Numerical Methods of Thermo-Fluid Dynamics I

Winter Semester 2017-2018

DELIVERABLE TASK II:

Numerical Solution of Boundary Layer Equation

Report by:

Name	:	Jishnu Jayaraj	Number	:	22448952
Name	:	Mohammad Moataz	Number	:	22455424

Chair of Fluid Mechanics

Department of Biochemical Engineering, Technical Faculty Friedrich-Alexander University Erlangen-Nuremberg

Contents

List of Figures

Introduction

We are given a flat plate over which a parallel laminar flow of fluid passes. The x coordinate is measured along the plate surface from the leading edge of the plate in the direction of flow, and y is measured from the surface in the normal direction.

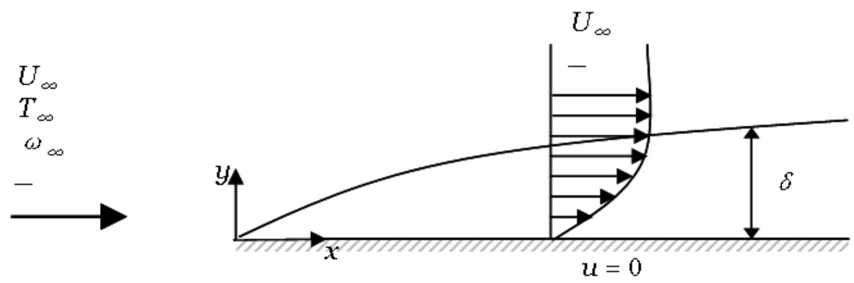

Figure 1: Laminar boundary layer over a flat plate

The dimensionless continuity and momentum equation for the given flow are

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

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

where Re is the Reynolds number, defined as $Re = u_\infty L/\nu$ where ν is the kinematic viscosity, L is the plate length and u_∞ is the free stream velocity of the outer flow. The velocity components u and v are dimensionless and are defined as $u = {}^{\bar{u}}/_{u_\infty}$ and $v = {}^{\bar{v}}/_{u_\infty}$ where $\bar{u}$ and $\bar{v}$ are the dimensional velocities. Lengths have been made dimensionless by using the plate length, $x = {}^{\bar{x}}/_{L}$

Boundary conditions are given by

$y = 0$ $\quad$ $u = v = 0$ $\quad$ no slip condition

$y = \infty$ $\quad$ $u \to 1$ $(\bar{u} \to u_\infty)$ $\quad$ free outer flow

1. Discretizing of Continuity Equation

For discretising $\frac{\partial u}{\partial x}$ we use the same scheme given in the question, and for discretising $\frac{\partial v}{\partial x}$ we use backward difference scheme to make use of the fact that v is known upstream starting from the given boundary conditions.

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

$$\frac{1}{2\Delta x}\left(u_{i,j} - u_{i-1,j} + u_{i,j-1} - u_{i-1,j-1}\right) + \frac{1}{\Delta y}\left(v_{i,j} - v_{i,j-1}\right) = 0 + O(\Delta x) + O(\Delta y)^2$$

Rearranging for $v_{i,j}$

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

This equation is first order accurate in Δx and second order accurate in Δy.

2. Discretizing Momentum Equation

$$u\frac{\partial u}{\partial x} + v\frac{\partial v}{\partial y} = \frac{1}{Re}\frac{\partial^2 u}{\partial y^2}$$

Because explicit methods have restrictive stability constraints we preferred to use an implicit method as follows

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

This equation is first order accurate in Δx and second order accurate in Δy.

The advantage of implicit methods is that they are unconditionally stable. However, this results in a nonlinear difference equation as it is clear from the first term $\frac{u_{i,j}}{\Delta x}\left(u_{i,j} - u_{i-1,j}\right)$ where we have ${u^2}_{i,j}$ and the second term $\frac{v_{i,j}}{2\Delta y}\left(u_{i,j+1} - u_{i,j-1}\right)$ where we have $v_{i,j}u_{i,j+1}$ and $v_{i,j}u_{i,j-1}$.

The simplest and most common strategy is to linearize that difference equations is by evaluating all coefficients at $i-1$. This is known as "Lagging" the coefficients, accordingly the equation becomes

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

Rearranging for $u_{i,j}$

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

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

3. MATLAB Code

The MATLAB code is written considering $\Delta x=\Delta y=h=0.0005$.

The horizontal velocity profile was uniform ($^{u}/_{u_\infty}=1$) just before hitting the edge of the plate. So, we can assume that just at the flat plate tip $u=0$ at $y=0$, and equals to 1 at all other nodes ($y>0$).

Also since the maximum boundary layer thickness ($\delta_{max}=\frac{5}{\sqrt{Re}}$) is less than the height of our study domain ($y=2\delta_{max}$), it can be assumed that $u=1$ at $y=2\delta$.

First the momentum equation will be used to solve for u where each iteration will solve for one particular column (implicitly) before it moves to the next one. After calculating u for a specific column then v can be calculated directly from continuity equation (explicitly).

For the code please check the appendix.

4. Numerical Solution for the Whole Domain

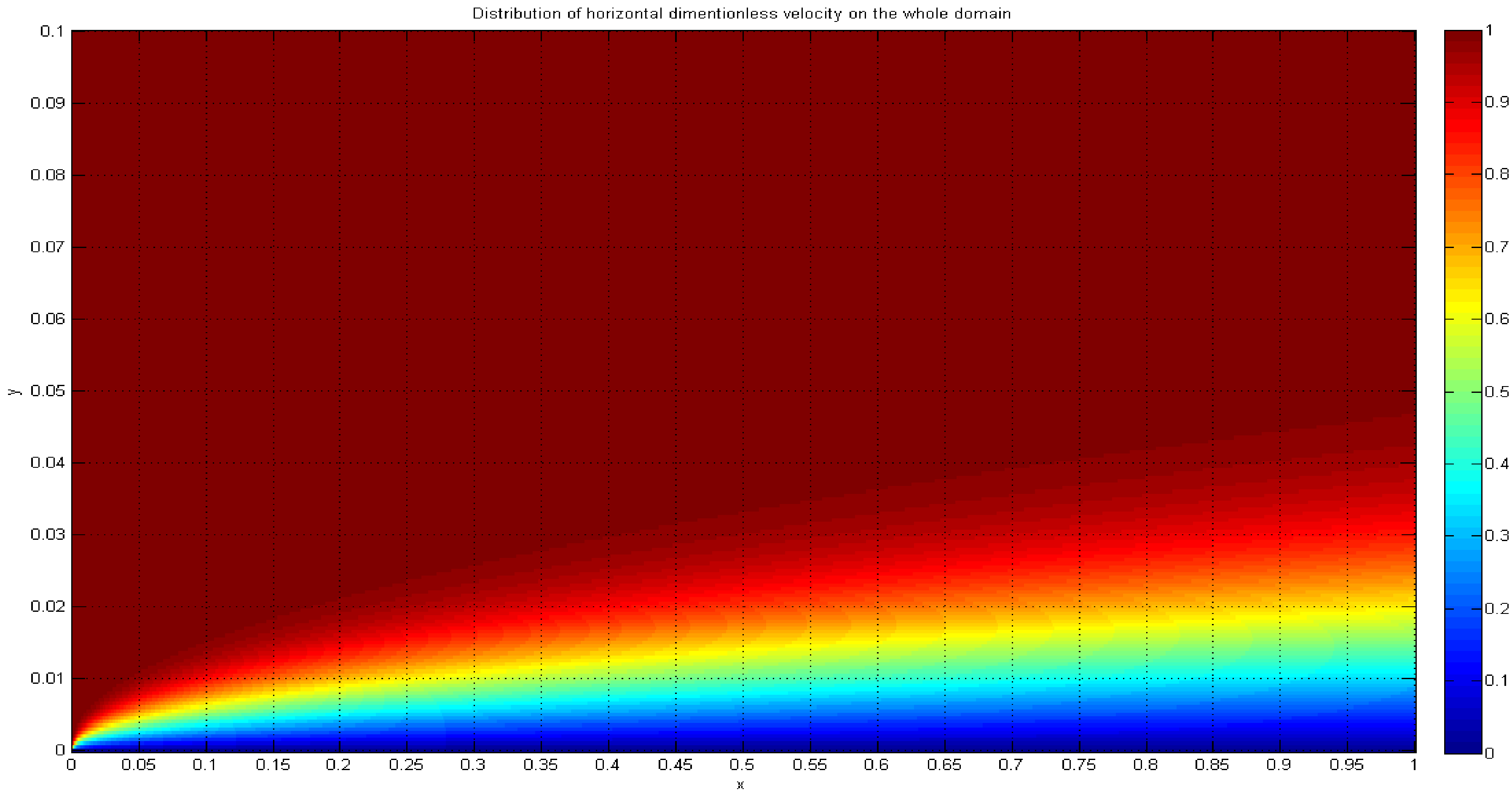

Figure 2: Contour of u over whole domain

The horizontal velocity of the particles in the first fluid layer adjacent to the plate becomes zero because of the no slip condition. This stagnant layer slows down the particle of neighboring fluid layers as a result of friction between particles of these two adjoining fluid layers at different velocities. This fluid layer slows down the particles of next layer and so on. Thus the presence of the plate is felt up to some normal distance δ (called the boundary layer thickness) from the plate beyond which the free stream velocity remains unchanged.

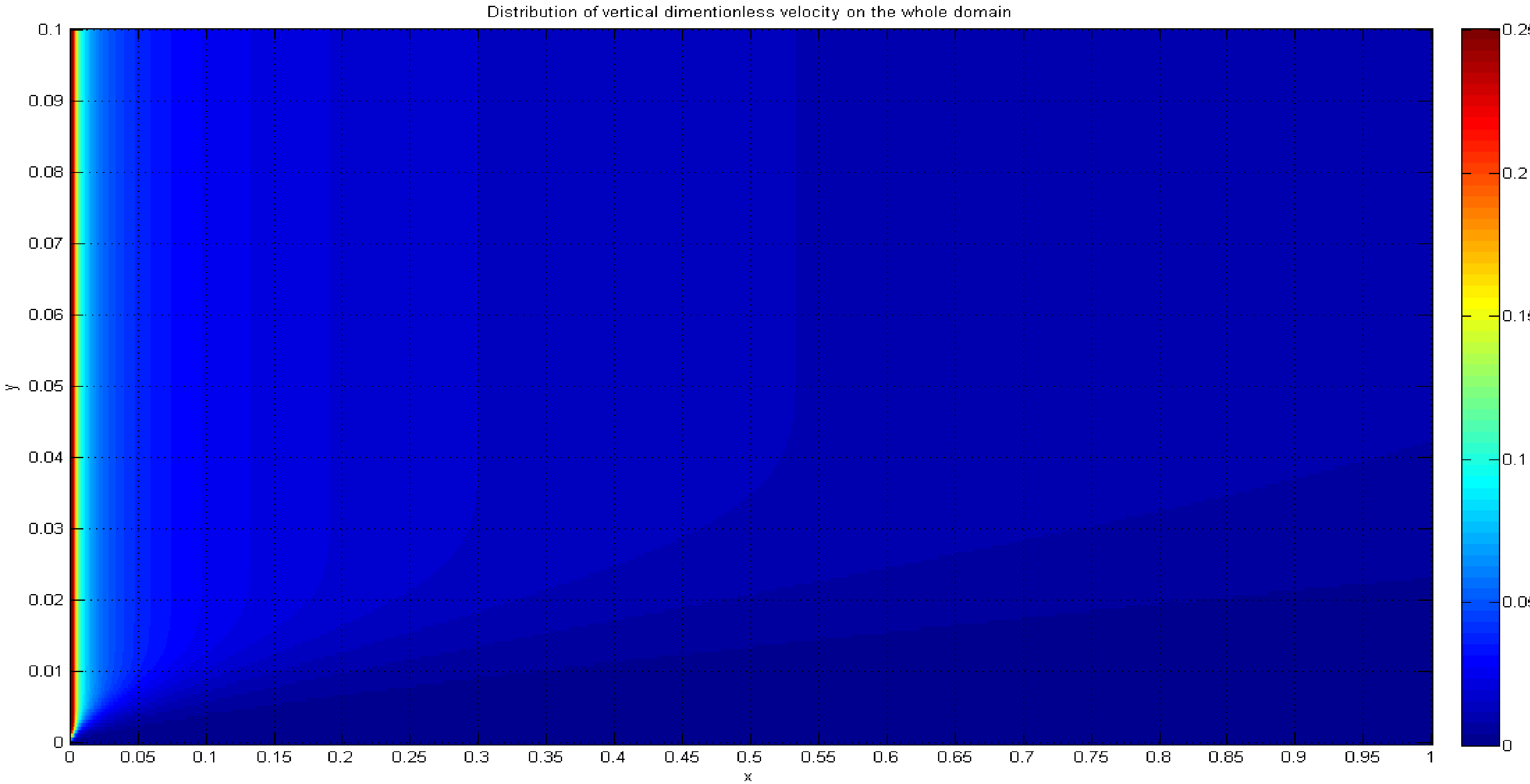

Figure 3: Contour of v over whole domain

The free stream velocity before hitting the plate is uniform and constant. But once the flow hits the tip of plate there is a drastic change in u (horizontal velocity component) with respect to direction of flow due to the no slip condition which makes the first fluid layer suddenly changes from u_∞ to 0. According to continuity equation the rate of change of horizontal velocity in x direction is proportional to rate of change of vertical velocity in y direction, so in order to compensate the drastic change in u, there is an equally abrupt change in v at the tip of the plate. But after that there are only negligible changes in the general profile of u and therefore v changes negligibly.

5. Velocity Profiles

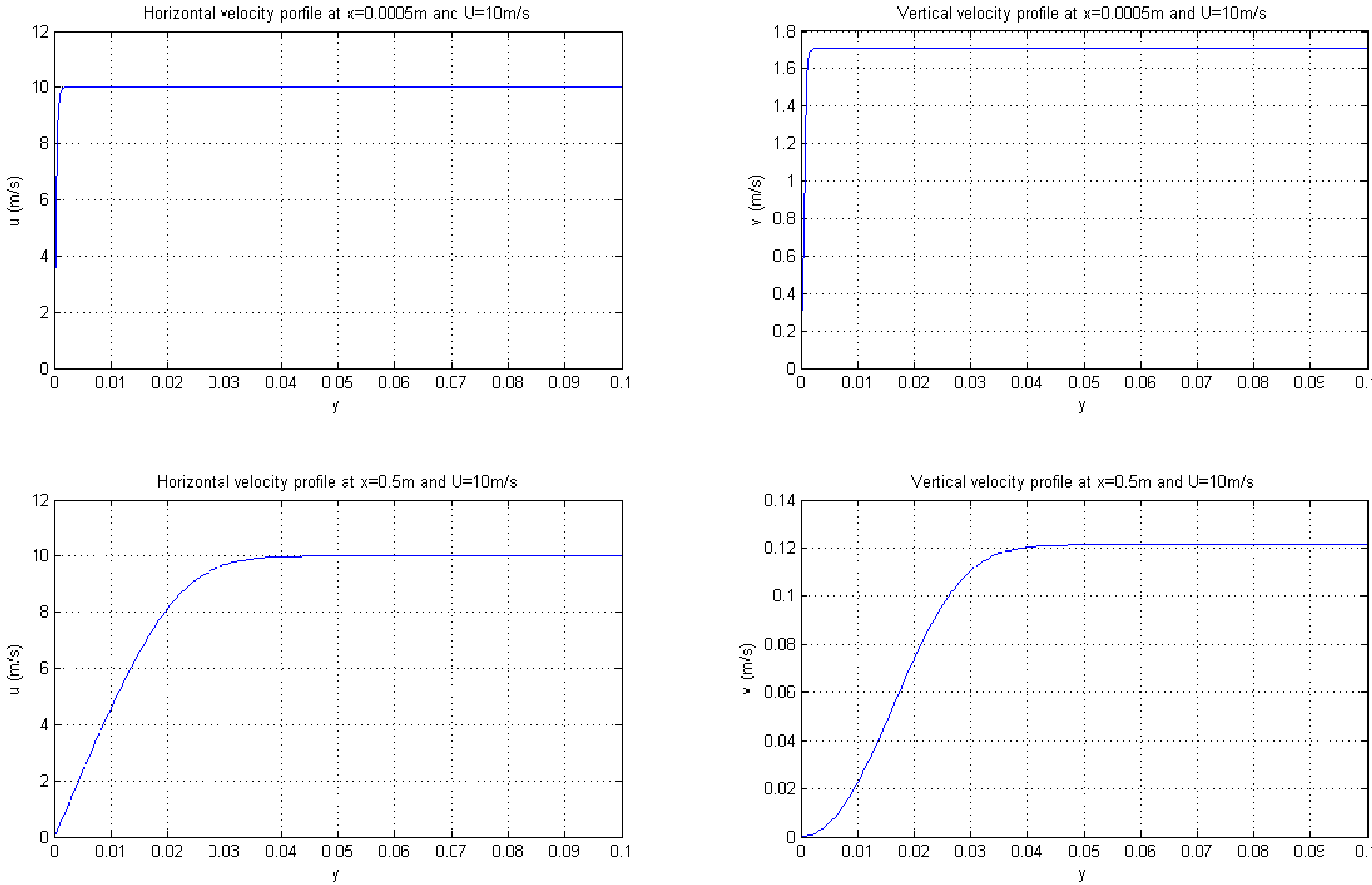

Figure 4: Velocity profiles

These 4 graphs show the horizontal and vertical velocity profiles over the flat plate at 0.0005m and 0.5m downstream. Since the horizontal velocity of the main stream was given by 10 m/s, the horizontal velocity profile at any place on the flat plate will always change from zero due to the no slip condition and up to 10m/s. It is clear from the horizontal velocity graph at 0.0005m that this change in the horizontal velocity profile is sudden, however, the change gets smoother when we go downstream as shown in the graph at 0.5m.

Since we are dealing with a laminar viscous flow, the vertical velocity component before the flow hits the flat plate was assumed to be zero, however, due to the sudden change in the horizontal velocity component, the vertical velocity had to rise suddenly also to reach 1.7m/s approximately at 0.0005m downstream to compensate the sudden retardation of horizontal velocity. As the flow continues downstream, the general shape of the horizontal velocity profile remains nearly the same and therefore the vertical velocity decreases again to reach 0.12m/s at 0.5m downstream.

6. Comparison with Blasius Solution

In order to be able to compare numerical and exact solutions of the boundary layer problem, first we need to unify the axes and to do so we used the following formula to transform y to η

$$\eta = \frac{y}{\delta(x)} = y\sqrt{\frac{Re}{x}}$$

The horizontal velocity component as calculated from Blasius exact solution can be considered as

$$\frac{u}{u_\infty} = f'(\eta)$$

The vertical velocity component can be calculated as follows

$$v = \sqrt{\frac{\nu\, u_\infty}{2x}}\,[\eta f'(\eta) - f(\eta)]$$

$$\frac{v}{u_\infty} = \sqrt{\frac{L}{2\,Re\,x}}\,[\eta f'(\eta) - f(\eta)]$$

After calculating and plotting the results we get the following graphs.

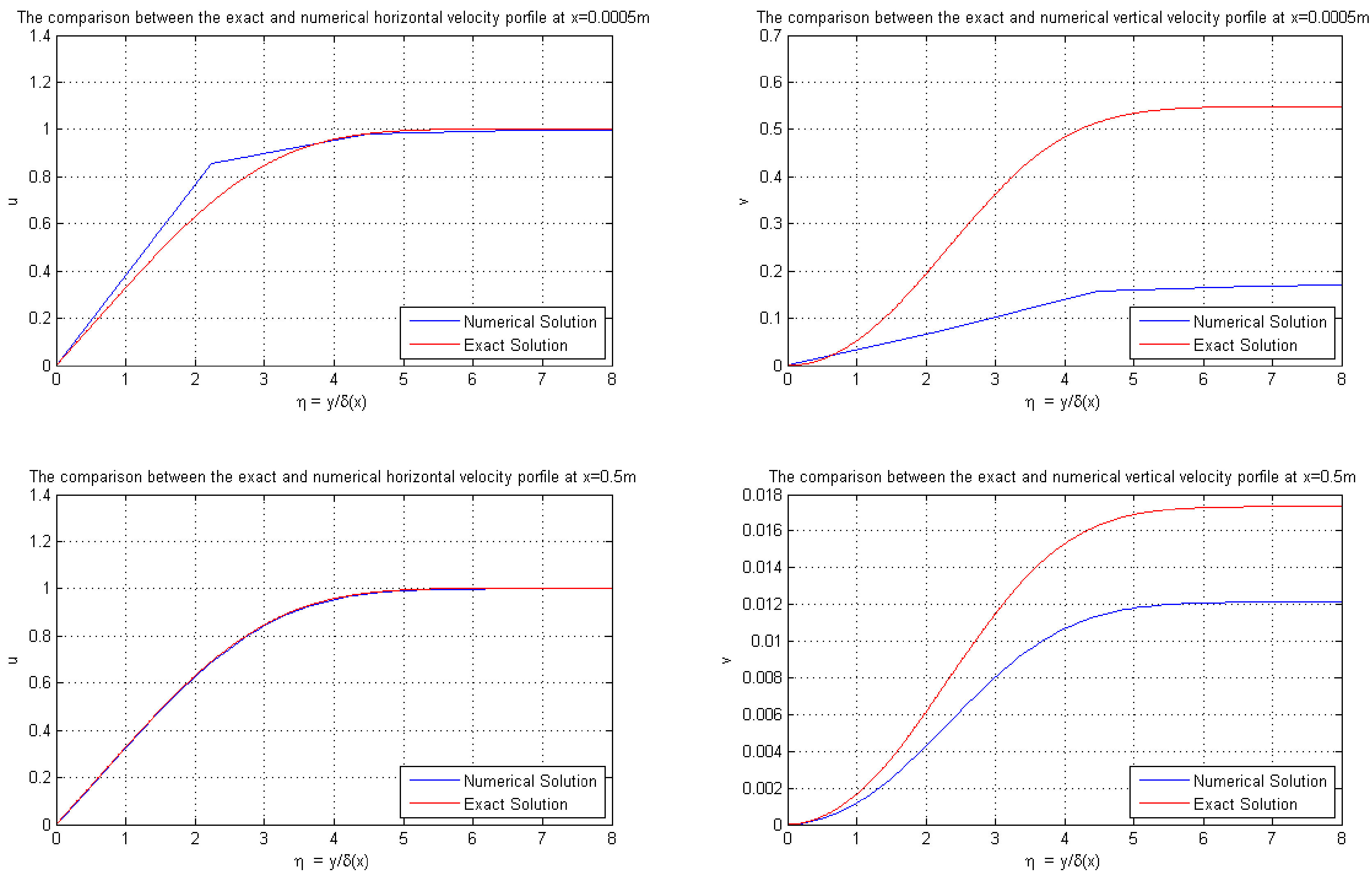

Figure 5: Comparison of numerical and Blasius solutions

From the graphs, it is clear that the deviation between numerical and exact solutions is larger upstream than it is downstream, that is because the sudden changes in velocity profiles at the edge of the plate requires finer grids than the ones that are being used here in order to be detected numerically in a more accurate manner.

Also, the error due linearizing the momentum equation by “Lagging” method is sensed more upstream where the changes are large. Other linearization methods like “Simple linearization” and “Newton linearization” iterative methods can be used to have better predictions to the coefficients and let the solution converge more quickly.

Errors in v might look larger than that of u due to the fact that v values are much smaller than u and with small values, MATLAB numerical accuracy will also affect the results.

But in general, the difference between the exact and numerical solution is still considered very low since the implicit methods usually converge quickly even with large grid sizes, even with “Lagging” linearization method, and that explains why the max error in the upstream graphs is less than 0.17 for u and less than 0.37 for v, even though the grid is not very fine.

Appendix

```
clear
close all
clc

% Given
h=0.0005;
Re=10000;

x=0:h:1;
y=0:h:2*5/Re^0.5;

% Boundary Conditions
v=zeros(length(y),length(x));
u=v;
u(length(y),:)=1;
u(:,1)=1;
u(1,1)=0;

% Initializing matrix for u
A=zeros(length(y)-2,length(y));
r=zeros(length(y)-2,1);
```

```
% Solving for u and v
for i=2:length(x)
    for j=2:length(y)-1
        A(j-1,j)=1+2/Re/h/u(j,i-1);
        A(j-1,j+1)=v(j,i-1)/2/u(j,i-1)-1/Re/h/u(j,i-1);
        A(j-1,j-1)=-v(j,i-1)/2/u(j,i-1)-1/Re/h/u(j,i-1);

        r(j-1)=u(j,i-1);
    end

    r(1)=r(1)-A(1,1)*u(1,i);
    r(end)=r(end)-A(end,end)*u(end,i);
    M=A(:,2:end-1);

    u(2:length(y)-1,i)=M\r;

    for j=2:length(y)
        v(j,i)=v(j-1,i)-0.5*(u(j,i)-u(j,i-1)+u(j-1,i)-u(j-1,i-1));
    end
end

% Showing results
SS=get(0,'ScreenSize');SW=SS(3);SH=SS(4);

figure('NumberTitle','off','OuterPosition',[0 30 SW SH-30]);
subplot(2,1,1);
imagesc(x,y,flipud(u)); colormap jet; colorbar; grid;
set(gca, 'XTick', linspace(0,max(x),21));
set(gca, 'YTick', linspace(0,max(y),11),'YTickLabel', linspace(max(y),0,11));
title('Distribution of horizontal dimentionless velocity on the whole domain'); xlabel('x'); ylabel('y');

subplot(2,1,2);
imagesc(x,y,flipud(v)); colormap jet; colorbar; grid;
set(gca, 'XTick', linspace(0,max(x),21));
set(gca, 'YTick', linspace(0,max(y),11),'YTickLabel', linspace(max(y),0,11));
title('Distribution of vertical dimentionless velocity on the whole domain'); xlabel('x'); ylabel('y');

figure('NumberTitle','off','OuterPosition',[0 30 SW SH-30]);
subplot(2,2,1); plot(y,u(:,x==0.0005)*10); grid;
```

```
title('Horizontal velocity porfile at x=0.0005m and U=10m/s'); xlabel('y'); ylabel('u (m/s)');

subplot(2,2,2); plot(y,v(:,x==0.0005)*10); grid;
title('Vertical velocity profile at x=0.0005m and U=10m/s');  xlabel('y'); ylabel('v (m/s)');

subplot(2,2,3); plot(y,u(:,x==0.5)*10); grid;
title('Horizontal velocity profile at x=0.5m and U=10m/s');  xlabel('y'); ylabel('u (m/s)');

subplot(2,2,4); plot(y,v(:,x==0.5)*10); grid;
title('Vertical velocity profile at x=0.5m and U=10m/s');  xlabel('y'); ylabel('v (m/s)');

% Blasius exact solution
deta=0.015;
eta=0:deta:8;
f=zeros(1,length(eta)); df=f; ddf=f; ddf(1)=0.3319;

for i=2:length(eta);
    f(i)=f(i-1)+df(i-1)*deta;
    df(i)=df(i-1)+ddf(i-1)*deta;
    ddf(i)=ddf(i-1)-0.5*f(i-1)*ddf(i-1)*deta;
end
uExact=df;
vExact1=(1/Re/0.0005/2)^0.5*(eta.*df-f);
vExact2=(1/Re/0.5/2)^0.5*(eta.*df-f);

% Comparing numerical to exact solutions
figure('NumberTitle','off','OuterPosition',[0 30 SW SH-30]);
subplot(2,2,1); plot(y*(Re/0.0005)^0.5,u(:,x==0.0005),eta,uExact,'r'); xlim([0 max(eta)]);grid;
title('The comparison between the exact and numerical horizontal velocity porfile at x=0.0005m');
xlabel('\eta = y/\delta(x)'); ylabel('u');
legend('Numerical Solution','Exact Solution','Location','SouthEast');

subplot(2,2,2); plot(y*(Re/0.0005)^0.5,v(:,x==0.0005),eta,vExact1,'r'); xlim([0 max(eta)]);grid;
title('The comparison between the exact and numerical vertical velocity porfile at x=0.0005m');
xlabel('\eta  = y/\delta(x)'); ylabel('v');
legend('Numerical Solution','Exact Solution','Location','SouthEast');

subplot(2,2,3); plot(y*(Re/0.5)^0.5,u(:,x==0.5),eta,uExact,'r'); xlim([0 max(eta)]);grid;
title('The comparison between the exact and numerical horizontal velocity porfile at x=0.5m');
```

```
xlabel('\eta  = y/\delta(x)'); ylabel('u');
legend('Numerical Solution','Exact Solution','Location','SouthEast');

subplot(2,2,4); plot(y*(Re/0.5)^0.5,v(:,x==0.5),eta,vExact2,'r'); xlim([0 max(eta)]);grid;
title('The comparison between the exact and numerical vertical velocity porfile at x=0.5m');
xlabel('\eta  = y/\delta(x)'); ylabel('v');
legend('Numerical Solution','Exact Solution','Location','SouthEast');
```