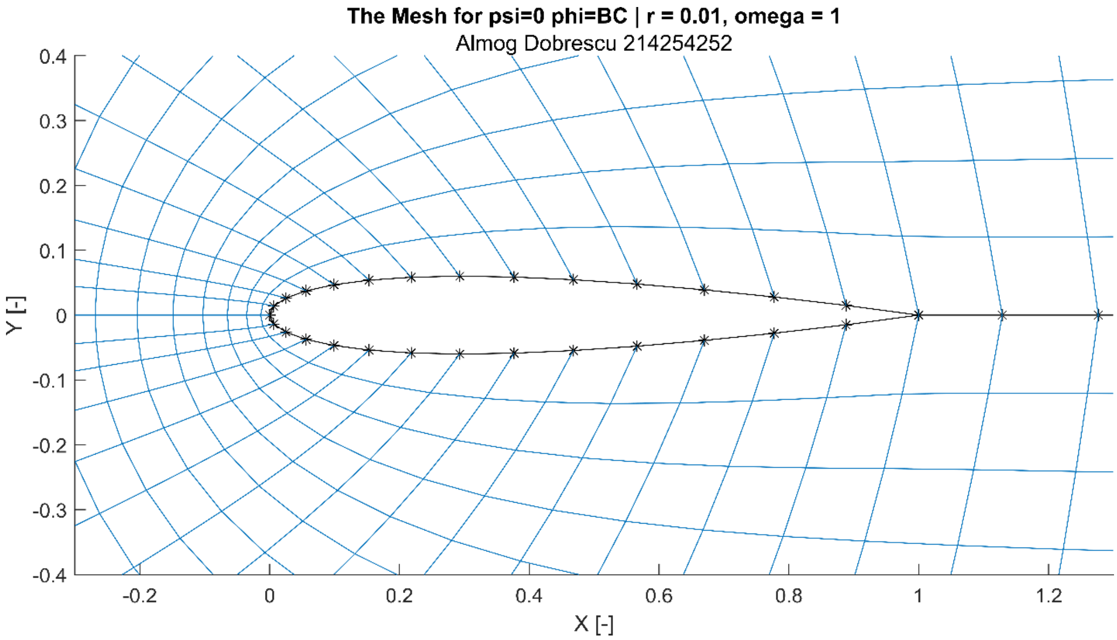

Figure 12: The mesh of $\psi = 0,\ \phi = BC$ - close up

We can see that there aren't a lot of points near the leading edge and a few point at the wake compared the $\phi = 0, \psi = BC$ case.

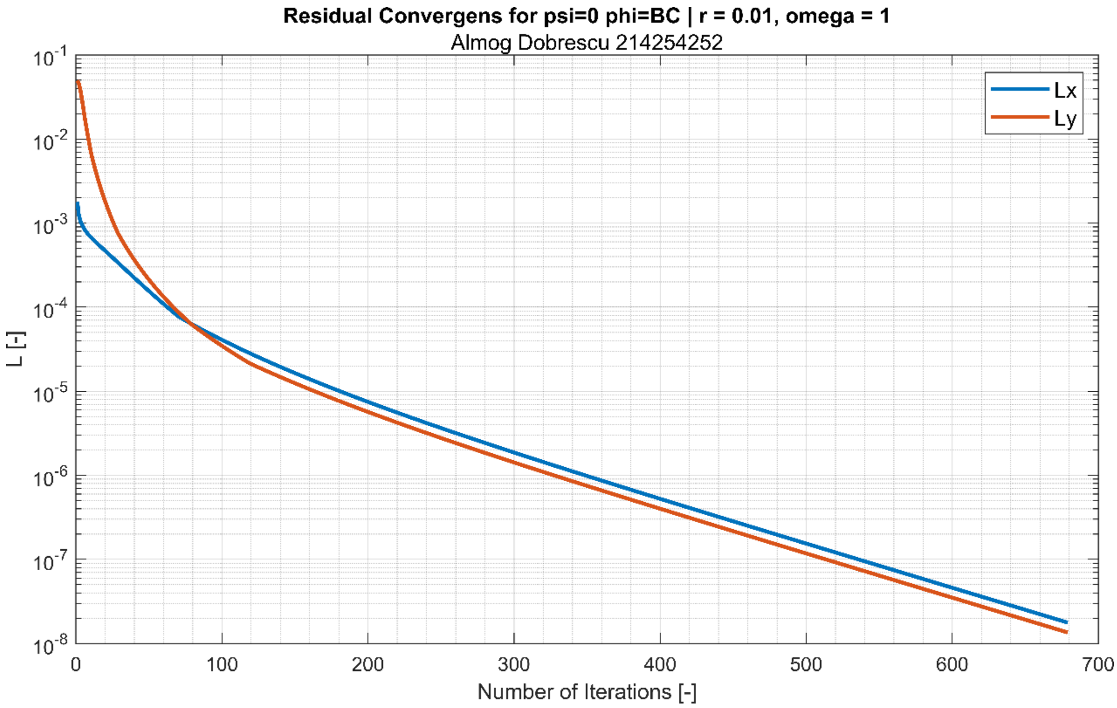

Figure 13: The residual convergens for $\psi = 0,\ \phi = BC$

We can see that after about 100 iterations the number of iterations decays exponential. Moreover, activating the control function in the η direction doubled the iteration number compared to the $\phi = 0, \psi = 0$ case.

ψ, ϕ is Calculated From the BC

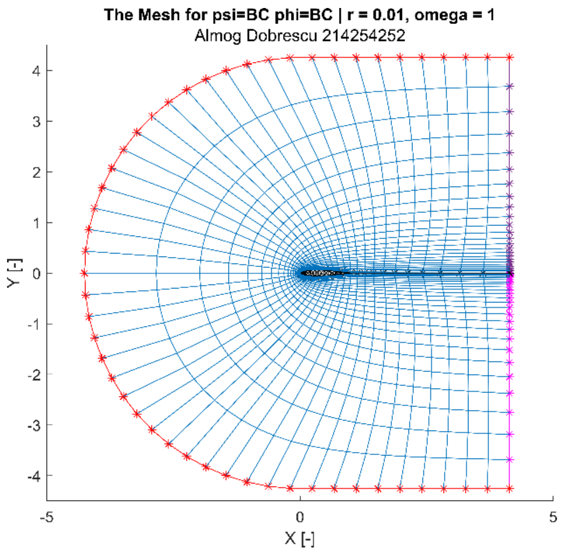

Figure 14: The mesh of $\psi = BC,\ \phi = BC$

When both control functions are active, we can see a combination of the affects described in figure 8 and 11. The lines are denser near the airfoil especially at the leading edge, and at the wake. The lines are spacious away from the center especially towards the free flow area.

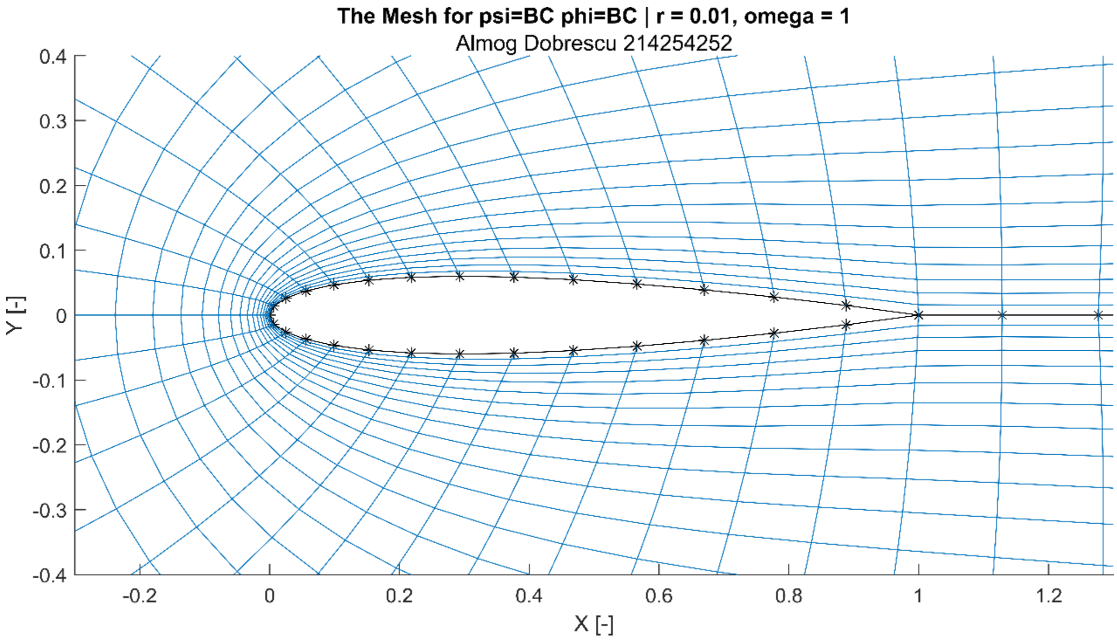

Figure 15: The residual convergens for $\psi = BC,\ \phi = BC$

We can see that there are a lot of points near the leading edge and the lines at the wake are perpendicular to the wake.

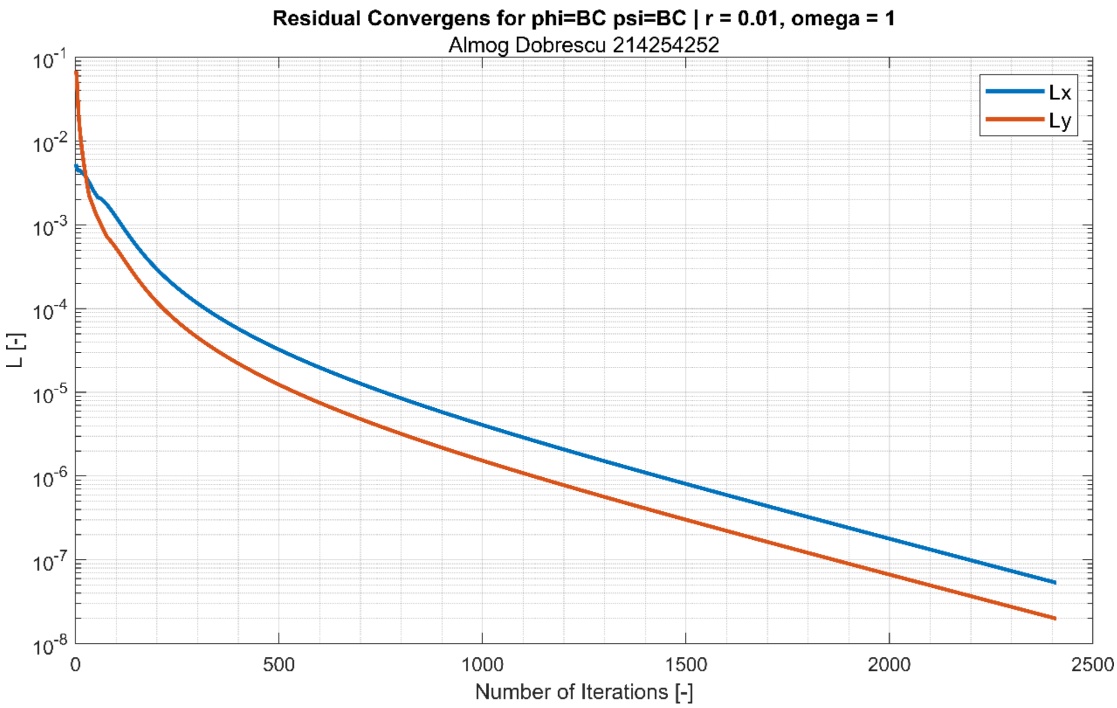

Figure 16: The residual convergens for $\psi = BC,\ \phi = BC$

We can see that after about 100 iterations the number of iterations decays exponential. Moreover, activating both control function increases the iteration number by six times compared to the $\phi = 0, \psi = 0$ case.

Influence of r and ω on Convergence

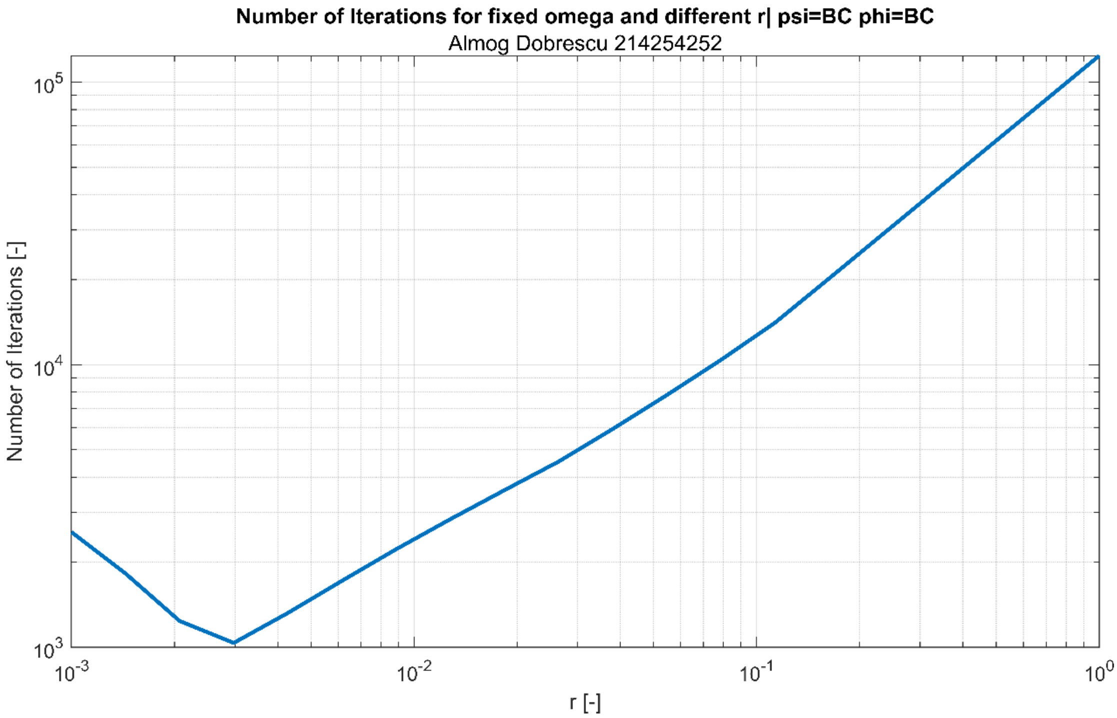

Figure 17: Number of iterations for fixed ω, different r

We can see clearly from the graph that there is an optimal value for the parameter r.

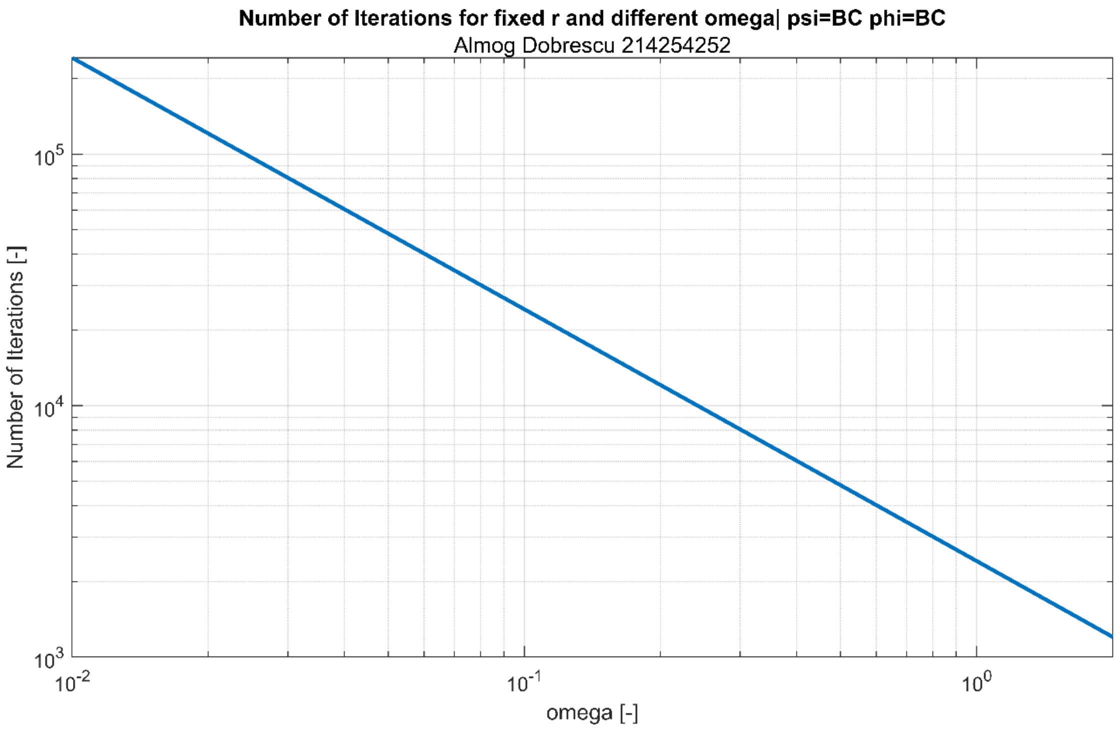

Figure 18: Number of iterations for fixed r, different ω

We can see that as ω increases, the number of iterations decays exponentially.

So, we would like to chose omega as close to 2 as possible.

Conclusion

The program run for 4 different combinations of $\phi\ and\ \psi$, checked the residual converges for each case and checked the effect of the converges parameters on the number of iterations. Here are some of the conclusions:

- The control function in the ξ direction has a big effect on the shape of the mesh and density of points around the leading edge and the airfoil wake.
- The control function in the η direction has a small effect on the shape of the mesh and doesn't change the points density around the airfoil wake.
- Activating the control functions increases the number of iterations until convergence.
- There is a clearly optimal value for the parameter r for which, the number of iterations is minimal.
- The bigger the ω the smaller the number of iterations, so the maximum value of ω is 2.