SA1: Interim Report 2

Freddie Ancliff (frja2) and Alex Oldham (amo57)

23rd May 2024

1 Exercise 1

Listing 1: ueintbit.m

```
function f = ueintbit(xa, ua, xb, ub)

%calculates the integral part of Eq10 in handout
%Thwaite's solution of the momentum integral equation
%Assumes a linear variation in ue

    %Calculate u_mean and delta x and u to simplify later
        equations
    u_mean = (ua+ub)*0.5;
    deltau = ub-ua;
    deltax = xb-xa;

    f = (u_mean^5 + 5/6 *u_mean^3*deltau^2 + 1/16 *u_mean
       *deltau^4)*deltax;

end
```

Listing 2: Exercise 1

```
clear
close all

%initialise basic parameters
ReL = 1e3;
npoints = 101;
x = linspace(0,1,npoints); %actually x/L
ue = ones(1,npoints); %actually ue/U
theta = zeros(1,npoints); %actually theta/L

I10 = 0; %integral part of equation 10 in handout
```

```
for j = 2:npoints %sum contributions to integral in equation 10 over all points
     intbit = ueintbit(x(j-1),ue(j-1),x(j),ue(j));
     I10 = I10 + intbit;
     theta(j) = sqrt(I10*0.45/ReL/(ue(j)^6)); %calculate mom thickness at each point
end

theta_blasius = 0.664/(ReL^0.5) * x.^0.5; %Blasius solution for momentum thickness

figure("Name", "Momentum Thickness ue=U");
plot(x,theta,"-", "Color","r","LineWidth",1)
hold on
plot(x,theta_blasius, "-", "Color","[0 0 1]","LineWidth",1)
hold off
set(gca,'Fontn','Times','FontSize',12,'linewidth',1)
xlabel('x/L')
ylabel('\theta/L')
legend("Thwaites","Blasius","Location","northwest")
%print -deps2c fig_ex1_mom_thickness.eps
```

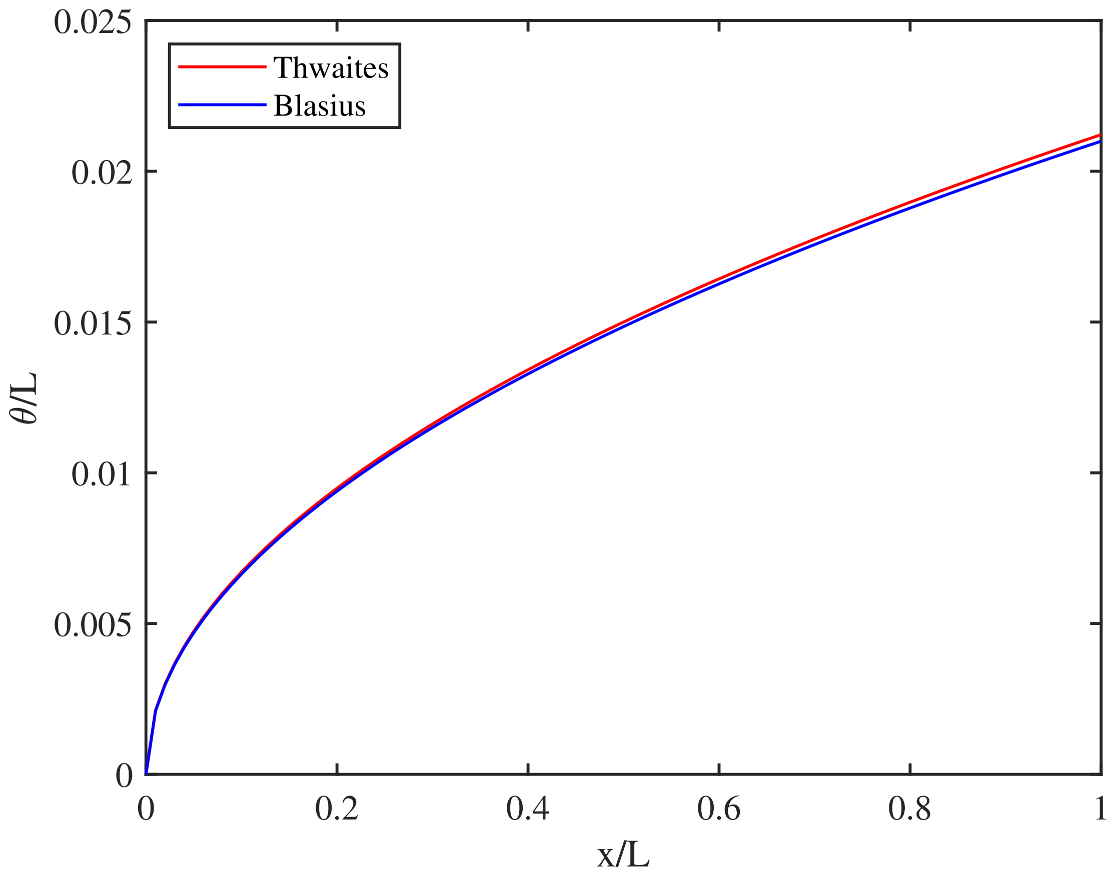

Figure 1: The momentum thickness, θ of a laminar, zero pressure gradient boundary layer, from Thwaites' method and the exact Blasius solution

2 Exercise 2

Listing 3: Exercise 2

```
clear
close all

%initialise basic parameters
ReL = 2e7;    %[5e6, 1e7, 2e7]
%velocity gradient of d(ue/U)/d(x/L)
vel_grad = 0.1;  %[-0.1, 0.0, 0.1]
npoints = 101;
x = linspace(0,1,npoints); %actually x/L
ue = linspace(1,1+vel_grad,npoints); %acctually ue/U
```

```
theta = zeros(1,npoints);
laminar = true; % initializes boundary layer state flag
i = 1;
I10 = 0; %integral part of eq10 in handout


while laminar && i < npoints
    i = i + 1;
    intbit = ueintbit(x(i-1),ue(i-1),x(i),ue(i));
    I10 = I10 + intbit;
    theta(i) = sqrt(I10*0.45/ReL/((ue(i))^6));

    m = -ReL*(theta(i))^2*vel_grad;
    He = laminar_He(thwaites_lookup(m));
    Rethet = ReL*ue(i)*theta(i);

    %test for laminar to turbulent transition
    if log(Rethet) >= 18.4*He-21.74
        laminar = false;
        disp([x(i) Rethet/1000])
    end
end
```

2.1 Transition locations and Re_θ values

Re_L [x10^6]	$\frac{d(u_e/U)}{d(x/L)}$	Transition Location $\frac{x}{L}$	Re_θ [x10^3] at transition
5	-0.1	0.48	1.08
	0.0	0.74	1.29
	0.1	-	-
10	-0.1	0.29	1.17
	0.0	0.37	1.29
	0.1	0.55	1.51
20	-0.1	0.17	1.25
	0.0	0.19	1.31
	0.1	0.22	1.38

Table 1: Laminar to turbulent transition locations

3 Exercise 3

Listing 4: Exercise 3

```
clear
```

```
close all

%initialise basic parameters
ReL = 8.957e5;
%velocity gradient of d(ue/U)/d(x/L)
vel_grad = -0.25;  %-0.25 for this case
npoints = 101;
x = linspace(0,1,npoints); %actually x/L
ue = linspace(1,1+vel_grad,npoints); %acctually ue/U
theta = zeros(1,npoints); %acctually theta/L
laminar = true; % initializes boundary layer state flag
i = 1;

int = 0; %natural transition point
ils = 0; %Laminar separation

I10 = 0; %equation 10 in handout
while laminar && i < npoints
    i = i + 1;
    intbit = ueintbit(x(i-1),ue(i-1),x(i),ue(i));
    I10 = I10 + intbit;
    theta(i) = sqrt(I10*0.45/ReL/(ue(i)^6));

    m = -ReL*(theta(i))^2*vel_grad;
    He = laminar_He(thwaites_lookup(m));
    Rethet = ReL*ue(i)*theta(i);

    %test for laminar to turbulent transition
    if log(Rethet) >= 18.4*He-21.74
        laminar = false;
        %disp([x(i) (Rethet/1000)])
        int = i;
    elseif m >= 0.09 %test for laminar separation
        laminar = false;
        ils = i;
    end
end

if int ~= 0
    disp(['Natural transition at ' num2str(x(int)) ' with
        Rethet ' num2str(Rethet)])
end
if ils ~= 0
    disp(['Laminar seperation at ' num2str(x(ils)) ' with
        Rethet ' num2str(Rethet)])
end
```

3.1 Laminar separation locations

Re_L	$\frac{d(u_e/U)}{d(x/L)}$	Separation Location $\frac{x}{L}$
10^3	-0.25	0.50
10^4	-0.25	0.50
10^5	-0.25	0.50

Table 2: Laminar separation locations

Transition supplants separation at Re_L = 9.0x10^5, with $\frac{d(u_e/U)}{d(x/L)}$ = -0.25.

4 Exercise 4

Listing 5: thickdash.m

```
function dthickdx = thickdash(xmx0, thick)
    %global variables used by thickdash
    global ReL ue0 vel_grad %#ok<GVMIS>

    theta = thick(1);
    de = thick(2);

    ue = ue0 + vel_grad*xmx0;

    Retheta = ReL*ue*theta;

    He = de/theta;

    if He >= 1.46
        H = (11*He+15)/(48*He-59);
    else
        H = 2.803;
    end

    c_f = 0.091416*((H-1)*Retheta)^-0.232*exp(-1.26*H);
    c_diss = 0.010024*((H-1)*Retheta)^-(1/6);

    dthickdx = [(c_f/2 - (H+2)/ue*vel_grad*theta); (
        c_diss - 3/ue*vel_grad*de)];
```

Listing 6: Exercise 4

```
clear
close all
```

```
%global variables used by thickdash
global ReL ue0 vel_grad %#ok<GVMIS>

ue0 = 1.0;
vel_grad = 0.0;
ReL = 1e7;

%initialise turbulent boundary layer at first point
x0 = 0.01;
thick0(1) = 0.023*x0*(ReL*x0)^(-1/6);
thick0(2) = 1.83*thick0(1);

[delx, thickhist] = ode45(@thickdash, [0 0.99], thick0);
%solve ODE (15) from handout up to x=L

x = x0 + delx;

theta7 = 0.037*x.*(ReL*x).^(-1/5); %1/7th power law
    approximation

theta9 = 0.023*x.*(ReL*x).^(-1/6); %1/9th power law
    approximation

figure("Name", "Comparisons of Theta");
plot(x,thickhist(:,1),"-", "Color","r","LineWidth",1)
hold on
plot(x,theta7, "-", "Color","[0 0.5 0]","LineWidth",1)
plot(x,theta9, "-", "Color","[0 0 1]","LineWidth",1)
hold off
set(gca,'Fontn','Times','FontSize',12,'linewidth',1)
xlabel('x/L')
ylabel('\theta/L')
legend("\theta/L calculated","1/7th power estimate","1/9
    th power estimate","Location","northwest")
%print -deps2c fig_ex4_mom_thickness_3methods.eps
```

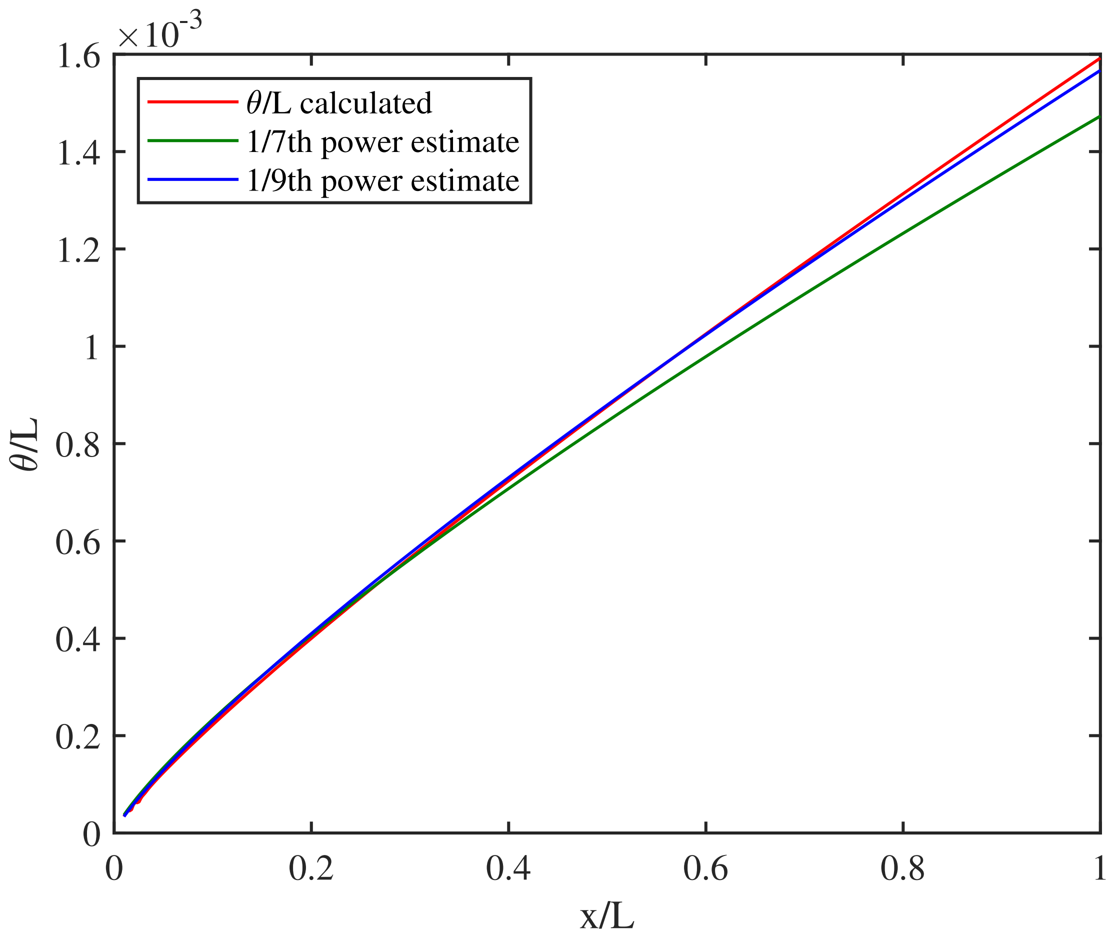

Figure 2: Momentum thickness, zero pressure gradient turbulent boundary layer

5 Exercise 5

Listing 7: Exercise 5

```
clear
close all
%global variables used by thickdash
global ReL ue0 vel_grad %#ok<GVMIS>

ue0 = 1.0;
vel_grad = -0.5;
ReL = [1e6, 1e7, 1e8];
%}
```

```
ue0 = 1.0;
vel_grad = -0.5;
ReL = 1e7;

%initialise turbulent boundary layer at first point
x0 = 0.01;
thick0(1) = 0.023*x0*(ReL*x0)^(-1/6);
thick0(2) = 1.83*thick0(1);

[delx, thickhist] = ode45(@thickdash, [0 0.99], thick0);
%solve ODE (15) from handout up to x=L

x = x0 + delx;

theta7 = 0.037*x.*(ReL*x).^(-1/5); %1/7th power law approximation

theta9 = 0.023*x.*(ReL*x).^(-1/6); %1/9th power law approximation


He = thickhist(:, 2)./thickhist(:, 1);
HeSep = 1.46.*ones(1,length(delx)); %turbulent separation below He=1.46

seploc = 0;
%check for turbulent separation
for i = 30:length(delx) %ignore numerical errors in first few points
%use plot below to decide size of loop
    if He(i-1)>1.46 && He(i)<1.46
        seploc = x(i);
        %could do linear interpolation for extra accuracy
        %less effect as number of points is increased
        break %no point checking further downstream values
    end
end

if seploc ~= 0
    seploc %#ok<NOPTS>
end

figure("Name", "H_e vs x/L");
plot(x,He,"-", "Color","r","LineWidth",1)
```

```
hold on
plot(x,HeSep,"-", "Color", "[0 0 1]","LineWidth",1)
hold off
set(gca,'Fontn','Times','FontSize',12,'linewidth',1)
xlabel('x/L')
ylabel('H_e')


figure("Name", "Comparisons of Theta and d_e");
plot(x,thickhist(:,1),"-", "Color","r","LineWidth",1)
hold on
plot(x,thickhist(:,2), "-", "Color","[0 0.5 0]","
    LineWidth",1)
hold off
set(gca,'Fontn','Times','FontSize',12,'linewidth',1)
xlabel('x/L')
ylabel('\theta/L,~\delta_e/L')
legend("\theta/L","\delta_e/L","Location","northwest")
%print -deps2c fig_ex5_theta_delta_e.eps
```

5.1 Turbulent separation locations

Re_L	$\frac{d(u_e/U)}{d(x/L)}$	Separation Location $\frac{x}{L}$
10^6	-0.5	0.895
10^7	-0.5	0.989
10^8	-0.5	-
10^7	-0.25	-
10^7	-0.95	0.523

Table 3: Turbulent separation locations

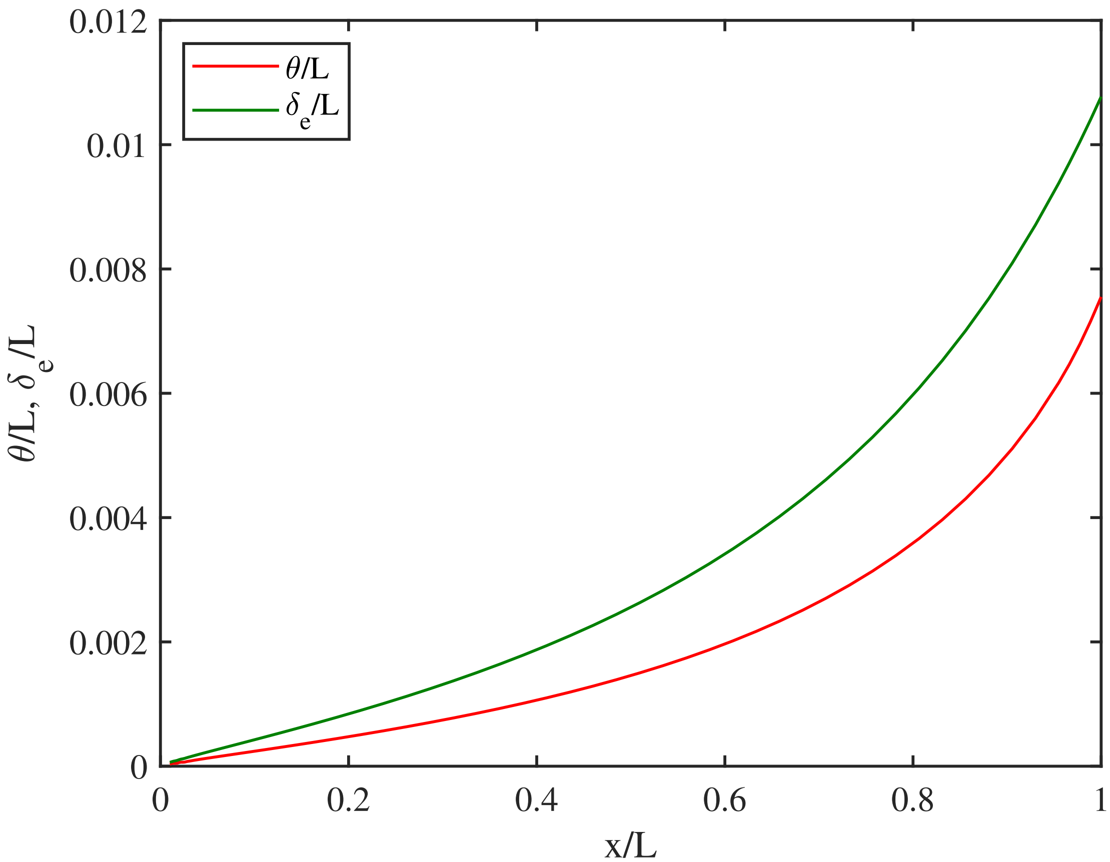

Figure 3: $\frac{\theta}{L}$ and $\frac{\delta}{L}$ against $\frac{x}{L}$, for a turbulent boundary layer at $Re_L = 10^7$, and $\frac{d(u_e/U)}{d(x/L)} =$ -0.5.

6 Exercise 6

6.1 Flowchart

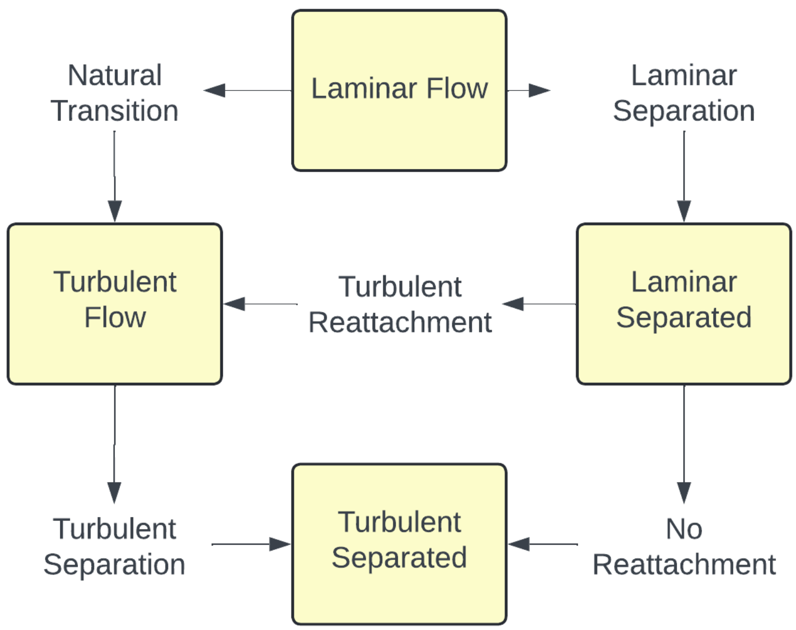

Figure 4: Boundary layer evolution flowchart

6.2 Script

Listing 8: Exercise 6

```
clear
close all

npoints = 101;

global ReL ue0 vel_grad %#ok<GVMIS>
%initialise basic parameters
ReL = 1e5;
%velocity gradient of d(ue/U)/d(x/L)
vel_grad = -0.381;

x = linspace(0,1,npoints); %actually x/L
```

```
ue = linspace(1,1+vel_grad,npoints); %acctually ue/U
theta = zeros(1,npoints); %acctually theta/L
He = 1.57258*ones(1,npoints);

laminar = true; % initializes boundary layer state flag

i = 1;

int = 0; %natural transition point
ils = 0; %Laminar separation
itr = 0; %Turbulent reattachement
its = 0; %Turbulent separation

I10 = 0; %equation 10 in handout

while laminar && i < npoints
    i = i + 1;
    intbit = ueintbit(x(i-1),ue(i-1),x(i),ue(i));
    I10 = I10 + intbit;
    theta(i) = sqrt(I10*0.45/ReL/(ue(i)^6));

    m = -ReL*(theta(i))^2*vel_grad;
    He(i) = laminar_He(thwaites_lookup(m));

    Rethet = ReL*ue(i)*theta(i);

    %test for turbulence
    if log(Rethet) >= 18.4*He(i)-21.74
        laminar = false;
        %disp([x(i) (Rethet/1000)])
        int = i;
    elseif m >= 0.09
        laminar = false;
        ils = i;
        He(i) = 1.51509;
    end
end
de = He(i)*theta(i);

while its == 0 && i < npoints

    thick0 = [theta(i), de];
    ue0 = ue(i);

    i = i + 1;
```

```
    [delx, thickhist] = ode45(@thickdash,[0,x(i)-x(i-1)],
        thick0);

    theta(i) = thickhist(end, 1);
    de = thickhist(end, 2);

    He(i) = de/theta(i);

    %disp(He(i))
    if He(i) >= 1.58 && ils ~= 0 && itr == 0 % Turbulent
        reattachent
         itr = i;

    elseif He(i) < 1.46 % Turbulent seperation
         its = i;
    end
end

He(i:npoints) = He(i);

while its ~= 0 && i < npoints
    i = i+1;
    theta(i) = theta(its)*(ue(its)/ue(i))^(2.803+2);
end

if int ~= 0
    disp(['Natural transition at ' num2str(x(int)) ' with
        Rethet ' num2str(ReL*ue(int)*theta(int))])
end
if ils ~= 0
    disp(['Laminar seperation at ' num2str(x(ils)) ' with
        Rethet ' num2str(ReL*ue(ils)*theta(ils))])
end
if itr ~= 0
    disp(['Turbuent Reattachment at ' num2str(x(itr)) '
        with Rethet ' num2str(ReL*ue(itr)*theta(itr))])
end
if its ~= 0
    disp(['Turbulent seperation at ' num2str(x(its)) '
        with Rethet ' num2str(ReL*ue(its)*theta(its))])
end


%save 6_025.mat -v7.3 %save data to a relevant file

%example plotting
```

```
figure("Name", "\theta/L vs x/L");
plot(x,theta,"-", "Color","r","LineWidth",1)
set(gca,'Fontn','Times','FontSize',12,'linewidth',1)
xlabel('x/L')
ylabel('\theta/L')

figure("Name", "H_e vs x/L");
plot(x,He,"-", "Color","r","LineWidth",1)
set(gca,'Fontn','Times','FontSize',12,'linewidth',1)
xlabel('x/L')
ylabel('H_e')
```

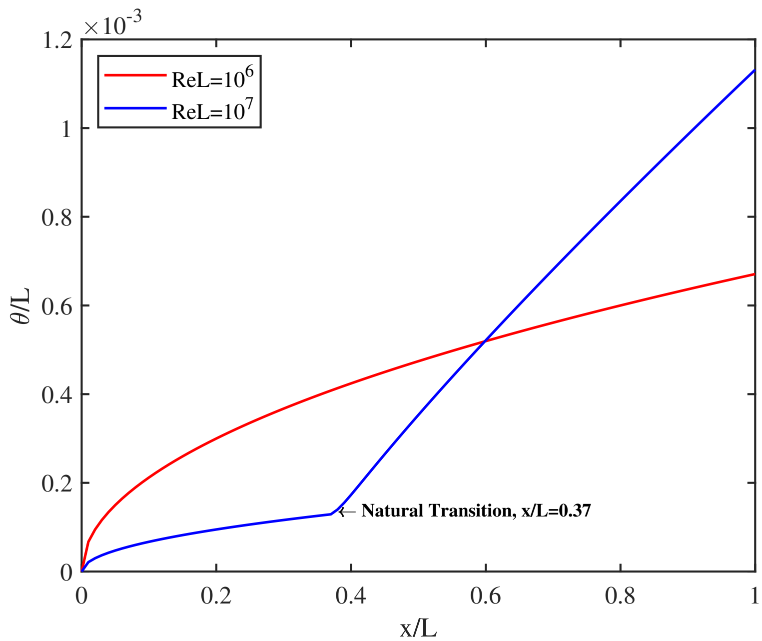

Figure 5: $\frac{\theta}{L}$ against $\frac{x}{L}$, for $\frac{d(u_e/U)}{d(x/L)} = 0$

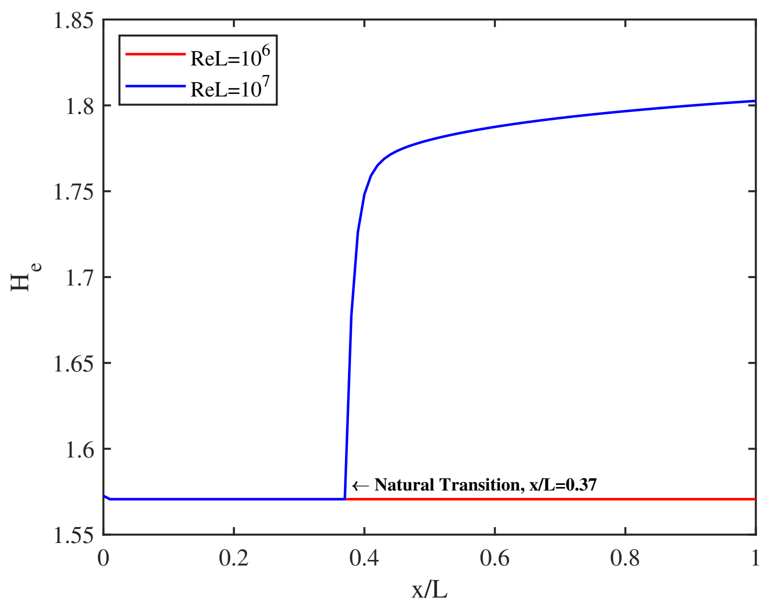

Figure 6: H_e against $\frac{x}{L}$, for $\frac{d(u_e/U)}{d(x/L)} = 0$

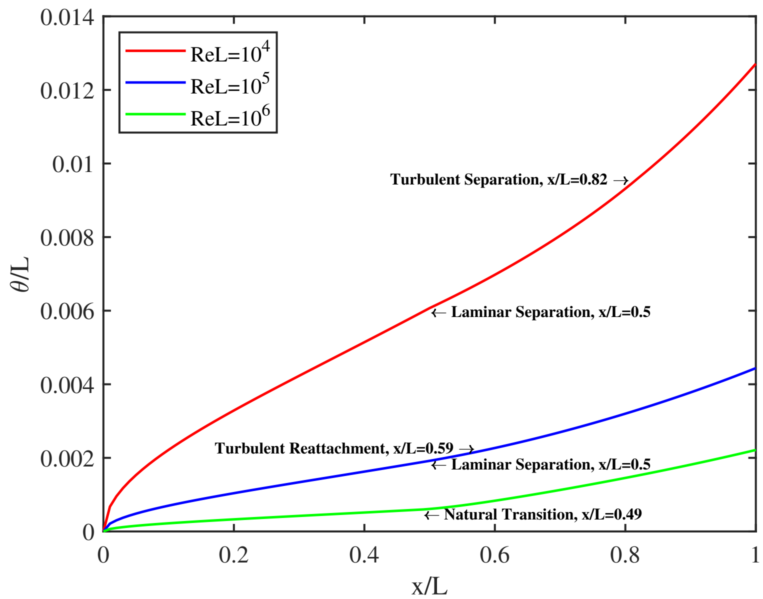

Figure 7: $\frac{\theta}{L}$ against $\frac{x}{L}$, for $\frac{d(u_e/U)}{d(x/L)}$ = -0.25

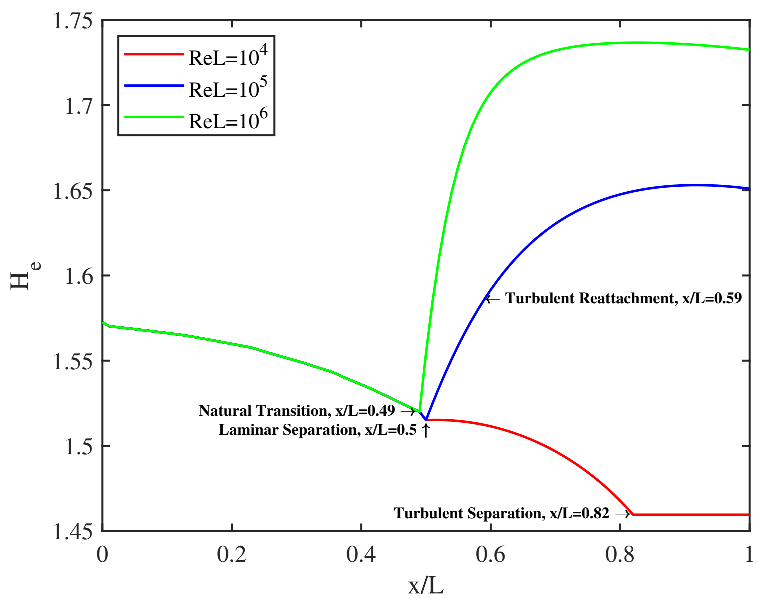

Figure 8: H_e against $\frac{x}{L}$, for $\frac{d(u_e/U)}{d(x/L)}$ = -0.25

6.3 Critical Velocity Gradient

At $Re_L = 10^5$, $\frac{d(u_e/U)}{d(x/L)}$ = -0.38 gives turbulent separation at x=L.