Numerical Solution of Similar Boundary Layer Equations

Laura Jones

Table of Contents

Section 1: The Blasius Boundary Layer On A Flat Plate At Zero Pressure Gradient 1

1.2.1 Express u and v in similarity variables 2

1.2.2. Derive the Blasius equation 2

1.2.3 Solve the Blasius equation numerically 3

1.2.4 Study the effect of wall transpiration 5

Section 2: Pressure-gradient effects on the flat plate boundary layer and wedge flows 7

2.2.1 Solve the Falkner-Skan Equation Numerically 8

2.2.2 Perform Parametric Studies on the Pressure Gradient 8

2.2.3 Perform Parametric Studies on Wall Transpiration 11

Section 3: Compressible Boundary Layer on a Flat Plate 14

3.2.1 Solve the System Numerically for a Positive Value of β 15

3.2.2 Perform Parametric Studies on β in the Range -0.1 < β < 0.1 16

3.2.3 Perform Parametric Studies on S in the Range -1 < S < 1 17

Conclusions 19

Bibliography 19

Appendix 20

# Section 1: The Blasius Boundary Layer On A Flat Plate At Zero Pressure Gradient

For the flow of fluid over a flat plate, the equations that govern the flow (Navier-Stokes) can be simplified. Using the assumptions of laminar and incompressible flow, the Navier-Stokes equations can be reduced to:

Where u and v are the stream wise and wall-normal velocity components, and is the kinematic viscosity of the flow.

The Blasius similarity solution can be applied to the equations by introducing a function called the stream function, defined as:

The Blasius similarity solutions is applied to these equations. This is where the independent variables are grouped together rather than each variable being solved individually (Wilson, 2014). There is then only one variable to be derived and the solution solved for. For this problem the following grouped variables were used:

Using the self-similar variables and to define the flow:

The flow can be defined by the relationship between the stream function and :

## 1.2.1 Express u and v in similarity variables

### For u:

### For v:

## 1.2.2. Derive the Blasius equation

Continuity equation**:**

Momentum equation :

Substituting these into the momentum equation:

### Re-arranging the substitution gives the Blasius equation:

## 1.2.3 Solve the Blasius equation numerically

The Blasius Equations were solved numerically using MatLab, although any high-level programming language would be appropriate- eg FORTRAN or C as these are both anti-aliasing languages.

The initial conditions imposed were:

* No slip at the wall
* Outside the boundary layer, the velocity is the free stream velocity 1

The solution of the equation was found using the shooting method and bisection in order to find a solution that met a defined tolerance of 0.001. The tolerance was set to get a balance between the computation time and accuracy of results as well as reducing the time taken to find a solution as the shooting and bisection method is very sensitive to the guess paramaters imposed- if the solution is not within the guess parameters the solution can never converge.

The code written can be found in Appendix A. The results produced are shown below.

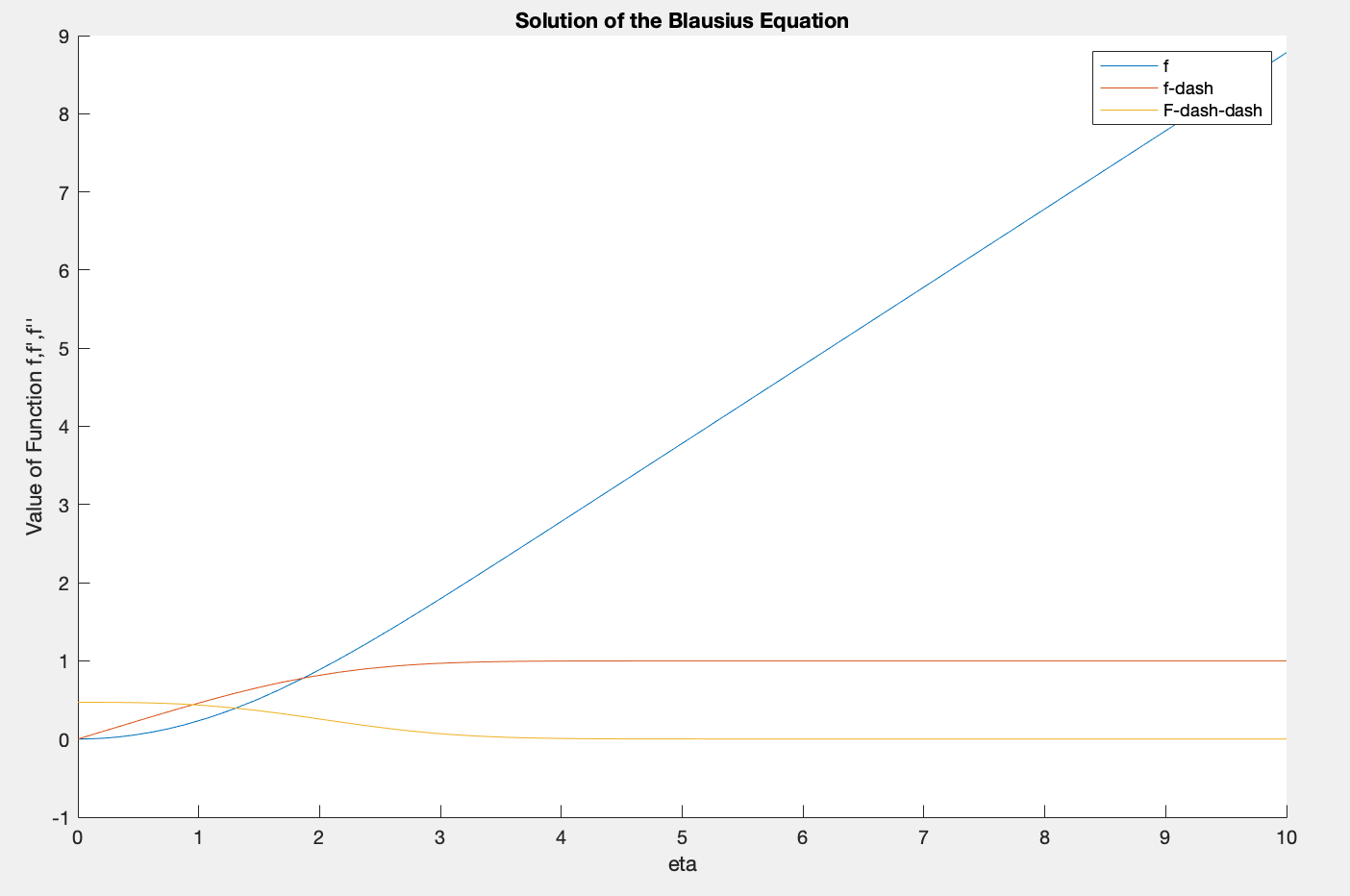


Figure 1 Solution of the Blasius equation

The solution of f was 0.4699 to 4 decimal places. When compared to literature, this value is close the values obtained by others. (Parveen, 2014) This suggests that the method used and derivation of the similarity variables is acuurate to an acceptable level and the rest of the investigation can be continued.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Source | Result of investigation | Parveen et al.  (Parveen, 2014) | Asalthembi  (Asaithambi, 1998) | Liao  (Liao, 1999) |
| Solution of =0 | 0.4699 | 0.46900 | 0.469601 | 0.46900 |

The figure could also be transposed to see the flow development as more representative of the physical boundary layer- the line f shows the height of the boundary layerand f’ is representative of the velocity profile.

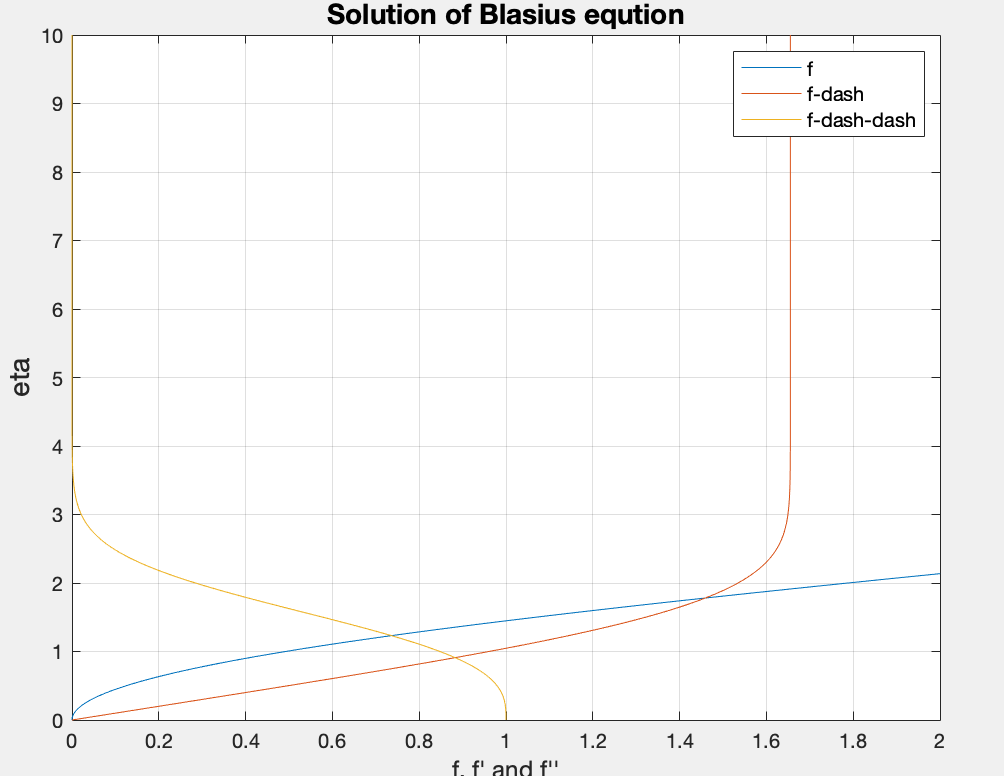


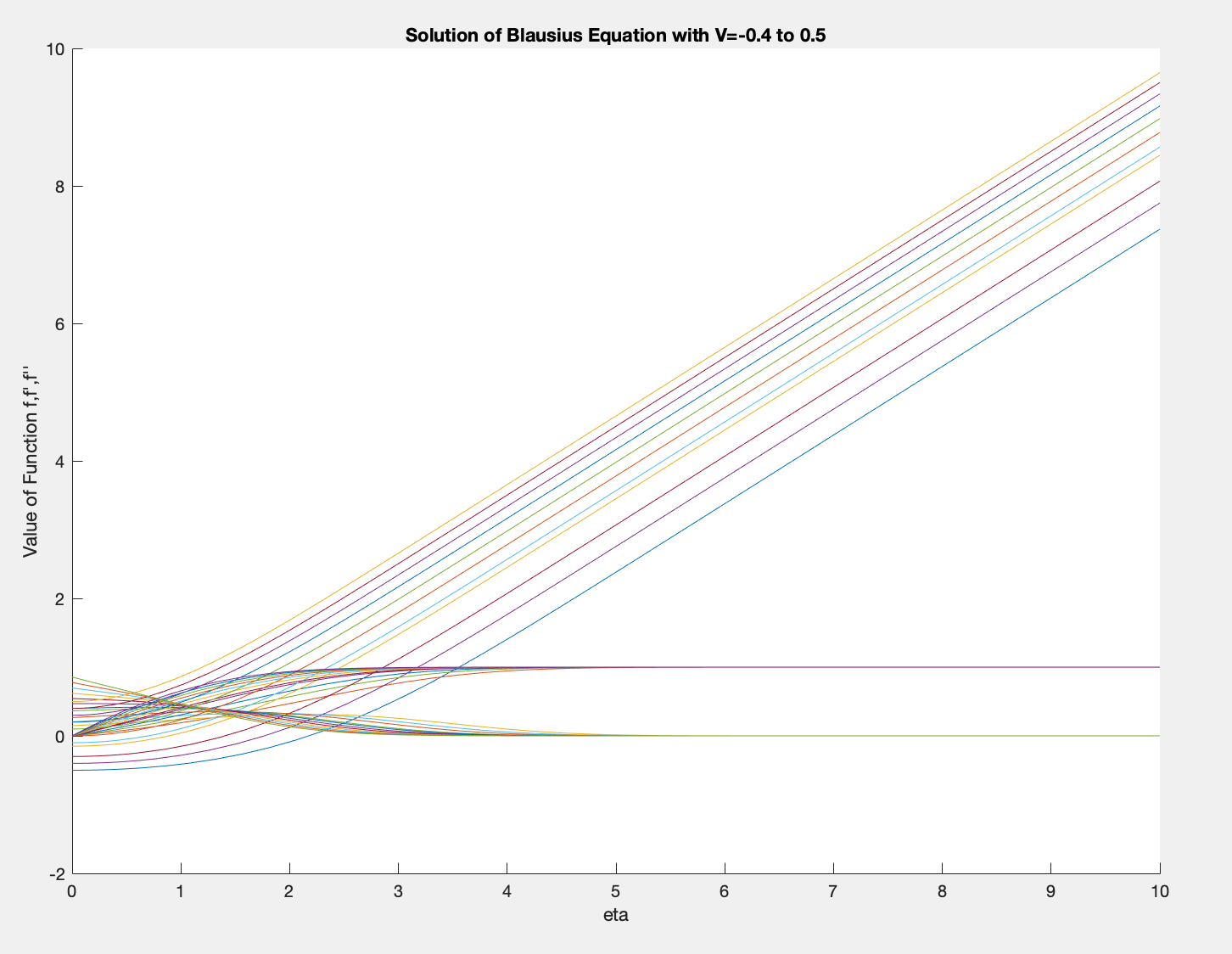
Figure 2 Solution of Blasius equation transposed to show physical boundary layer properties

## 1.2.4 Study the effect of wall transpiration

By varying the value of , the wall transpiration can be changed. A positive valve of Vw indicates ‘blowing’ and a negative value indicates ‘sucking’. The effect of wall transpiration on the Blasius solution was investigated.

|  |  |  |
| --- | --- | --- |
| Vw |  | Wall shear stress |
| -0.4 | 0.2812 | 0.2803 |
| -0.3 | 0.3378 | 0.3374 |
| -0.2 | 0.3985 | 0.3980 |
| -0.1 | 0.4640 | 0.4648 |
| 0 | 0.5313 | 0.6000 |
| 0.1 | 0.6020 | 0.6010 |
| 0.2 | 0.6741 | 0.6741 |
| 0.3 | 0.7475 | 0.7445 |
| 0.4 | 0.8383 | 0.8305 |
| 0.5 | 0.9052 | 0.9035 |

Figure 3 Solutions of the Blasius equation for different wall transpiration situations



Looking at the velocity profile of the solutions (shown below), the effect of a large negative value of Vw causes the development of the boundary layer to be slower as the gradient of the velocity ratio curve is smaller. By increasing the value of Vw, the wall shear stress increases up to 1 when Vw increases beyond 0.5. In a physical sence, this appears to agree as when the boundary layer is ‘sucked’ through the wall, the development is expected to be slower as the bottom layers are removed. The plot below shows that at large, negative values of Vw, the development of the boundary layer is more linear as the line is straighter and less curved, suggesting that the turbulence in the boundary layer is greatly reduced- this is seen to have a greater effect when the wedge flows and compressible flows are investigated.

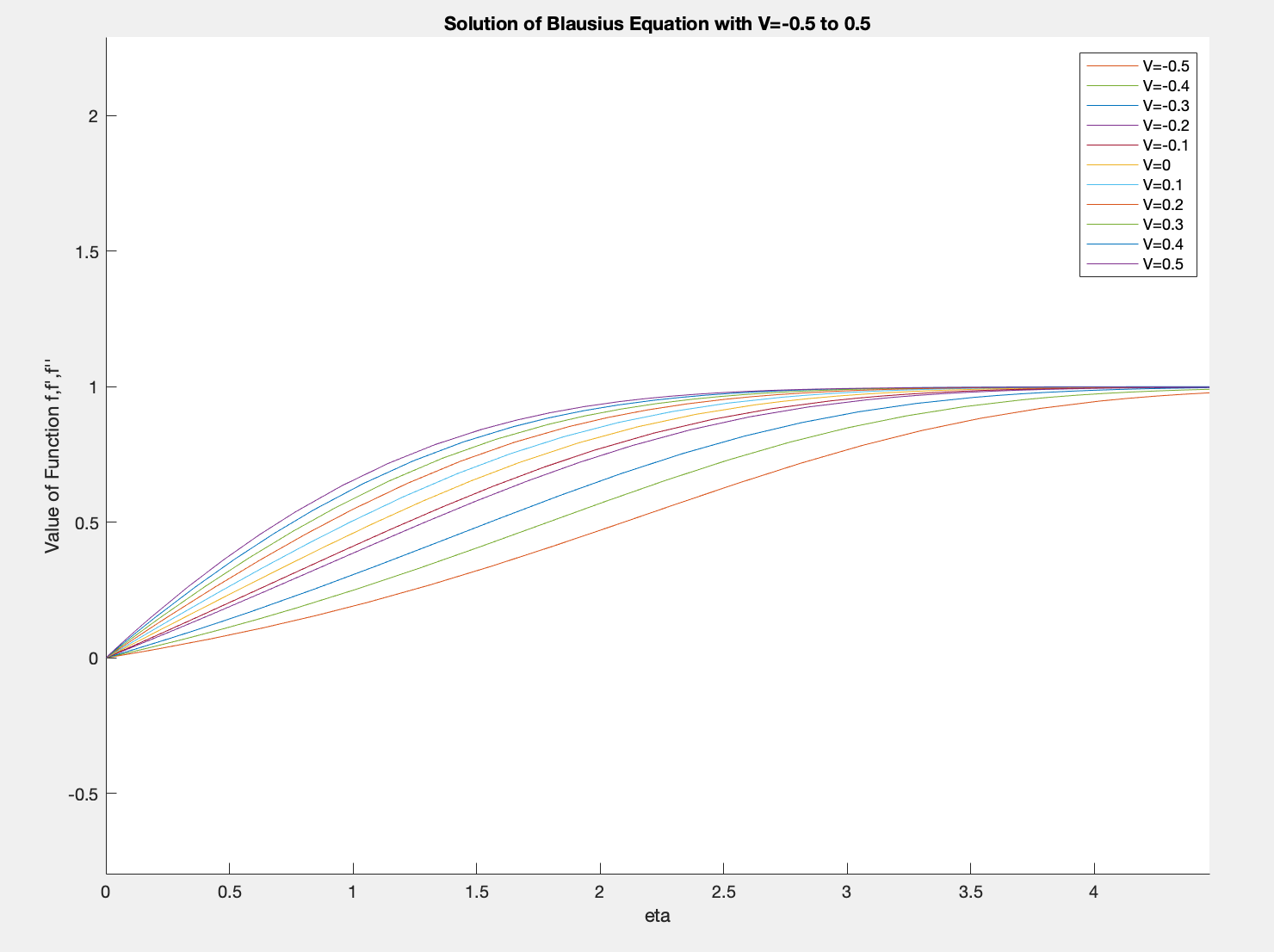


Figure 3 Velocity profile of the boundary layer for a range of Vw

# Section 2: Pressure-gradient effects on the flat plate boundary layer and wedge flows

In wedge flows or flows with pressure gradients, the Blasius solution can be modified to become the Falkner-Skan equations. The boundary conditions described in the section above still remain but a new parameter (β) is introduced. The value of β is dependent on the equivalent value of the pressure gradient dp/dx. If the equivalent pressure gradient is positive, then the value of β is positive, for a flat plate the value of β would remain as 0. For an adverse or negative pressure gradient, the value of β will be negative (Javed & Ali Shah, 2017)

can be defined by a similarity variable:

Where m is a function that defines the free stream velocity distribution over the wedge and is given by:

The similarity variable eta is modified to be dependent on the gradient function, m:

This produces the Falkner-Skan equation in place of the Blasius:

## 2.2.1 Solve the Falkner-Skan Equation Numerically

To solve the Falkner-Skan equation numerically, the value of was set as 0.05 for the investigation. 2 new ‘guess’ variables had to be introduced for S’ as the values at was unknown. These were then set to converge in a similar way to the guess variable for of with a tolerance of 0.001 set.

The solution of was 0.5312. The solution is shown below and the code used can be seen in Appendix A.

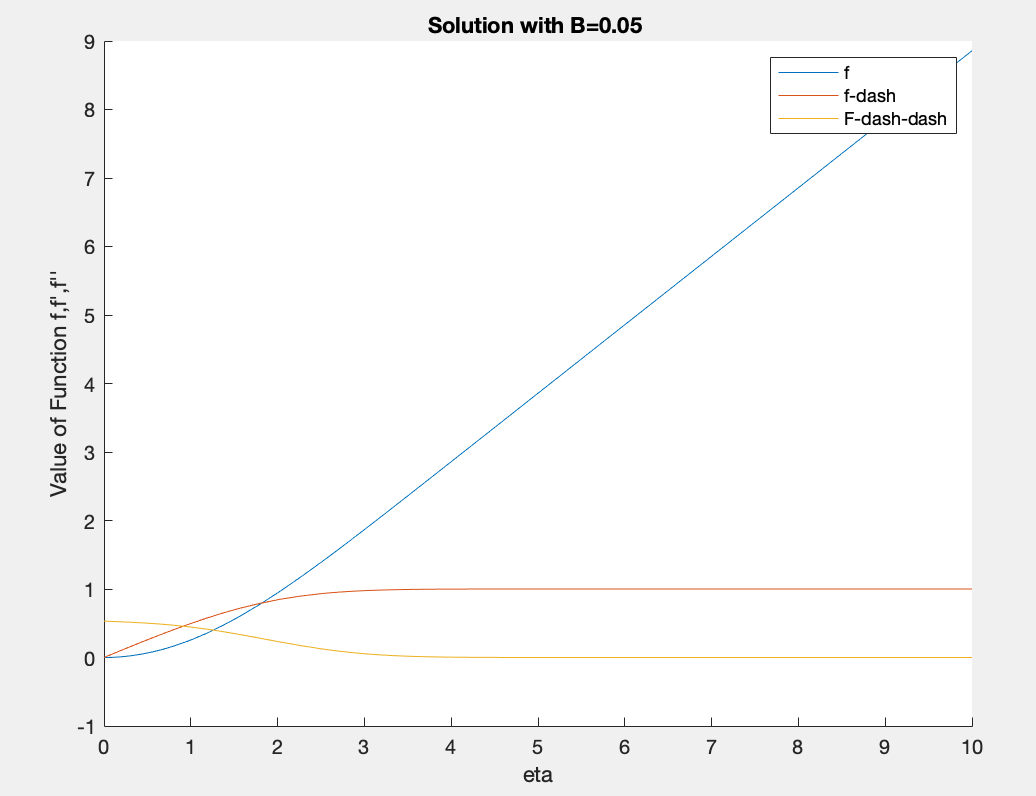


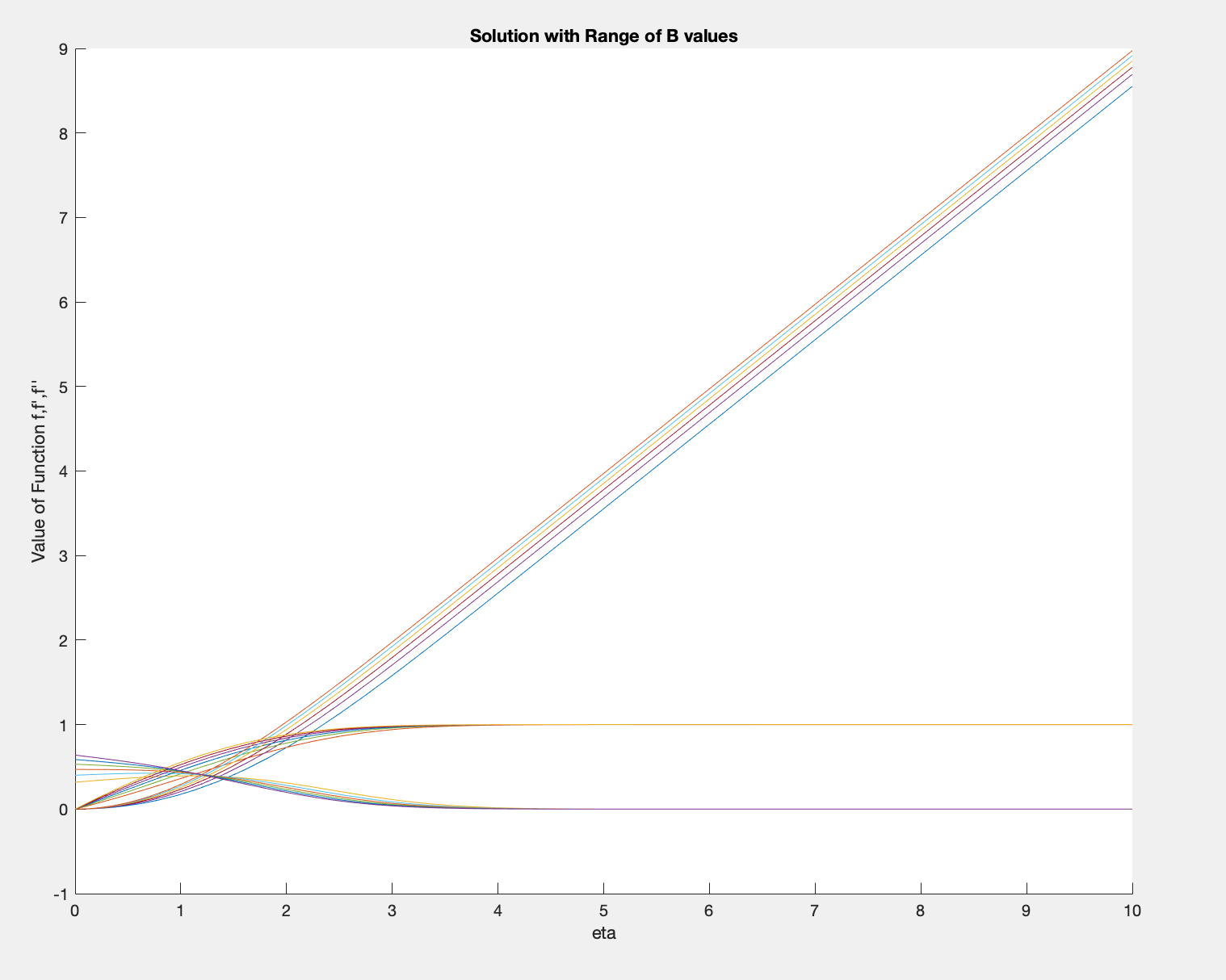
Figure 5 Solution with B=0.05

## 2.2.2 Perform Parametric Studies on the Pressure Gradient

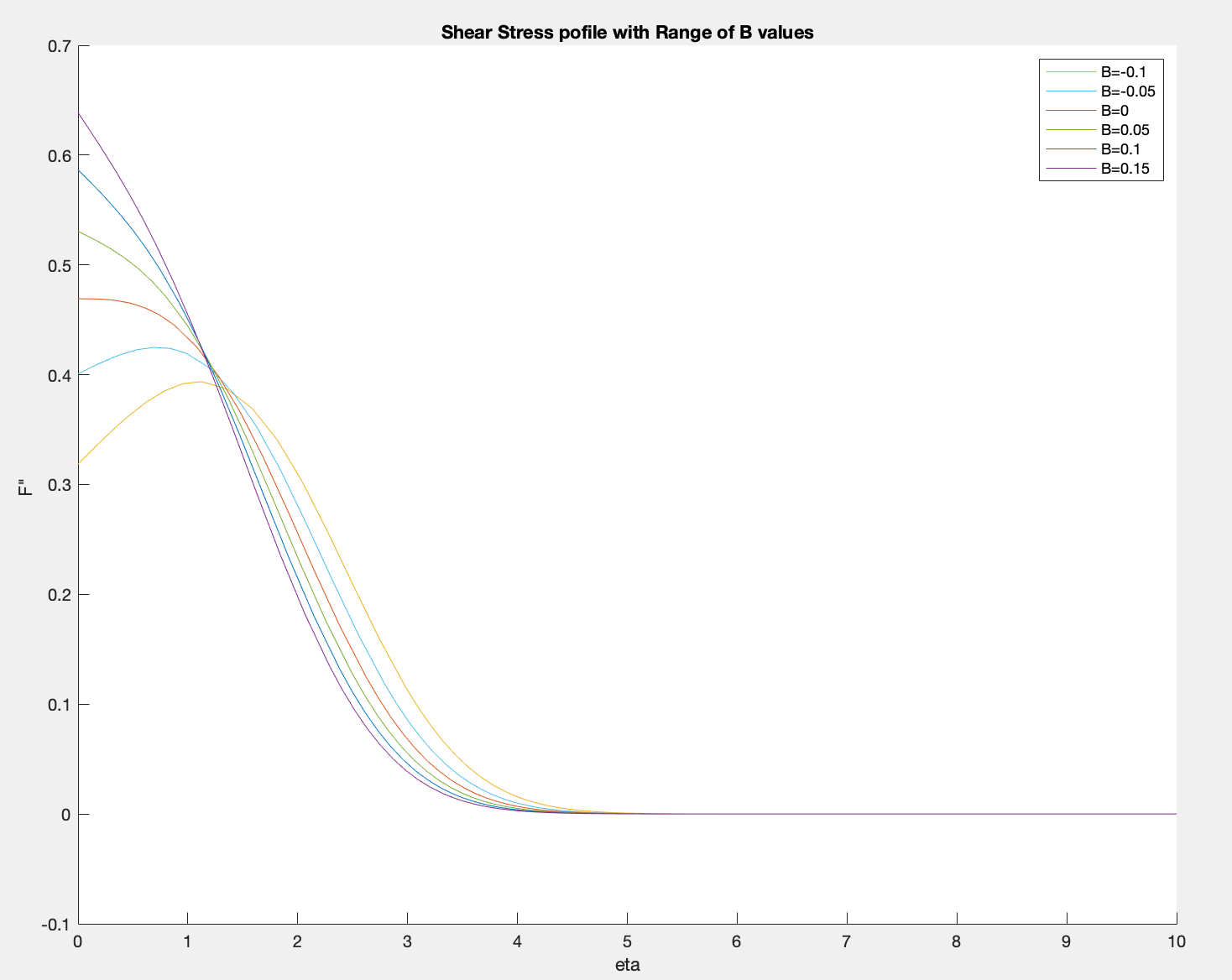
The solution over a range of values were investigated and the results are shown below.

|  |  |
| --- | --- |
| Value of |  |
| -0.1 | 0.3176 |
| -0.05 | 0.4003 |
| 0 | 0.4680 |
| 0.05 | 0.5303 |
| 0.1 | 0.5861 |
| 0.15 | 0.6377 |

Figure 6 Solution for a range of B values



Looking at the profile for the shear stress (shown below), it can be seen that the wall shear stress increases with increasing values of . However, when the value of becomes less than 0, the point at which the shear stress is at a maximum is shifted streamwise to a larger value of . The boundary layer velocity profile also develops at a slower rate which can be seen in the velocity profile graph in Appendix C.



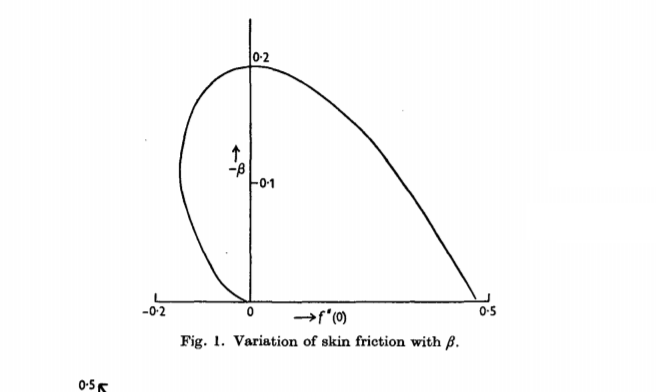


Figure 8 Solution of β for skin friction (Stewartson, 1953)

Shift in maximum shear stress

Figure 7 Shear stress profile for a range of B values

The limit of the value of are described by Stewartson as seen to the left. The value of reaches a limit in the negative as there will be 2 solutions. If the wall shear stress is negative, then there are 2 potential values of that can satisfy the flow state- the correct value should be chosen to ensure the expected value of wall shear stress is found. Theoretically should have not limit in the positive direction, however the code written was unable to find a solution for a value above 0.2. Hartree showed in 1937 that the limits were:

(Hartree, 1937)

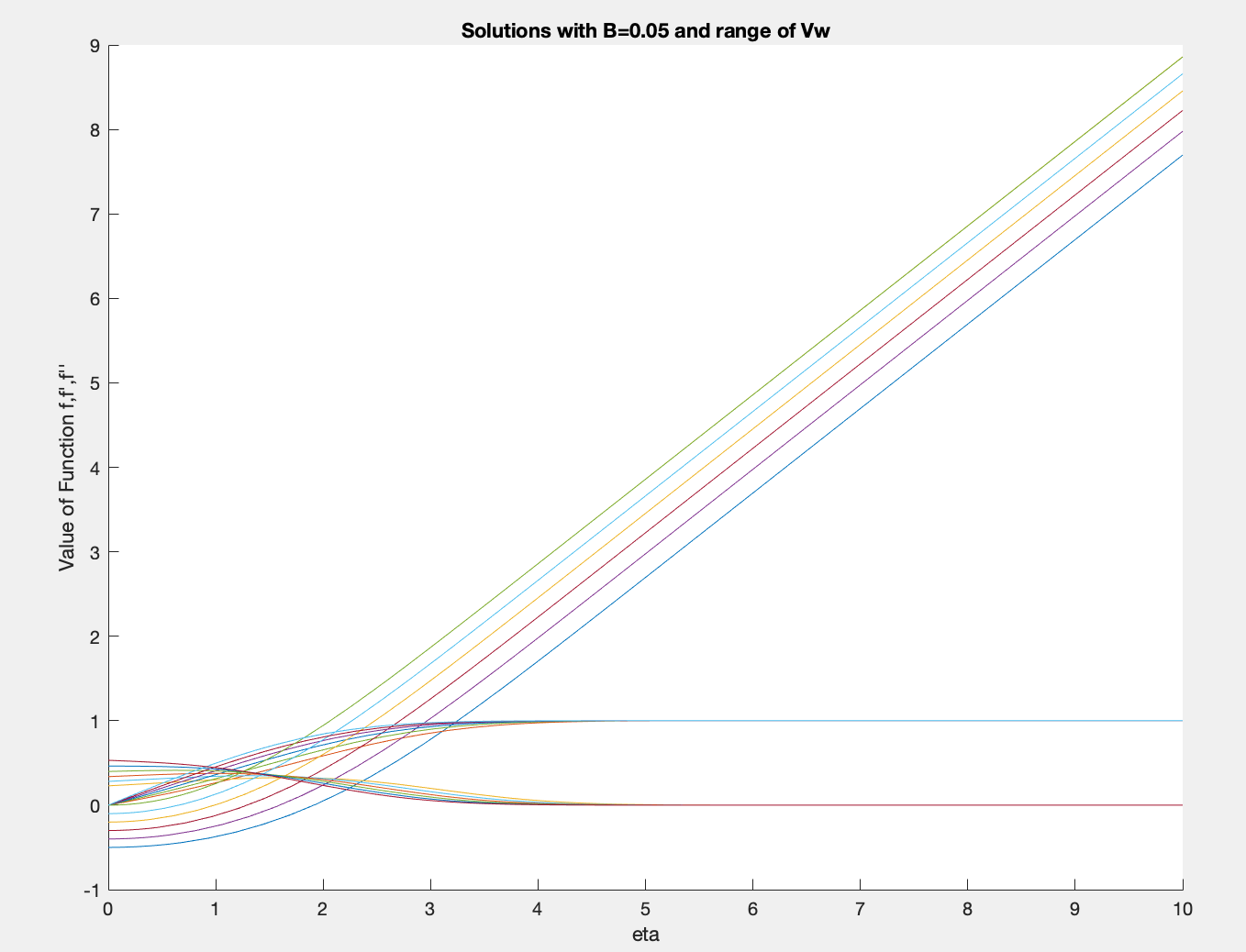
For more negative values of , the implication is that the velocity in the boundary layer is greater than the free stream- which is physically unacceptable.

However, the calculation and method used in this investigation could not converge on a solution above 0.2.

## 2.2.3 Perform Parametric Studies on Wall Transpiration

By changing the value of Vw, the wall transpiration can be modified. Positive values of Vw represent ‘blowing’ and negative represent ‘sucking’ of fluid over a porous wall. Keeping the same value of , the effect of changing Vw can be seen:

Figure 9 Solution of F-K equations with a range of Vw



The velocity profile of the flow at different Vw values show how the boundary layer develops. At large, positive values of Vw, the boundary layer develops faster than no transpiration. With a negative value, the boundary layer develops more slowly as the lower layers are sucked through the wall, which causes the gradient of the velocity to decrease.

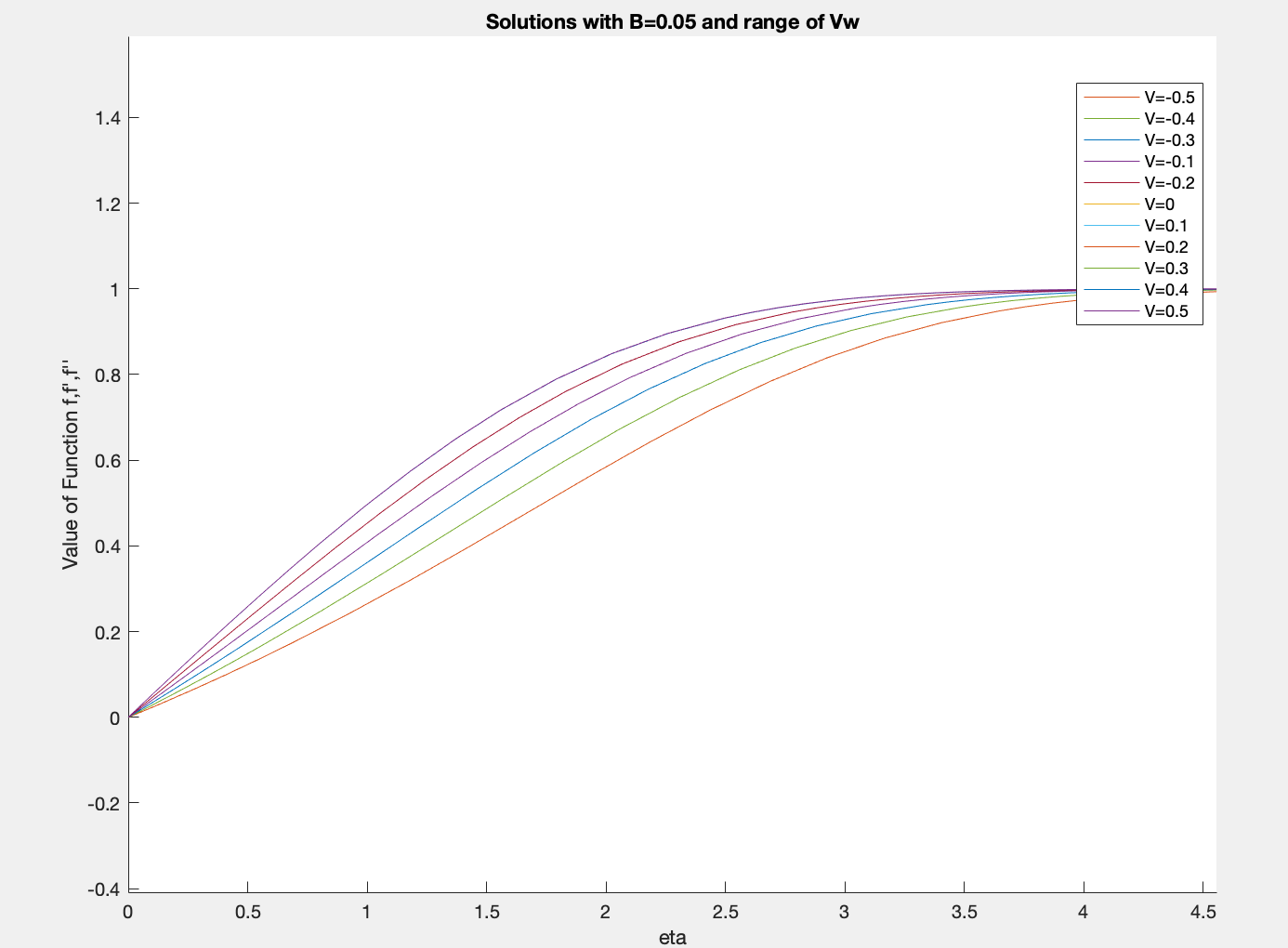


Figure 10 Velocity profile of the boundary layer for a range of Vw values

The shear stress at the wall increases with increasing Vw as expected. As seen in the earlier section, the point of maximum shear stress is shifted to increasing values. This was seen with the studies in section 1. The maximum shear stress has been shifted further stream-wise than seen earlier- this is expected as the flow is now wedge flow and so the pressure gradient is working complimentary to the flow direction, this can be seen in Appendix F.

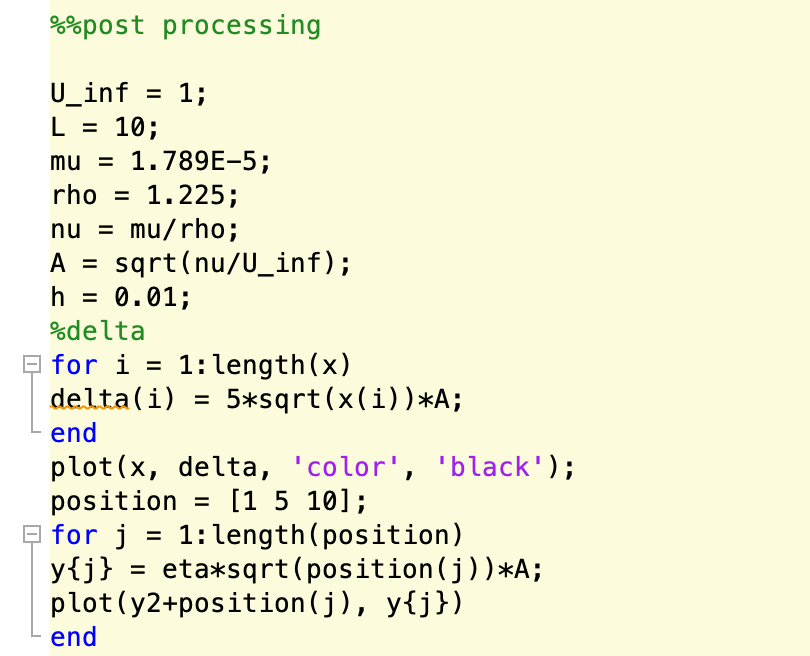


Figure 11 Code used

The boundary layer thickness can then be calculated with the variable defined as in the code. The boundary layer thickness can then be calculated numerically. (Stewartson, 1953)

The plot below shows the boundary layer thickness for a plate of 10m and the variables with conditions imposed as shown above. The velocity profile is also shown at points along x.

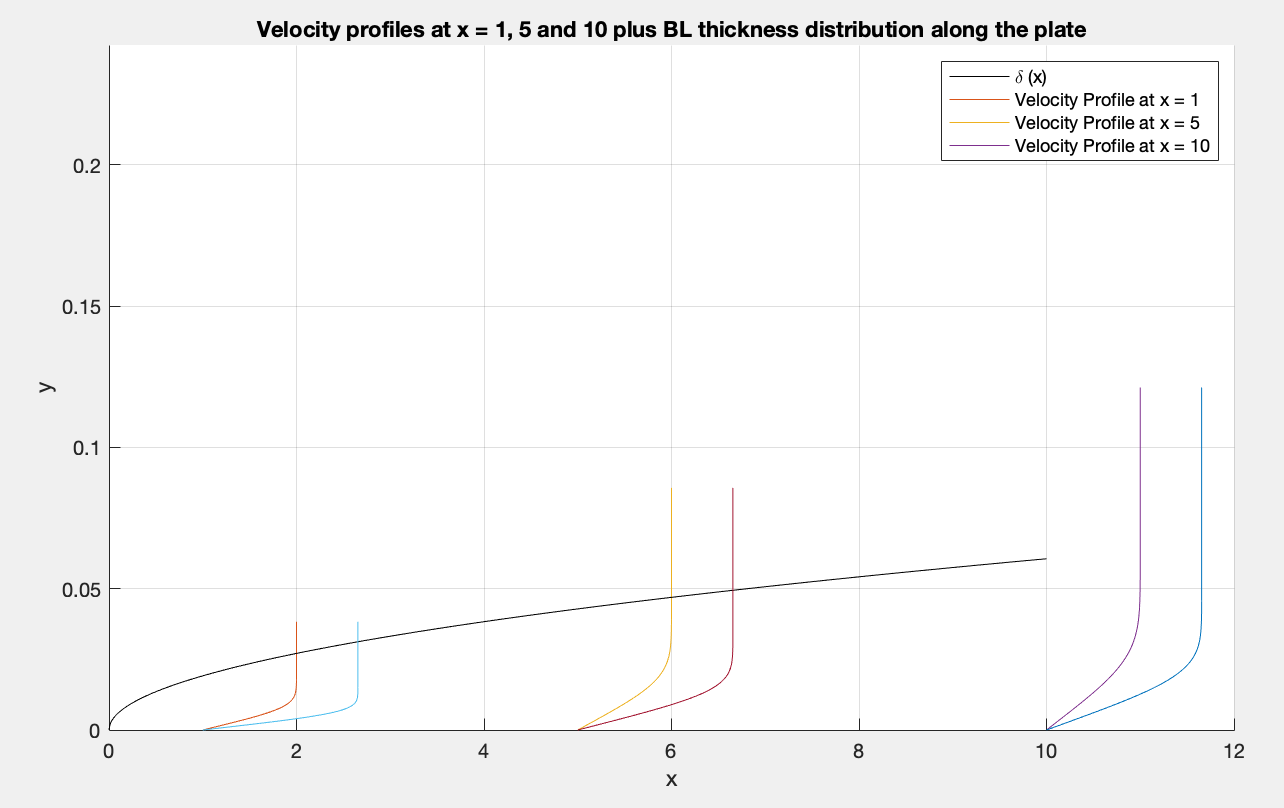


Figure 12 Velocity profiles and boundary layer thickness for plate of length 10

The momentum thickness is the distance that the external potential flow is pushed outwards due to the boundary layer velocity being less than the free-stream. (Weyburne, 2006)

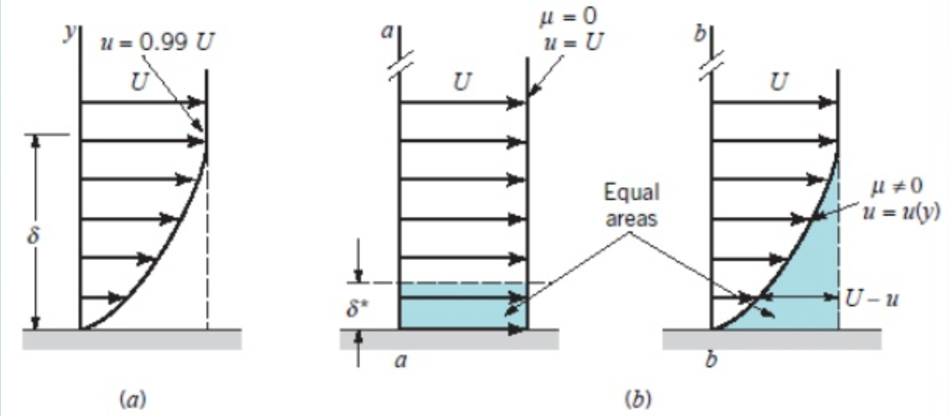


Figure 13 Momentum thickness equivalent

The skin friction coefficient based on local parameters in the flow can be calculated from:

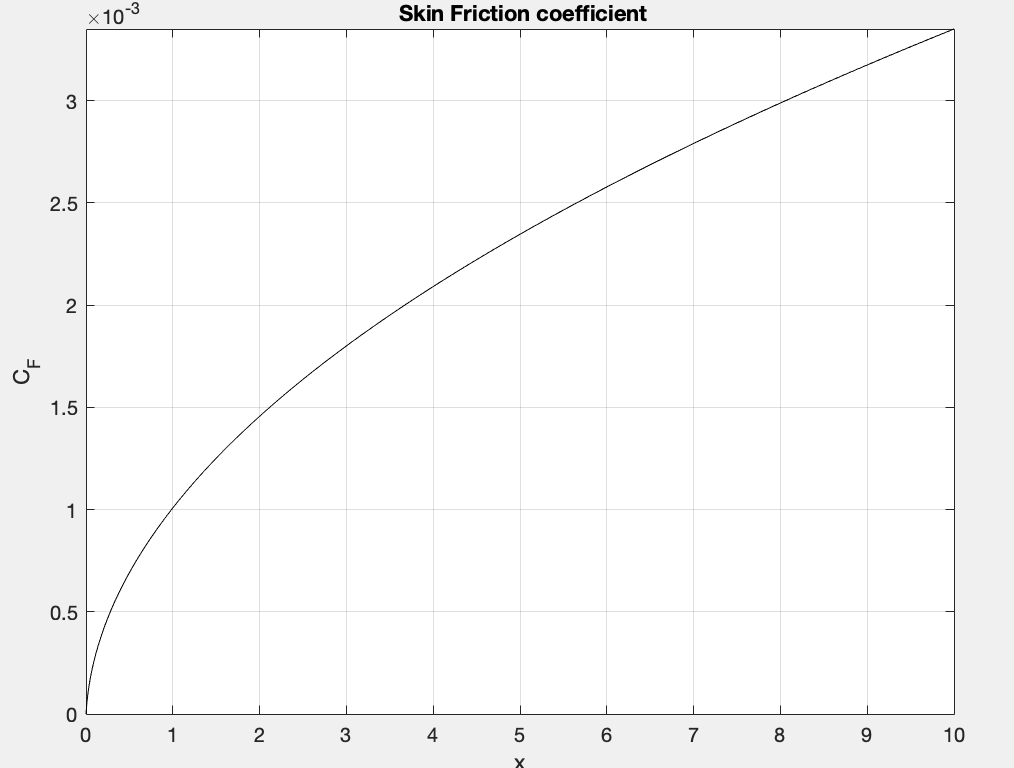


Figure 14 Skin Friction coefficient cumulative over the plate

# Section 3: Compressible Boundary Layer on a Flat Plate

The previous sections have focused on incompressible flow and have assumed as such, this is ok if the fluid is incompressible and the Mach number is very close to 0. When the Mach number of a fluid flow becomes finite and larger than 0.3, the compressibility of the flow must be considered as the density of the flow can change. This allows for the creation of two separate but dependant boundary layers- one of viscous flow and one due to thermal effects. This required a new definition of the similarity variable and the introduction of the energy equation.

The re-arrangement of these equations results in 2 sets differential equations that must be solved simultaneously:

S is a function of the flow enthalpy, which is defined by the temperature and energy boundary layer in the flow. The flow temperature can be expressed, with γ as specific heat a constant for air of 1.4.:

## 3.2.1 Solve the System Numerically for a Positive Value of β

To solve the equations numerically the following two third order and second order (respectively) ODEs were split into 5 single order ODEs:

To do this, take a Nth-order differential equation, i.e ( (Kahrom, n.d.)

Y(N) = f[x, y, y’, y’’,…yn-2, yn-1]

Variables are defined in order i.e. f(1) = y, f(2)=y’ f(3)=y’’ etc. The derivative functions are then calculated as follows:

y(3) = f(X, Y(N), Y(N-1)…..Y(2), Y(1))

y(2)=Y(n-1)

y(1)=Y(n-2)

…………………………..

F(N)=Y(N-1)

The single order ODEs from this are entered into MATLAB to solve the system numerically. This produced the following five single order ODEs:

y(1) = f (2)

y(2) = f (3)

y(3) =-f (1)f(3) - β(1-f(2)2 - f(5))

y(4) = -f(1)f(4)

y(5) = f(4)

The initial conditions of the flow must be imposed:

S = S0 & f=f '=0 when η = 0, and f’ -> 1 & S-> 0 as η -> ∞

The flow was solved with =0.1 and S=0.5

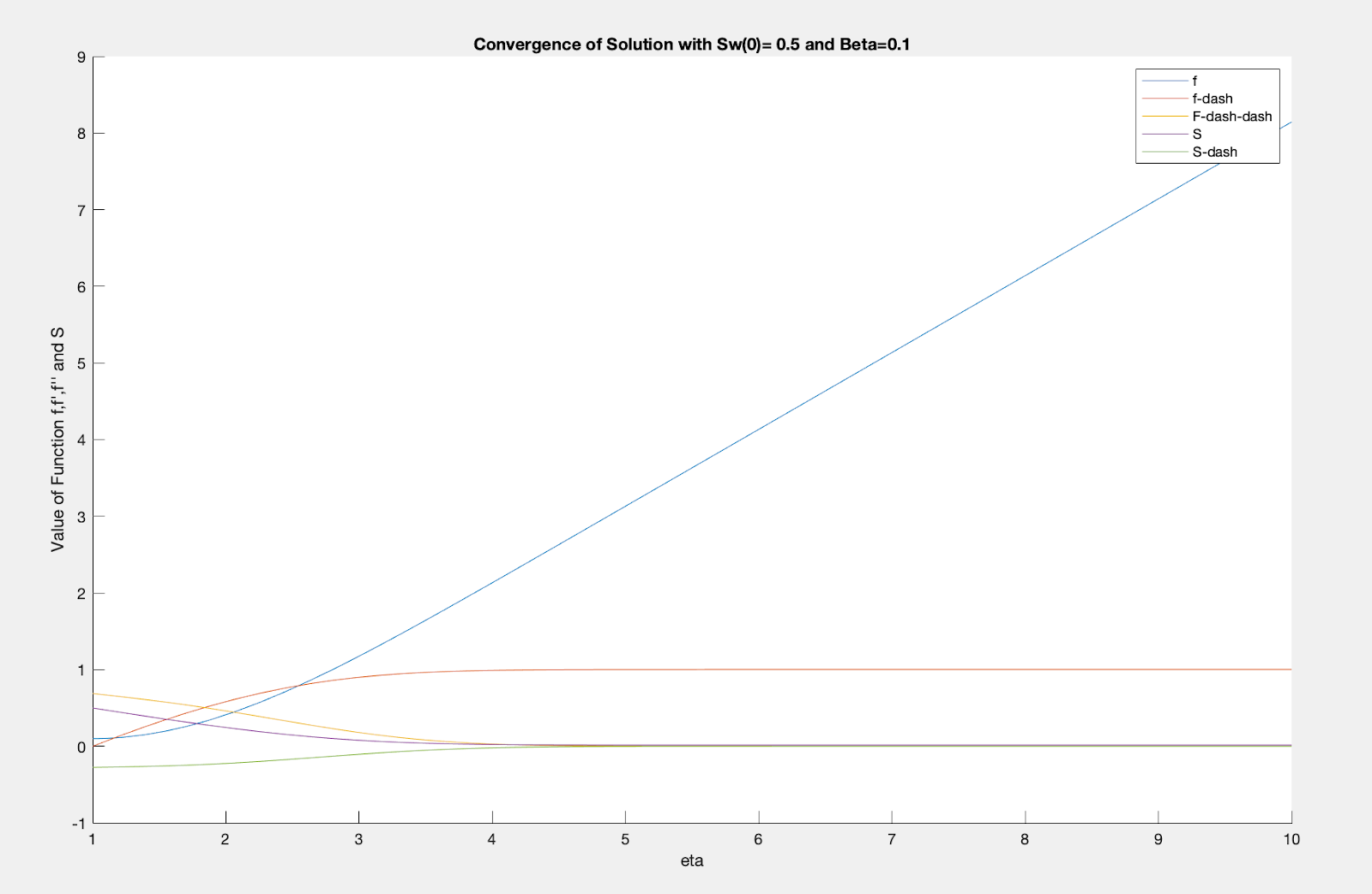


Figure 16 Solution with B=0.1 and Sw=0.5

The solution for the Blasius equation in this case was 0.6191. The relationships seen in the figure above are similar to those described by Oudheusden in 1997 and so the results were trusted to be accurate to a sufficient level. (Van Oudheusden, 1997)

The code for the calculation can be found in Appendix A.

## 3.2.2 Perform Parametric Studies on β in the Range -0.1 < β < 0.1

To investigate how the pressure gradient affects the flow, the value of Sw was set to 0.5 and a range of values were calculated. The plotted results can be seen below.

|  |  |  |
| --- | --- | --- |
| B value | F solution | S-solution |
| -0.1 | 0.2781 | -0.2 |
| -0.05 | 0.3875 | -0.2273 |
| 0 | 0.4695 | -0.2312 |
| 0.05 | 0.5436 | -0.2417 |
| 0.1 | 0.6191 | -0.2484 |

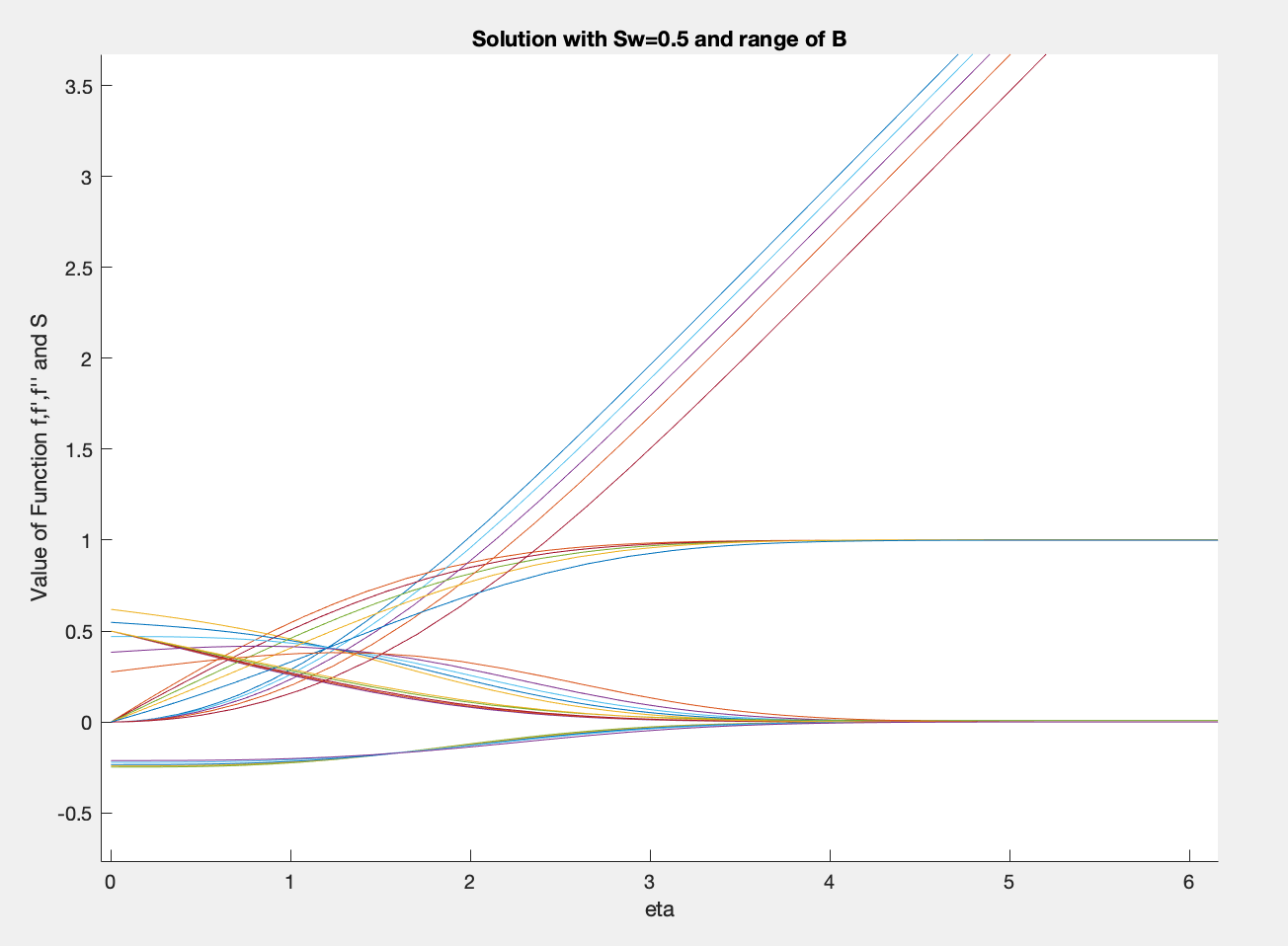


Figure 17 Solutions of flow with range of B values

From the velocity profile in Appendix D, it can be seen that reducing the value of caused the boundary layer to develop more slowly and the gradient of the development to be more linear. With a higher value of , the boundary layer reached 95% of the free stream velocity sooner than a lower value of - this makes physical sense as the flow is encouraged to develop by a favourable pressure gradient.

The same shear stress shift is seen with the variation of as seen previously, the minimum value tested caused the location of the maximum shear stress to be shifted to . This can be seen in the plot in Appendix E.

## 3.2.3 Perform Parametric Studies on S in the Range -1 < S < 1

To asses how the value of S(0) affected the boundary layer, the value of was set to 0.05 and a range of S(0) values were calculated.

|  |  |  |
| --- | --- | --- |
| Sw value | F solution | S-solution |
| -1 | 0.5023 | 0.4883 |
| -0.5 | 0.5117 | 0.2305 |
| 0 | 0.5313 | 0 |
| 0.5 | 0.5374 | -0.2408 |
| 1 | 0.5751 | -0.4906 |

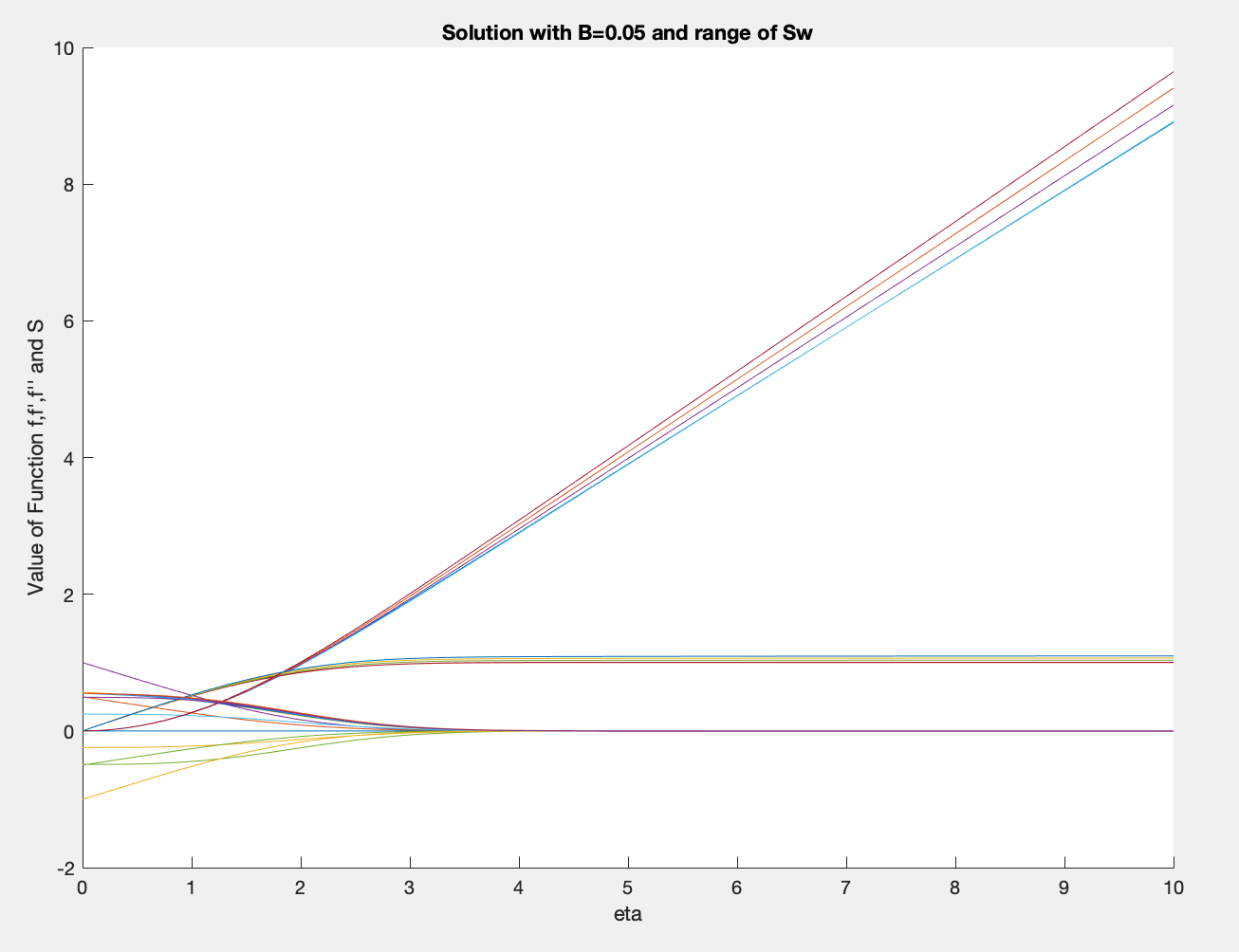
****

Figure 15 Results for fixed B and a range of Sw values

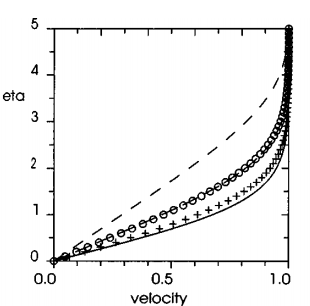
If the velocity profile is studied (Appendix E), it can be see that changing the value of Sw doesn’t have a large impact on the development of the boundary layer as the gradients of the lines for all cases are very similar. It can be seen that at larger, negative Sw values, the final velocity (U/Uinf) was larger- this may be due to the tolerance error allowed in the code and not the physical difference between the flows. If the code tolerance is ignored, it can be seen that the negative values of S allow for a larger boundary layer to develop as the value of f( is higher at lerge negative values of Sw. If this is interpreted physically, this means that at a lower imposed flow enthalpy/cooling wall, that the boundary layer development is laminar for a larger region.

Figure 16 Published results of F' and eta for varying values of the flow enthalpy (Van Oudheusden, 2004)

These can be compared to published results (shown left). In the figure below, it can be seen that the results show a smoother development of the boundary layer at lower Sw values (dashed line) than at higher values- these figures have transposed axis. The obtained results from calculation do not show a consistent value of f’(, whereas the published results do. The difference may be due to the allowed error in the tolerance of the calculation.

# Conclusions

To improve the results of the study, the Prandlt number should not be assumed to be constant (which it was in this investigation). The method used for the searching of the solution should also be improved- the method used was shooting and bisection which was very suseptable to error with initial guess values. To improve the results, a Runge-Kutta algorithm would be more efficient and the use of an integration function to find the first guess values would reduce the number of iterations required. The specified tolerance could also be reduced with increased computational time allowance to ensure the error in the result was minimal.

# Bibliography

Asaithambi, A., 1998. A finite-difference method for the Falkner-Skan equation. *Applied Mathematics and Computation,* 92(2), pp. 135-141.

Hartree, D., 1937. On an equation occurring in Falkner and Skan's approximate treatment of the equations of the boundary layer. *Mathematical Proceedings of the Cambridge Philosophical Society,* 33(2), pp. 223-239.

Javed, Y. & Ali Shah, S., 2017. *Experimental Findings for Velocity Profiles and Boundary Layer Thickness of Blasius Flow.* Islamabad, Aerospace Science and Engineering.

Liao, S.-J., 1999. A uniformly valid analytical solution of two-dimensional viscous flow over a semi-infinite flat plate. *Fluid Mechanics,* 385(1), pp. 101-128.

Parveen, S., 2014. Numerical solution of the Falkner Skan equation by using shooting techniques. *ISOR Journal of Mathematics,* 10(6), pp. 78-83.

Stewartson, K., 1953. Further Soltutions of the Falkner-Skan Equations. *Mathematical Transactions of the Cambridge Philosophical Society,* 50(3), pp. 454-465.

Van Oudheusden, B., 1997. A complete Crocco integral for two-dimensional laminar boundary layer flow over an adiabatic wall for Prandtl numbers near unity. *Journal of Fluid Mechanics,* 353(1), pp. 313-330.

Van Oudheusden, B., 2004. Compressibility Effects on the Extended Crocco Relation and the Thermal Recovery Factor in Laminar Boundary Layer Flow. *Journal of Fluid Engineering,* 126(1), pp. 32-41.

Weyburne, D., 2006. A Mathematical Description of the Fluid Boundary Layer. *Applied Mathematics and Computation,* 175(2), pp. 1675-1684.

# Appendix

**Appendix A- Code**

Main code:

%%

%ODE solver for Blausius equns (05/12/19 . V1.5 Laura Jones date: 20/12/19

%%

guess\_1 = 0.1;

guess\_2 = 0.95; %make sufficiently large to capture above answer

guess\_3 = -1; %these get removed for section 1 and 2

guess\_4 = 0.5;

tol = 0.001; %tolerance of the answer- set as desired

error = 1; %set guess1 error

s\_error = 1; %set to zero for incompressible

count = 1; %count iterations

Sw =1; %comment out for section 1 and 2

v\_w = 0; %values of Vw set to 0 for no-transpiration

%%

%conditional while loop

while error > tol && s\_error >tol %while error is bigger than specified tolerance, do this

[eta,y]=ode45(@ode,[0 10],[v\_w;0; guess\_1]); %guess\_3 ; Sw]); %use ode45 to solve system using ode function created

[eta\_2,y\_2]=ode45(@ode,[0 10],[v\_w; 0;guess\_2]); % guess\_4; Sw]); %same as above

tau\_1 = y(end,2); %extract last value eta= inf

tau\_2 = y\_2(end,2); %same as above with guess 2

S\_1 = y(end,5);

S\_2 = y\_2(end,5);

error\_1 = abs(1-tau\_1); %find error from known value of f''(inf) = 0

error\_2 = abs(1-tau\_2); %same as above for second guess

error\_3= abs(S\_1); %comment out for section 1 and 2

error\_4 = abs(S\_2); %comment out for section 1 and 2

midpoint = 0.5\*(guess\_1 + guess\_2); %bisection to find next guess

midpoint\_s = 0.5\*(guess\_3 + guess\_4);%comment out for section 1 and 2

if error\_1 > error\_2 %if guess 2 was closer, make this into new guess 1

tau\_1 = tau\_2; %make better guess the one to use

tau= y\_2(:,2);

error = error\_2; %transfer error over

guess\_1= guess\_2;

guess\_2 = midpoint;

elseif error\_1< error\_2 %if guess 1 was better, then the error for the loop is defined by guess 1

error = error\_1;

tau=y(:,2);

guess\_2 = midpoint;

end

%include this loop for section 3, otherwise comment out

if error\_3 > error\_4

S\_1 = S\_2;

s\_error = error\_4;

guess\_3 = guess\_4;

guess\_4 = midpoint\_s;

elseif error\_3 < error\_4

s\_error = error\_3;

guess\_4 = midpoint\_s;

end

disp("current error " + error + " " + s\_error); %comment out if error convergence not needed

count = count +1;

disp("Current iteration " + count); %displays current iteration number

end%end of while loop

tau\_b = y(1,3);

%%

%plot results as the values change

hold on

plot(eta\_2,y\_2) %use one with smaller error

plot(eta,y)

title("Solution with B=0.05 and range of Vw ") %change as according

xlabel("eta")

ylabel("Value of Function f,f',f''") %include S and S' in 3rd section

legend('f','f-dash','F-dash-dash'); %change as according

%%

%post processing

%set up of variables

x=zeros(1001,11);

eta=zeros(1001,11);

U\_inf = 1;

L = 10;

mu = 1.789E-5;

rho = 1.225;

nu = mu/rho;

A = sqrt(nu/U\_inf);

h = 0.01;

%delta- finds bounday layer thickness

for i = 1:length(x)

delta(i) = 5\*sqrt(x(i))\*A;

end

plot(x, delta, 'color', 'black');

position = [1 5 10];

for j = 1:length(position)

y(j) = eta\*sqrt(position(j))\*A;

plot(y2+position(j), y{j})

end

%drag coeff - calculate Cf over plate cumulativly

y\_3=y(:,3);

for i = 1 : length(x)

tau\_x(i) = mu\*U\_inf\*y\_3(1)\*sqrt(U\_inf/2/nu/x(i));

end

for i = 1: length(x)

cfx(i) = tau\_x(i)\*2/rho/U\_inf^2;

% cfxt(i) = 0.664/sqrt(rho\*U\_inf\*x(i)/mu); %incompressible

end

%skin friction- calculate skin friction over plate

CF\_T = 0;

dx = x(2)-x(1);

for i = 2 : length(x)

CF\_T = CF\_T + 1/L\*cfx(i)\*dx;

end

CDF\_T = CF\_T\*2;

CF = 0;

for i = 2 : length(x)

CF(i) = 1/L\*cfx(i)\*dx;

CF(i) = CF(i-1)+CF(i);

CDF(i) = CF(i)\*2;

D(i) = 1/2\*rho\*U\_inf^2\*L\*CDF(i);

end

ODE solver:

function y = ode(eta,f)

% f() f = f(1) f' = f(2) f" = f(3)

b=0.05; %commnent out for section 1

y = zeros(3,1); %section 1 and 2

y = zeros(5,1); %Section 3 only

% y() is array of the derivatives of f()

y(1) = f(2);

y(2) = f(3);

y(3) = (-f(1)\*f(3)); %no transpiration

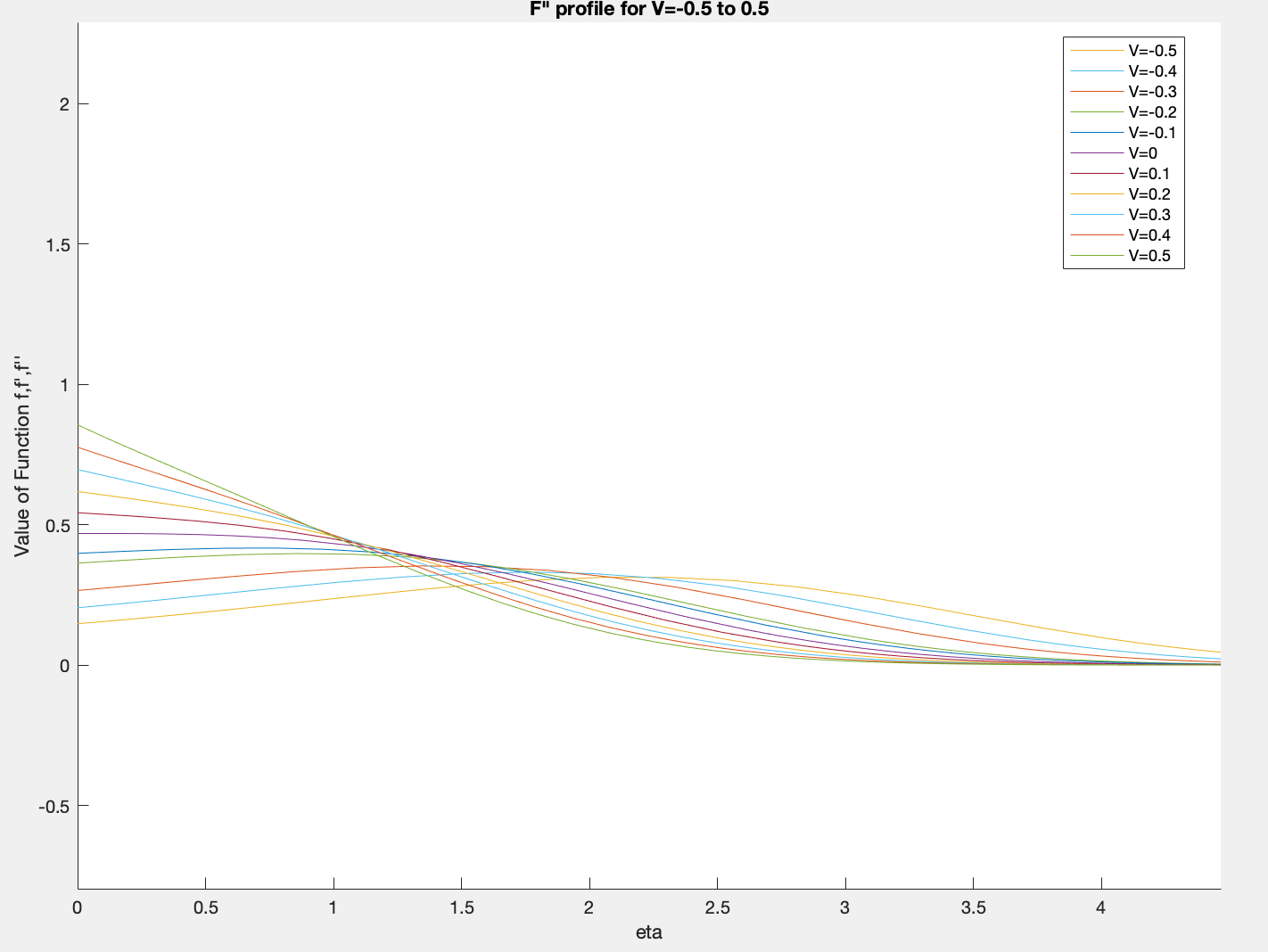
y(3) = (-f(1)\*f(3))-b\*(1-(f(2)\*f(2))); %section 2 with beta values- comment out as apropriate

y(3) = (-f(1)\*f(3))-b\*(1-(f(2)\*f(2)))-f(5)); %section 3 with Sw values, comment out as appropriate

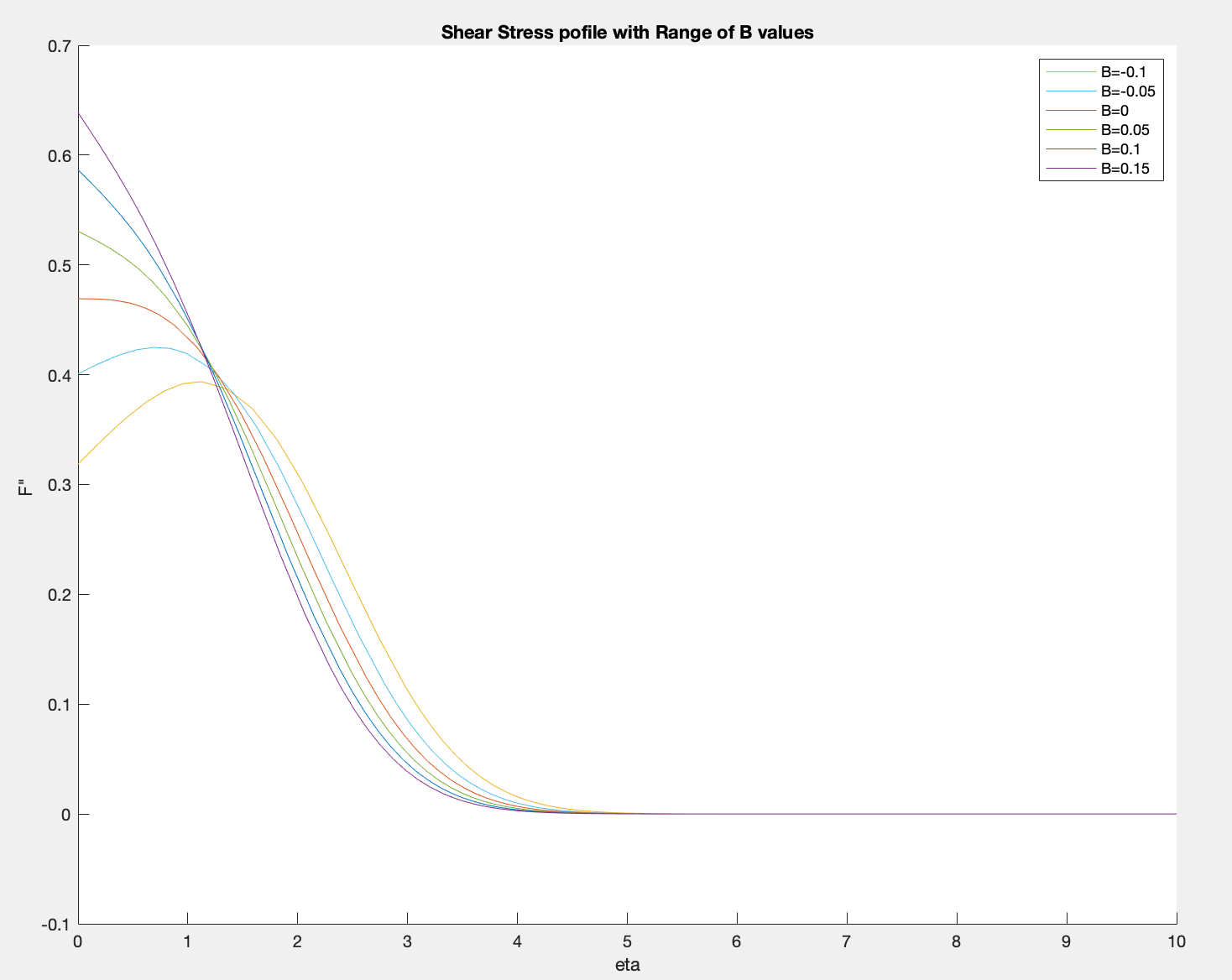
y(4) = f(4); %comment out for section 1 and 2

y(5) = -f(1)\*f(4); %commmet out for section 1 and 2

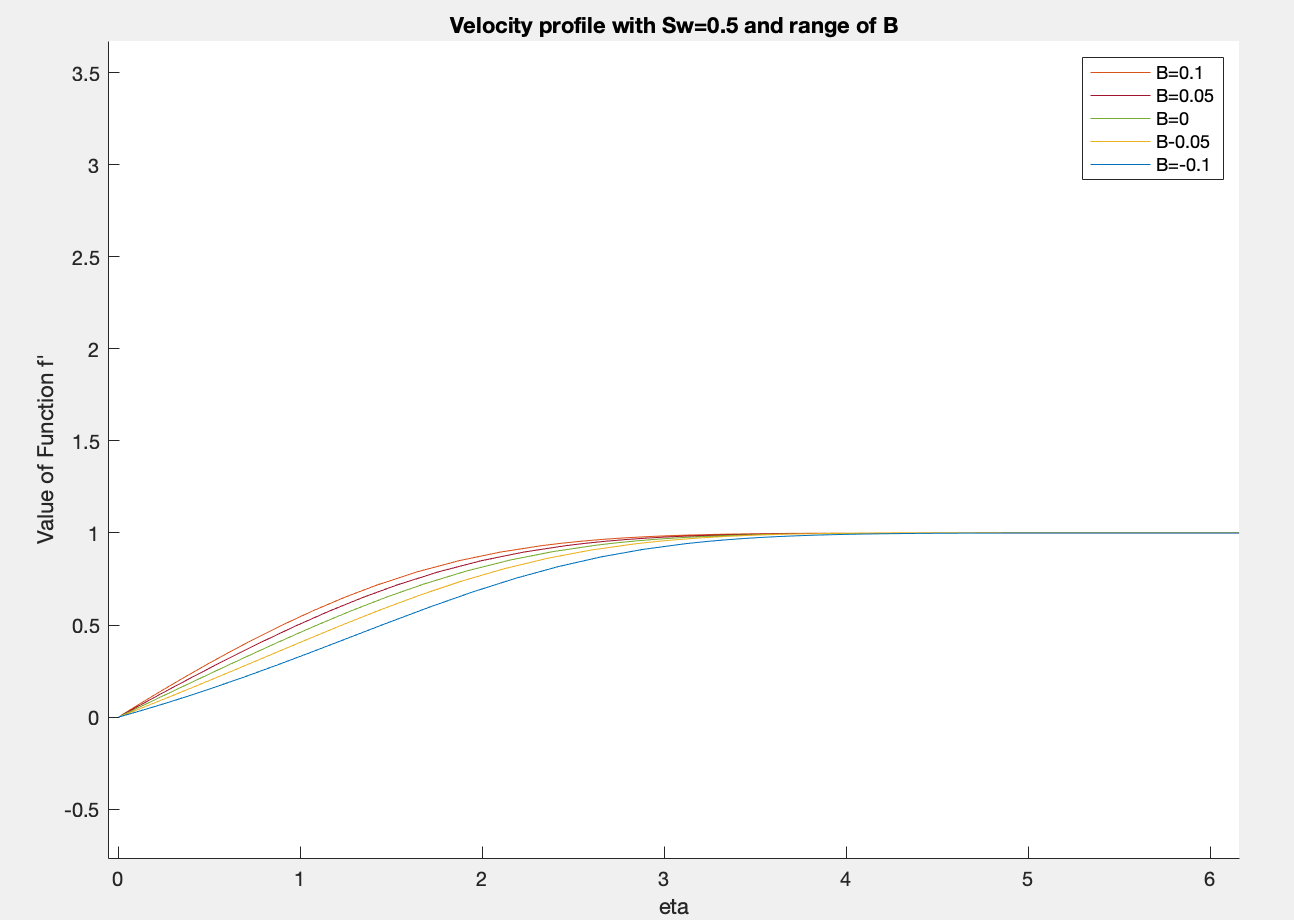
**Appendix B- Wall shear stress values for a range of Vw values.**

****

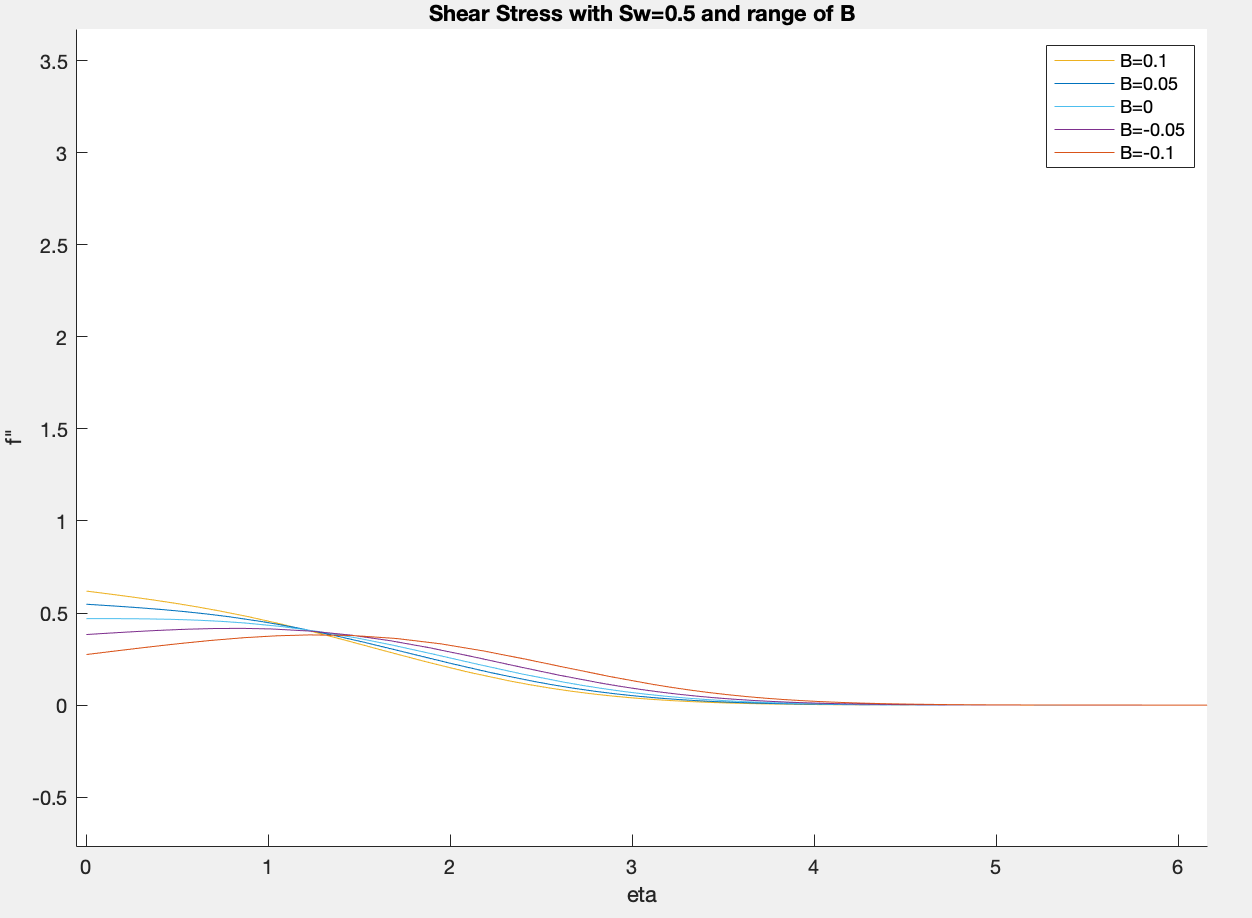
**Appendix C- Velocity profile of flow at range of Vw values**



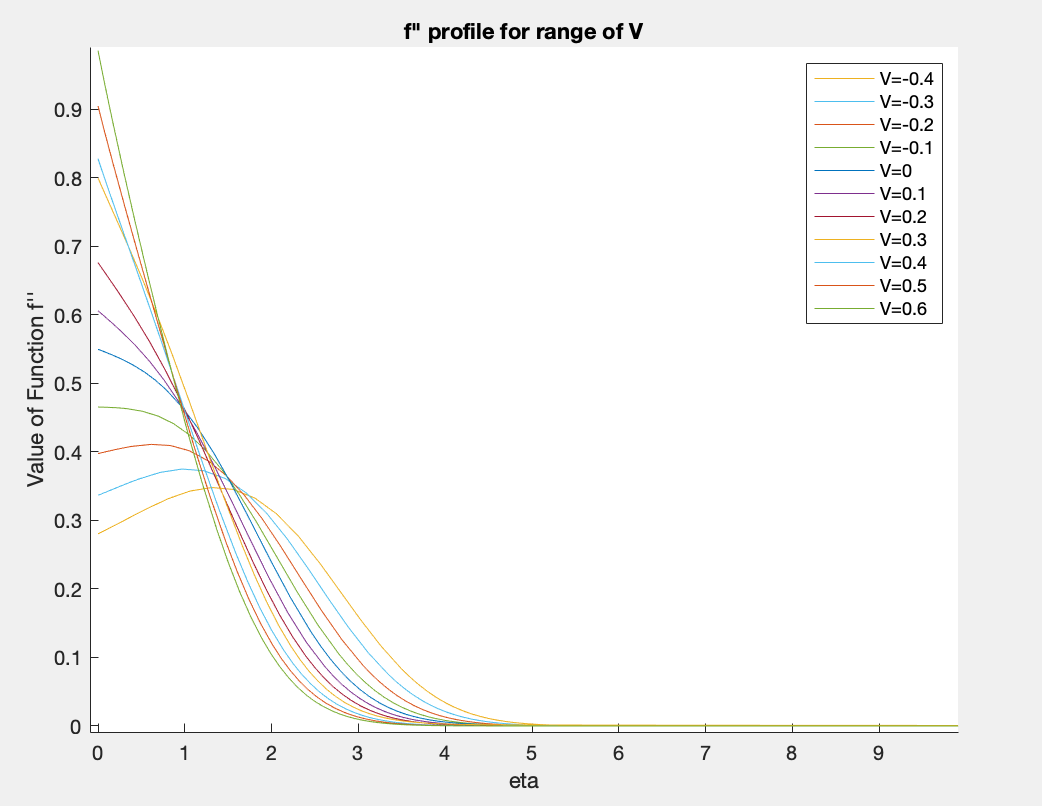
**Appendix D- Velocity profile for fixed value of Sw and range of B values**



**Appendix E- Shear stress profile for Sw=0.5 and range of B**



**Appendix F-**



**Appendix G- Velocity profile for fixed B and range of Sw**

