CFD Project-2

Abhijit Venkat (22110008) and Aditya Prasad (22110018)

Department of Mechanical Engineering, IIT Gandhinagar

Instructor: Prof. Dilip S. Sundaram

Contents

1. Introduction

The study of viscous flow over a flat plate is a classical problem in fluid mechanics, first addressed by Ludwig Prandtl in 1904. Prandtl introduced the concept of the boundary layer, which explains how fluid flow near a solid surface behaves differently from the free stream flow far from the surface. This distinction arises due to the effects of viscosity, which causes the fluid to "stick" to the surface, creating a velocity gradient in the region close to the wall.

For an incompressible, laminar flow past a flat plate, the governing equations reduce to a simplified form known as the **boundary layer equations**, which are derived from the Navier-Stokes equations. These equations assume that the Reynolds number of the flow is large enough so that the flow far from the plate remains unaffected by the presence of the plate, allowing us to focus on the thin region close to the surface where viscous effects dominate.

1.1. Governing Equations

The flow of interest is steady, laminar, and incompressible. The governing equations are derived from the laws of conservation of mass and momentum:

1. Continuity Equation: The conservation of mass for incompressible flow is expressed as:

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 \tag{1}$$

where u and v are the velocity components in the x and y directions, respectively.

2. Momentum Equation (x-direction): The conservation of momentum in the x-direction (accounting for the effect of viscosity) is given by:

$$u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} = \nu\frac{\partial^2 u}{\partial y^2} \tag{2}$$

where ν is the kinematic viscosity of the fluid.

1.2. Boundary Layer Concept

The boundary layer is a thin region near the plate where the velocity of the fluid increases from zero (due to the no-slip condition at the wall) to the free-stream velocity U_∞ far from the plate. As a result, most of the fluid's velocity change occurs within this thin region. The thickness of the boundary layer, denoted as $\delta(x)$, increases with distance along the plate. For a laminar boundary layer, this growth is proportional to $x^{1/2}$, where x is the distance from the leading edge of the plate.

The key challenge in solving the boundary layer equations numerically lies in accurately capturing the sharp velocity gradients that exist near the wall. In this project, we aim to solve the steady boundary layer equations using the **finite difference method** and compare the results with the well-known analytical solution provided by Blasius.

1.3. Boundary Conditions

The boundary conditions for the problem are as follows:

- At the wall ($y = 0$): The no-slip condition implies $u = 0$ and $v = 0$.
- Far from the wall ($y \to \infty$): The velocity approaches the free-stream velocity, i.e., $u = U_\infty$.

2. Analytical Solution

To simplify the equations, we introduce the similarity variable:

$$\eta = y\sqrt{\frac{U_\infty}{\nu x}} \tag{3}$$

This variable reduces the problem to a function of η only. The velocity in the x-direction is expressed as:

$$u = U_\infty f'(\eta) \tag{4}$$

and the y-direction velocity is given by:

$$v = \frac{\sqrt{\nu U_\infty}}{2\sqrt{x}}\left(\eta f'(\eta) - f(\eta)\right) \tag{5}$$

2.1. Blasius Equation

Substituting the similarity variable into the momentum equation, we obtain the Blasius equation:

$$f'''(\eta) + \frac{1}{2}f(\eta)f''(\eta) = 0 \tag{6}$$

2.2. Boundary Conditions and relation with Boundary Layer Thickness

The boundary conditions for the Blasius equation are:

$$f(0) = 0, \tag{7}$$
$$f'(0) = 0, \tag{8}$$
$$f'(\eta) \to 1 \quad \text{as} \quad \eta \to \infty \tag{9}$$

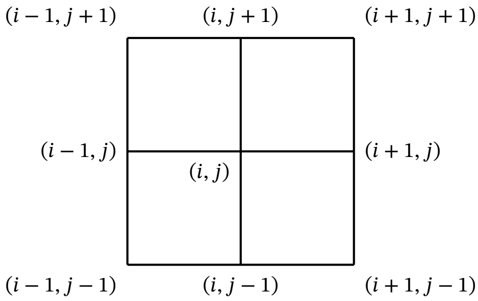
After solving the differential equation using the boundary conditions, we get the relation for the boundary layer thickness, which is defined as the distance from the plate where the velocity reaches 99% of the free-stream velocity U_∞. The relation for the boundary layer thickness, derived from the Blasius solution, is given by:

$$\delta(x) = \frac{4.91x}{\sqrt{Re_x}} \tag{10}$$

where $Re_x = \frac{U_\infty x}{\nu}$ is the Reynolds number based on the distance x from the leading edge.

3. Discretization and Schemes employed

The momentum equation for the flat plate boundary layer exhibits parabolic characteristics. This necessitates the use of a space marching scheme for its solution. Although the equations describe a steady-state condition, the x-direction behaves analogously to time in a time-marching scheme. When solving for a specific x position, the velocity components u and v at the adjacent downstream location are unknown. Consequently, while computing the solution at the (i, j) node, information from the $(i + 1, j)$ node is unavailable.



3.1. Euler Explicit method

The explicit method is a conditionally stable method, which means that we need to choose the discretization parameters carefully. This means that our solution may be stable or unstable (Blow up) depending on the values of Δx and Δy. We have chosen the total number of points in the x direction to be n_x and the total number of points in the y direction to be n_y. Therefore, $\Delta x = \frac{L}{n_x - 1}$ and $\Delta y = \frac{H}{n_y - 1}$.

$$(\frac{\partial u}{\partial y})_{i,j} \approx \frac{u_{i,j+1} - u_{i,j-1}}{2\Delta y} \tag{11}$$

$$(\frac{\partial^2 u}{\partial y^2})_{i,j} \approx \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{\Delta y^2} \tag{12}$$

$$(\frac{\partial u}{\partial x})_{i,j} \approx \frac{u_{i+1,j} - u_{i,j}}{\Delta x} \tag{13}$$

Now, when we substitute the discretized derivatives from equations 11, 12, and 13 into equation 2, we get the following:

$$u_{i+1,j} + v_{i,j}\frac{u_{i,j+1} - u_{i,j-1}}{2\Delta y} = \nu\frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{\Delta y^2} \tag{14}$$

Rearranging this equation so that we can isolate $u_{i+1,j}$ into the LHS:

$$u_{i+1,j} = u_{i,j} + \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{u_{i,j}}\frac{\nu\Delta x}{\Delta y^2} - \frac{u_{i,j+1} - u_{i,j-1}}{2}\frac{v_{i,j}\Delta x}{u_{i,j}\Delta y} \tag{15}$$

The only unknown in this equation is $u_{i+1,j}$, which has been written in terms of all the knowns. Therefore, we have explicitly calculated the value of $u_{i+1,j}$. Now, we need to make sure that the continuity equation is also satisfied. Basically, from the continuity equation, we can find out the value of $v_{i+1,j}$. The equation for that is given below.

$$\frac{\partial v}{\partial y} \approx \frac{v_{i+1,j} - v_{i+1,j}}{\Delta y} \tag{16}$$

After substituting equations 13 and 16 into equation 1:

$$v_{i+1,j} = v_{i+1,j-1} - \frac{\Delta y}{\Delta x}(u_{i+1,j} - u_{i,j}) \tag{17}$$

Equations 15 and 17 are the ones that are needed to be solved for each grid point. This would directly give us the u and v velocity values for each node.

3.2. Euler Implicit method

The Euler Implicit method is an unconditionally stable numerical method for solving the steady boundary layer flow problem. In this method, we solve for the unknowns at the current grid point (i.e., at $i + 1$) by forming a system of linear equations. The Gauss-Seidel method is used to iteratively solve this system for each column.

Discretized Derivatives

To discretize the momentum equation, we use the following finite difference approximations for the spatial derivatives:

1. The first derivative of u with respect to x:

$$(\frac{\partial u}{\partial x})_{i+1,j} \approx \frac{u_{i+1,j} - u_{i,j}}{\Delta x} \tag{18}$$

where $u_{i+1,j}$ is the unknown at the current step, and $u_{i,j}$ is the known value at the previous step.

2. The first derivative of u with respect to y:

$$(\frac{\partial u}{\partial y})_{i+1,j} \approx \frac{u_{i,j+1} - u_{i,j-1}}{2\Delta y} \tag{19}$$

where $u_{i,j+1}$ and $u_{i,j-1}$ are the known values at the previous step.

3. The second derivative of u with respect to y:

$$(\frac{\partial^2 u}{\partial y^2})_{i+1,j} \approx \frac{u_{i+1,j+1} - 2u_{i+1,j} + u_{i+1,j-1}}{\Delta y^2} \tag{20}$$

where $u_{i+1,j+1}$, $u_{i+1,j}$, and $u_{i+1,j-1}$ are unknowns at the current step.

Momentum Equation Discretization

The governing momentum equation for the boundary layer flow is:

$$u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} = \nu\frac{\partial^2 u}{\partial y^2} \tag{21}$$

Substituting the discretized derivatives into this equation gives:

$$u_{i,j}\frac{u_{i+1,j} - u_{i,j}}{\Delta x} + v_{i,j}\frac{u_{i+1,j+1} - u_{i+1,j-1}}{2\Delta y} = \nu\frac{u_{i+1,j+1} - 2u_{i+1,j} + u_{i+1,j-1}}{\Delta y^2} \tag{22}$$

Rearranging this equation to solve for all the (i+1) terms, we move all the $i + 1$ terms to the left-hand side (LHS) and the i terms to the right-hand side (RHS):

$$u_{i+1,j}\left(\frac{u_{i,j}}{\Delta x} + \frac{2\nu}{\Delta y^2}\right) + u_{i+1,j+1}\left(\frac{v_{i,j}}{2\Delta y} - \frac{\nu}{\Delta y^2}\right) - u_{i+1,j-1}\left(\frac{v_{i,j}}{2\Delta y} + \frac{\nu}{\Delta y^2}\right) = \frac{u_{i,j}^2}{\Delta x}$$

We get an implicit relation between u_{i+1} terms. We solved this using an iterative approach using the Gauss-Seidel scheme.

Continuity Equation Discretization

The continuity equation is used to update the vertical velocity $v_{i+1,j}$. The discretized form of the continuity equation is:

$$\frac{\partial v}{\partial y} \approx \frac{v_{i+1,j} - v_{i+1,j-1}}{\Delta y} \tag{23}$$

Substituting this into the continuity equation gives:

$$v_{i+1,j} = v_{i+1,j-1} - \frac{\Delta y}{\Delta x}\left(u_{i+1,j} - u_{i,j}\right) \tag{24}$$

Detailed Explanation of Gauss Seidel iterative solver:

In the discretized X-momentum equation, we have a system of linear equations for a particular column of $(i + 1)$. To solve this system iteratively using the Gauss-Seidel method, we introduce a new notation that includes the iteration number as a superscript. This helps distinguish between the column index (which remains constant during the solution process for a particular column) and the iteration number (which changes as we refine our solution).

The Gauss-Seidel method involves the following steps:

1. Initialize the solution:

$$u_{i+1,j}^{(0)} = \text{initial guess for all } j \tag{25}$$

2. For each iteration $n = 0, 1, 2, ...$, update the solution:

$$u_{i+1,j}^{(n+1)} = \frac{\frac{(u_{i,j}^{(n)})^2}{\Delta x} + u_{i+1,j+1}^{(n)}(\frac{v_{i,j}^{(n)}}{(2\Delta y)} - \frac{\nu}{\Delta y^2}) - u_{i+1,j-1}^{(n)}(\frac{v_{i,j}^{(n)}}{(2\Delta y)} + \frac{\nu}{\Delta y^2})}{\frac{u_{i,j}^{(n)}}{\Delta x} + \frac{2\nu}{\Delta y^2}} \tag{26}$$

3. Check for convergence:

$$\max_j |u_{i+1,j}^{(n+1)} - u_{i+1,j}^{(n)}| < \varepsilon \tag{27}$$

where ε is a predefined tolerance.

Explanation of the above pseudo-code of Gauss Seidel

- In step 1, we start with an initial guess for $u_{i+1,j}$ at all j positions. This could be zero, or a value from a previous time step or iteration.
- In step 2, we update our solution. The superscript $(n+1)$ represents the new value we're calculating, while (n) represents values from the previous iteration. We solve for $u_{i+1,j}^{(n+1)}$ by isolating it on the left side of the equation and using the known values from the previous iteration on the right side.
- In step 3, we check for convergence. We calculate the maximum absolute difference between the new values $u_{i+1,j}^{(n+1)}$ and the previous values $u_{i+1,j}^{(n)}$ across all j positions.
- The tolerance ε is a small positive number (e.g., 10^{-6}) that defines our convergence criterion. When the maximum change in our solution from one iteration to the next is smaller than this tolerance, we consider our solution to have converged.
- If the convergence criterion is not met, we repeat steps 2 and 3, using the newly calculated values as the starting point for the next iteration.
- This process continues until either the solution converges (i.e., the change between iterations is smaller than our tolerance), or we reach a maximum number of iterations (to prevent infinite loops in case of non-convergence).

The Gauss-Seidel method is effective because it uses the most recently calculated values within each iteration. This often leads to faster convergence compared to methods that only use values from the previous iteration.

3.3. Crank-Nicolson method

We use the Crank-Nicolson scheme to discretize out governing equations. We evaluate our governing PDEs at (x= i+1/2,y= j) instant. The Crank-Nicolson scheme allows us to have second-order truncation errors in both the x and y domains along with unconditional stability.

We discretize our x-momentum equation as

$$u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} = \nu\frac{\partial^2 u}{\partial y^2} \tag{28}$$

We evaluate the above PDE at x= i+1/2; y= j instant.
We have discretized the first term using the central difference:

$$(u\frac{\partial u}{\partial x})_{i+1/2,j} \approx u_{i+1/2,j}\frac{u_{i+1,j} - u_{i,j}}{\Delta x} \tag{29}$$

We have approximated the $u_{i+1/2,j}$ by $u_{i,j}$ to simplify our scheme.

$$(u\frac{\partial u}{\partial x})_{i+1/2,j} \approx u_{i,j}\frac{u_{i+1,j} - u_{i,j}}{\Delta x} \tag{30}$$

We have discretized the second term as:

$$(v\frac{\partial u}{\partial y})_{i+1/2,j} = v_{i+1/2,j}\frac{(\frac{\partial u}{\partial y})_{i+1,j} + (\frac{\partial u}{\partial y})_{i,j}}{2} \tag{31}$$

We have approximated the $v_{i+1/2,j}$ by $v_{i,j}$ to simplify our scheme.

$$(v\frac{\partial u}{\partial y})_{i+1/2,j} \approx v_{i,j}\frac{(\frac{\partial u}{\partial y})_{i+1,j} + (\frac{\partial u}{\partial y})_{i,j}}{2} \tag{32}$$

We have the third term, as given below. We use the trapezoidal approximation to approximate the second-order differential term.

$$(\nu\frac{\partial^2 u}{\partial y^2})_{i+1/2,j} = \nu\frac{(\frac{\partial^2 u}{\partial y^2})_{i+1,j} + (\frac{\partial^2 u}{\partial y^2})_{i,j}}{2} \tag{33}$$

$$(\nu\frac{\partial^2 u}{\partial y^2})_{i+1/2,j} = \nu\frac{\frac{u_{i+1,j+1} - 2*u_{i+1,j} + u_{i+1,j-1}}{\Delta y^2} + \frac{u_{i,j+1} - 2*u_{i,j} + u_{i,j-1}}{\Delta y^2}}{2} \tag{34}$$

Upon combining all three discretized terms of the X-momentum together. We get an implicit relation as stated below:

$$LHS = u_{i,j}\frac{u_{i+1,j} - u_{i,j}}{\Delta x} + \frac{v_{i,j}}{4\Delta y}(u_{i+1,j+1} - u_{i+1,j-1} + u_{i,j+1} - u_{i,j-1}) \tag{35}$$

$$RHS = \frac{\nu}{2\Delta y^2}(u_{i+1,j+1} - 2u_{i+1,j} + u_{i+1,j-1} + u_{i,j+1} - 2u_{i,j} + u_{i,j-1}) \tag{36}$$

Rearranging this equation to solve for all the (i+1) terms, we move all the (i+1) to the LHS and keep all the other terms on the RHS. We get an implicit relation:

$$LHS = u_{i+1,j+1}(\frac{v_{i,j}}{4\Delta y} - \frac{\nu}{2\Delta y^2}) + u_{i+1,j}(\frac{u_{i,j}}{\Delta x}\frac{\nu}{\Delta y^2}) + u_{i+1,j-1}(-\frac{v_{i,j}}{4\Delta y} - \frac{\nu}{2\Delta y^2}) \tag{37}$$

$$RHS = \frac{u_{i,j}^2}{\Delta x} - \frac{v_{i,j}}{4\Delta y}(u_{i,j+1} - u_{i,j-1}) + \frac{\nu}{2\Delta y^2}(u_{i,j+1} - 2u_{i,j} + u_{i,j-1}) \tag{38}$$

We solve for these (i+1,j+1), (i+1,j), (i+1,j-1) terms using an iterative approach using the Gauss-Seidel solver, similar to the implicit case.

Now, we need to make sure that the continuity equation is also satisfied. Basically, from the continuity equation, we can find out the value of $v_{i+1,j}$. The equation for that is given below.

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 \tag{39}$$

We evaluate the continuity equation at (i+1/2, j) instant.
We discretized the first term using central difference about i+1/2 as:

$$\frac{\partial u}{\partial x}_{i+1/2,j} = \frac{u_{i+1,j} - u_{i,j}}{\Delta x} \tag{40}$$

We discretized the second term using forward difference as:

$$\frac{\partial v}{\partial y}_{i+1/2,j} = \frac{v_{i+1/2,j} - v_{i+1/2,j-1}}{\Delta y} \tag{41}$$

We replace the $v_{i+1/2,j}$ and $v_{i+1/2,j-1}$ by $v_{i+1,j}$ and $v_{i+1,j-1}$ terms respectively, to simplify our scheme.

$$\frac{\partial v}{\partial y}_{i+1/2,j} = \frac{v_{i+1,j} - v_{i+1,j-1}}{\Delta y} \tag{42}$$

Upon substituting both terms into our discretized continuity equation we get an explicit expression for $v_{i+1,j}$ as:

$$v_{i+1,j} = v_{i+1,j-1} - \frac{\Delta y}{\Delta x}(u_{i+1,j} - u_{i,j}) \tag{43}$$

Hence, equations 38 and 43 would directly give us the u and v velocity values for each node.

4. Results and Discussions

For consistency, we shall take the basic parameters to be the same between all the three methods employed. The length of the flat plate is taken to be $L = 1m$. The height of the domain is $H = 0.2m$. Apart from these values, it is mentioned that the Reynold's number is $Re_L = 10^4$, and the free stream velocity is taken to be $U_{inf} = 1\frac{m}{s}$. From this, the kinematic viscosity is defined as $\nu = \frac{U_{inf}L}{Re_L} = 10^{-4}\frac{m^2}{s}$. Additionally, as discussed in the section on Analytical solution

$$\delta(x) = \frac{4.91x}{\sqrt{Re_x}}$$

For $x = L = 1m$: $\delta(L) = 0.0491m$
We can use this value to compare our results for the boundary layer thickness.

4.1. Explicit Method

In this section, we shall discuss the different graphs and trends that are seen for the solution obtained using the Euler Explicit method. We know that the Euler Explicit scheme is conditionally stable. This scheme is first-order accurate in x, and second-order accurate in y. Therefore, the values of n_x and n_y must be chosen carefully. The condition for stability is given by:

$$\Delta x < \frac{u_{i,j}\Delta y^2}{2\nu}$$

In order to ensure that we follow this criterion, we choose $n_x = 10000$ points and $n_y = 300$ points. This gives us $\Delta x = L/(n_x - 1)$ and $\Delta y = H/(n_y - 1)$. After solving for the u and v velocities at each node of the discretized domain, we get the following contour plots for the velocity fields:

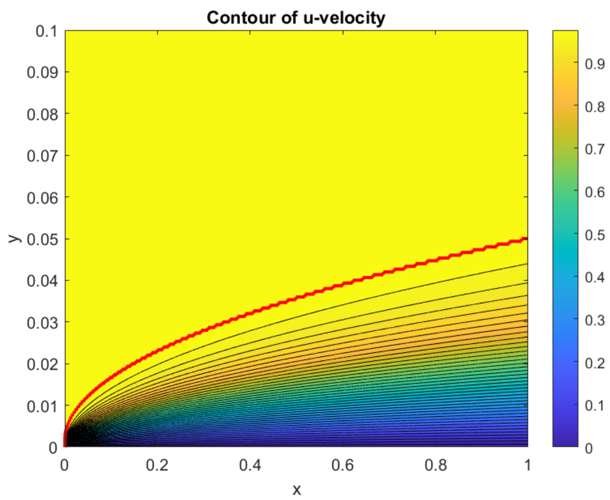


Figure 1. Contour plot of the u velocity field for Explicit method

The velocity field depicted in the contour plot captures the expected behavior of the flow. The dark blue region represents areas where the x-velocity is very low, while the lighter contours, transitioning to yellow, indicate increasing u-velocity magnitudes. As the contours approach the free-stream region, the u-velocity approaches the free-stream velocity, U_∞.

This contour plot also clearly illustrates the shape of the boundary layer. The boundary layer thickness, δ, is defined as the distance from the plate where the u-velocity magnitude reaches $0.99 \times U_\infty$. For each point along the x-axis, we extract and store the corresponding y-coordinate from the solution field, where this condition is satisfied. The red line visible in the plot represents the boundary layer obtained from the numerical solution.

As discussed earlier, the boundary layer thickness at $x = L$ is expected to be 0.0491 m. From the contour plot, it is evident that the numerical results closely match this theoretical value.

In the contour plot below, there are very few visible contours. This is because the v velocity is almost zero everywhere. This is to be expected since the flow is majorly in the x direction. Initially, towards the left boundary, we can see a slight variation in velocity. This might be due to the instantaneous appearance of the flat plate, which might then dissipate over the entire field and eventually become a very small value. However, the gradient of this velocity field does matter to us, as it is responsible for satisfying the continuity equation. No matter how small the y velocity is at all points, we cannot comment on the gradient of velocity.

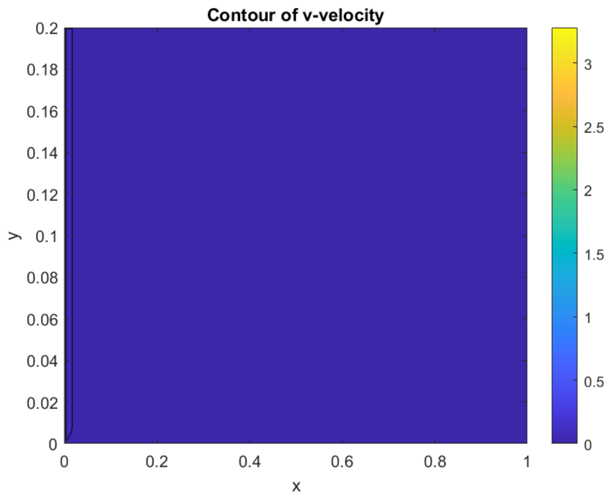


Figure 2. Contour plot of the y velocity field for Explicit method

The plot of the normalized x-velocity, $F'(\eta)$, versus the similarity variable, η, is a key result in boundary layer theory. It provides a universal velocity profile for laminar flow over a flat plate, based on the solution to the Blasius equation.

The similarity variable η is defined as:

$$\eta = y\sqrt{\frac{U_\infty}{\nu x}}$$

which combines the physical coordinates x and y, allowing the velocity profiles at different x-positions to collapse into a single curve.

In the plot below:
- The x-axis represents η, a scaled distance from the plate.
- The y-axis represents $F'(\eta)$, which is the normalized x-velocity, defined as:

$$F'(\eta) = \frac{u(x, y)}{U_\infty}$$

where $u(x, y)$ is the x-velocity at a given point, and U_∞ is the free-stream velocity.

At $\eta = 0$, $F'(0) = 0$, reflecting the no-slip condition at the wall. As η increases, $F'(\eta)$ increases, representing how the x-velocity grows away from the wall until it reaches 99% of the free-stream velocity U_∞, typically around $\eta \approx 5$. This point marks the edge of the boundary layer. Beyond this point, the velocity asymptotically approaches U_∞, with $F'(\eta) \approx 1$ in the free-stream region.

The plot captures how the boundary layer develops along the plate, with the velocity increasing from zero at the plate to the free-stream velocity outside the boundary layer. The rapid increase near the plate and the flattening of the curve at higher η values indicate the transition from the viscous boundary layer to the inviscid free-stream flow. The thickness of the boundary layer can be inferred from the region where $F'(\eta)$ approaches 1.

We can see that the numerical solution is overlapping for almost the entire plot. We have plotted only till $\eta = 6$ for the numerical solution to show the trend. This indicates that the numerical solution is very close to the Analytical solution.

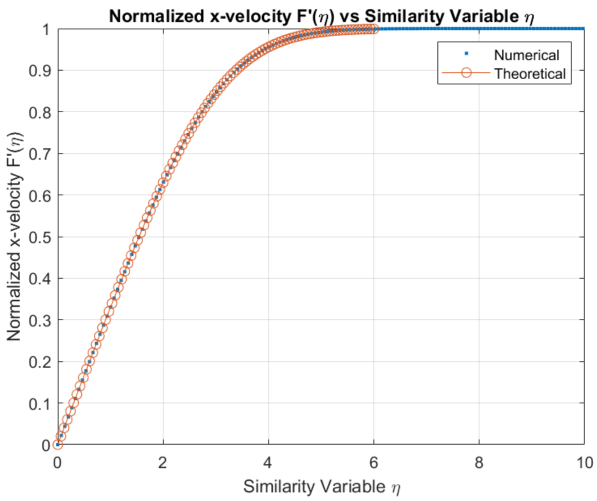


Figure 3. $f'(\eta)$ vs η for Explicit method

In the final graph for the explicit method, we plot the variation of boundary layer thickness (δ) as a function of x. Both the Analytical and the numerical thickness trends have been plotted. The dotted line shows the numerically obtained boundary layer function, while the solid line represents the analytical boundary layer function. Clearly, the solution matches the given parameters to a very good extent. This is the same dotted line we plotted in the u velocity field contour to help identify where the boundary layer is separating.

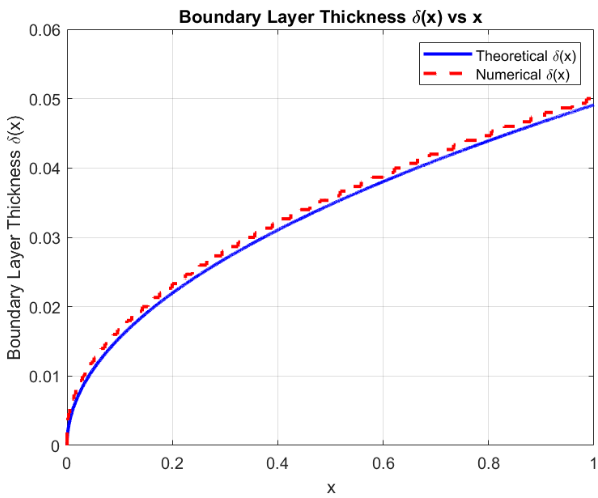


Figure 4. δ (boundary layer thickness) varying with x for Explicit method

4.2. Implicit Method

Now, we shall discuss the trends observed while employing the Euler implicit method. Since this method is unconditionally stable, we have chosen different values of n_x and n_y. The more refined the grid is, the more accurate our results should be. This method is also first-order accurate in x and second-order accurate in y. For our simulation, we have chosen $n_x = 3000$ and also $n_y = 3000$. This is a major improvement in meshing as compared to the Euler explicit method, where we could only take n_y to be about 300 while ensuring that n_x is also within our computational limits. Here, we had to use the Gauss-Seidel iterative approach to solve the equations for each column. This simulation takes slightly more time than the Euler explicit method, but since it is unconditionally stable, we should still get better results than the Explicit method.

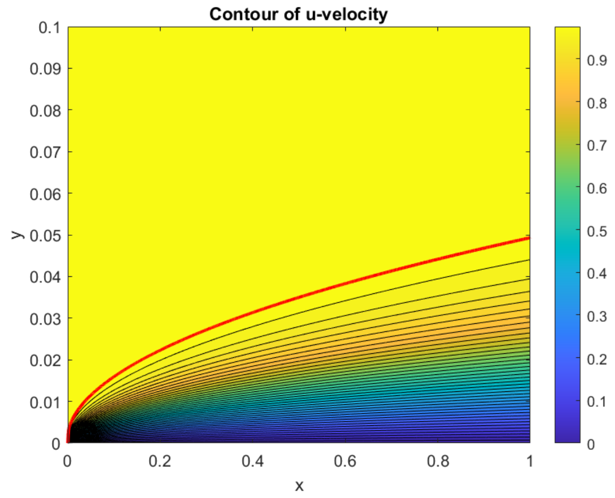


Figure 5. Contour plot of the x velocity field for Implicit method

The first contour plot represents the u-velocity field, which appears quite similar to the results obtained from the explicit scheme. One noticeable difference is that the boundary layer seems thinner in this case. However, this is not due to an actual change in the boundary layer thickness. Rather, it occurs because the plot does not have enough contour levels to capture the detailed variation in the velocity field. If we were to add more contours, the region near the plate would become entirely black, causing a loss of detail in that area.

Similar to the explicit case, we have plotted the boundary layer in this contour as well, using the same definition: the boundary where $u = 0.99 \times U_\infty$. From the plot, we observe that, as expected, the boundary layer thickness at $x = L$ approaches the theoretical value of 0.0491 m.

Moving on to Figure 6, the v velocity field is quite different from the one we obtained using the Explicit method. We can see a few contours here, but most of them are of really small values of v. This is still consistent with the fact that the v would be very small for all points in the domain. Similar to the Explicit contour, here also, in the beginning, there seems to be some velocity present. This can be treated as the error that comes with this scheme.

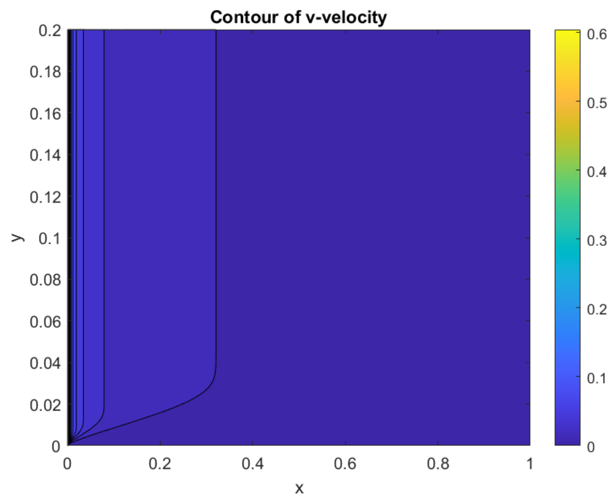


Figure 6. Contour plot of the y velocity field for Implicit method

Looking at the plot for Normalized x velocity vs the similarity variable η, we can see that the Numerical solution and the Theoretical solution overlap completely. Visually, one cannot determine any difference between the two. We would have to zoom the graph quite a lot to even see the different curves present in the plot.

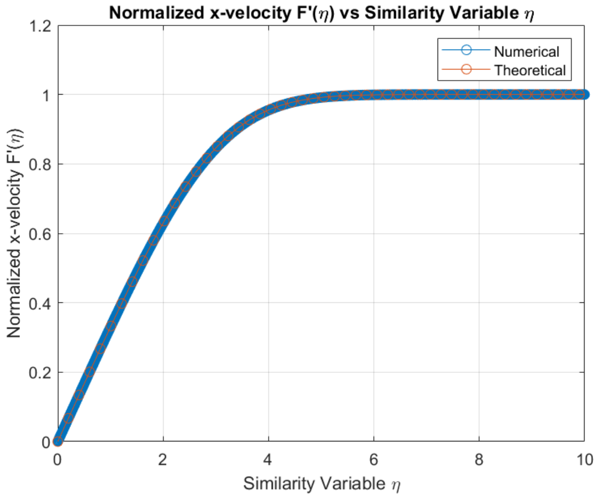


Figure 7. $f'(\eta)$ vs η for Implicit method

Finally, we come to the last plot for this method, which describes the variation of boundary layer thickness δ with x. Clearly, from the plot, the numerical and theoretical graphs are for all practical purposes, coinciding. The error for this graph is also minimal. The only part where we seem to be getting some results which are not entirely accurate is the y velocity field.

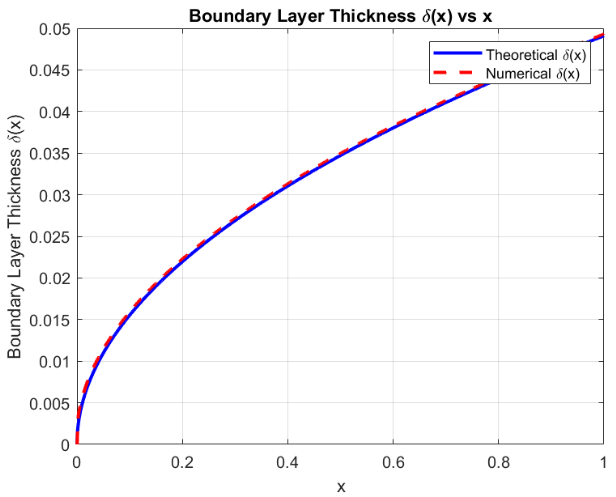


Figure 8. δ (boundary layer thickness) varying with x for Implicit method

4.3. Crank-Nicolson Method

We now arrive at our final method, which is the Crank-Nicolson method. This method can be thought of as a mix of the implicit and Explicit methods. This is because we take the average of the derivatives at the intermediate point between two nodes. This method gives us something that can be solved iteratively but with second-order accuracy in both x and y. For our simulation case, we take the number of points to be the same as in the implicit case. $n_x = 3000$ points and $n_y = 3000$ points. Therefore, before we even begin the analysis, we must be able to predict that this method is much more accurate than both the above-discussed methods.

Clearly, when we look at our first contour plot of the x velocity, we can see that it resembles the original two plots. However, similar to the case of the implicit contour plot for x velocity, the number of contours must be increased in order to capture the boundary layer completely. However, this would lead to a loss of visual information on the plot near the plate. Here, as well, we have marked the boundary layer thickness curve, which is defined as $0.99 \times U_{inf}$. The thickness, in this case, also tends towards the theoretical value of 0.0491m.

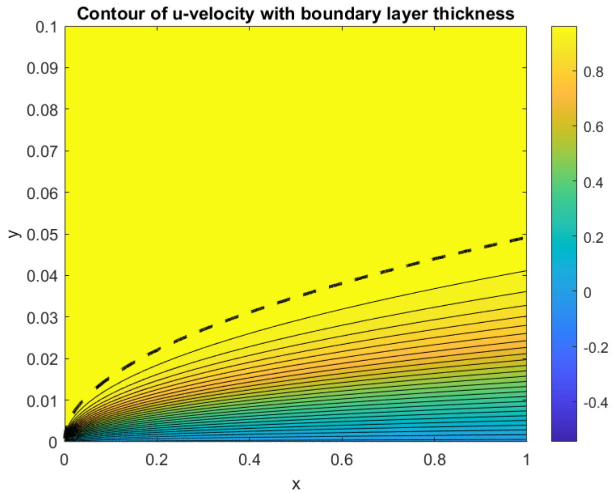


Figure 9. Contour plot of the x velocity field for Crank-Nicolson method

Our next plot is the y velocity contour or the $v(x, y)$ velocity field. This plot is quite different from both the methods discussed above. While, again, most of the region has v tending to 0, there are a few regions where the $v(x, y)$ becomes negative. There is no proper reason for this type of error. We do not know if it is even an error or not.

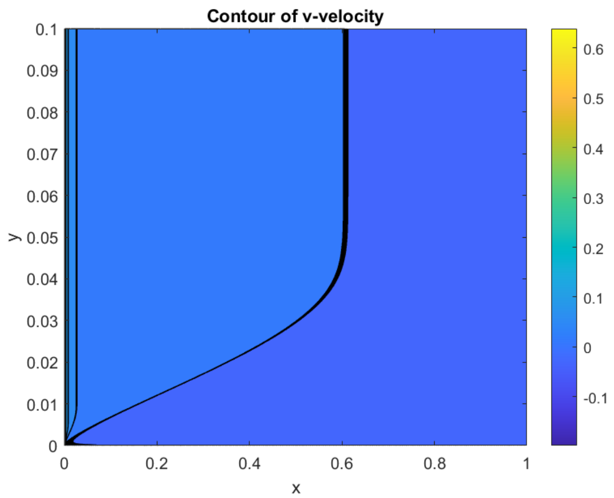


Figure 10. Contour plot of the y velocity field for Crank-Nicolson method

Coming to figures 11 and 12, the intuition of the graphs and what they represent is obviously the same for all methods. In this case, for the normalized x-velocity vs η curve, the theoretical and numerical curves completely overlap each other. The naked eye cannot distinguish between the two curves, and we need to zoom a lot into the curve to be able to see both of them distinctly.

The same goes for the boundary layer thickness vs x plot. The theoretical and numerical curves are barely distinguishable.

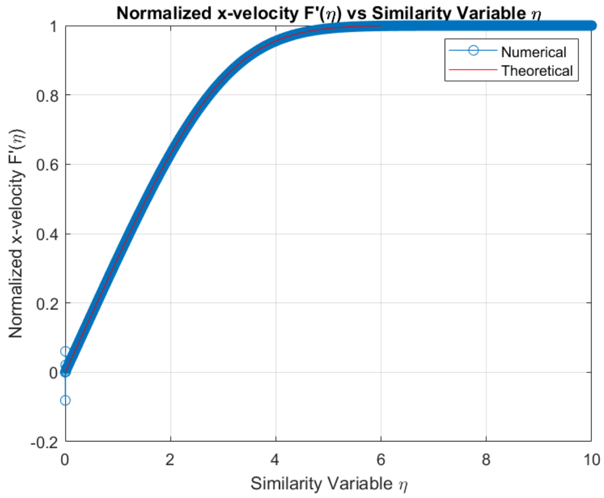


Figure 11. $f'(\eta)$ vs η for Crank-Nicolson method

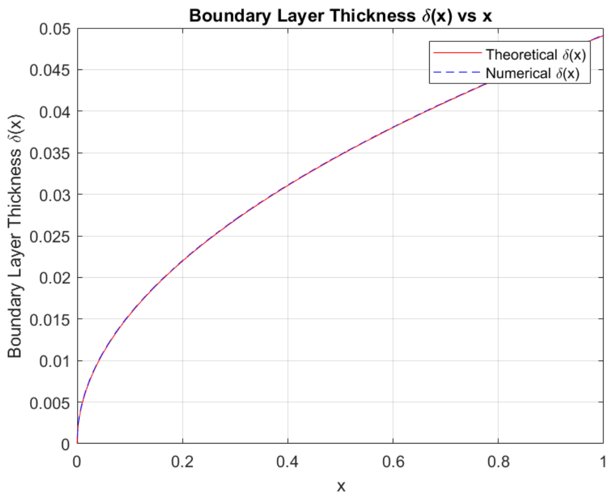


Figure 12. δ (boundary layer thickness) varying with x for Crank-Nicolson method

One interesting point to note is that the Crank-Nicolson method took the most amount of compute time to produce results. This is because it is an iterative method and is second-order accurate in both x and y simultaneously, while the others are first-order accurate in x and second-order accurate in y. Naturally, it is expected to take more time than the Euler implicit method to converge. The Euler Explicit method took the least time because it is not based on an iterative approach.

5. Acknowledgements

We would like to express our sincere gratitude to Professor Dilip Srinivas Sundaram for providing us with the opportunity to work on this project. We would also like to thank all the Teaching assistants for their invaluable help in sorting out all the bottlenecks throughout this project. Their dedication and expertise were crucial to the success of this project.