Cairo University
Faculty of Engineering
Aerospace Engineering Dept.

Report (2)
AER 3120 - Aerodynamics (Boundary Layer)
3^{rd} Year, 2^{nd} Semester 2021/2022

Local Skin friction coefficient and wall sheer stress calculations using Pohlhausen Method

By:

Mohammad Khaled Gamal Ali
Sec:2, BN:14

Submitted to:

Dr Hesham M. Elbanna

Contents

Abstract

In the last part of the project, we calculated the pressure coefficient for a given number of airfoils [NACA0006, NACA0012, NACA0018] at $U_\infty = 50\frac{m}{s},\ \alpha = 0^\circ$ using a source vortex panel method.

In this part we will calculate the local skin friction coefficient $C_{fx}$and the wall shear stress τ_w for 2 cases:

- Assuming all laminar flow across the airfoil
- Accounting for transition

Using the Pohlhausen method in addition to the smith model to account for the transition

Introduction

One of the earliest and, until recently, most widely used approximate methods for the solution of the boundary layer equation is that developed by Pohlhausen. This method is based on the momentum equation of Kármán, which is obtained by integrating the boundary layer equation across the layer. In the case of steady flow over an impermeable surface, the momentum equation reduces to

$$\frac{\tau_\omega}{\rho U^2} = \frac{d}{dx}\delta_2 + \frac{2\delta_2 + \delta_1}{U}\frac{dU}{dx}$$

Where $\tau_w = \mu\left(\frac{\partial u}{\partial y}\right)|_{y=0}$ is the skin friction, $\delta_1 = \int_0^\infty (1-\bar{u})dy$ is the displacement thickness, and $\delta_2 = \int_0^\infty \bar{u}(1-\bar{u})dy$ is the momentum thickness of the boundary layer. The boundary conditions for the boundary layer equations are $u = v = 0\ at\ y = 0, u \to U(x)\ as\ y \to \infty$

In the Pohlhausen's method, and similar approximate methods, a form for the velocity profile u(x,y) is sought which satisfies the momentum equation and some of the boundary condition to .It is hoped that this form will approximate to the exact profile, which satisfies all the conditions to as well as.

Pohlhausen's method Functions

$$Z_{i+1} = Z_i + \left[\frac{\mathrm{F}(\Lambda)}{U}\right]_i \Delta x \qquad [1]$$

$$K(\Lambda) = \left(\frac{37}{315} - \frac{1}{945}\Lambda - \frac{1}{9072}\Lambda^2\right)^2 \Lambda \qquad [2]$$

$$\mathrm{f}_1(\Lambda) \equiv \frac{\delta_1}{\delta_2} = \frac{\frac{3}{10} - \frac{\Lambda}{120}}{\frac{37}{315} - \frac{1}{945}\Lambda - \frac{1}{9072}\Lambda^2} \qquad [3]$$

$$\mathrm{f}_2(\Lambda) \equiv \frac{\tau_w \delta_2}{\mu_\infty U} = \left(2 + \frac{\Lambda}{6}\right)\left(\frac{37}{315} - \frac{1}{945}\Lambda - \frac{1}{9072}\Lambda^2\right) \qquad [4]$$

$$\mathrm{F}(\Lambda) = 2\mathrm{f}_2(\Lambda) - 4K(\Lambda) - 2K(\Lambda)\mathrm{f}_1(\Lambda) \qquad [5]$$

Methodology

We need to calculate these parameters for the NACA0012 at $\alpha = 0°$

Boundary Layer (Disturbance) Thickness(δ).

Momentum (Deficiency) Thickness (δ_2).

Displacement Thickness (δ_1).

(Local) Skin Friction (Stress) Coefficient (C_{fx}).

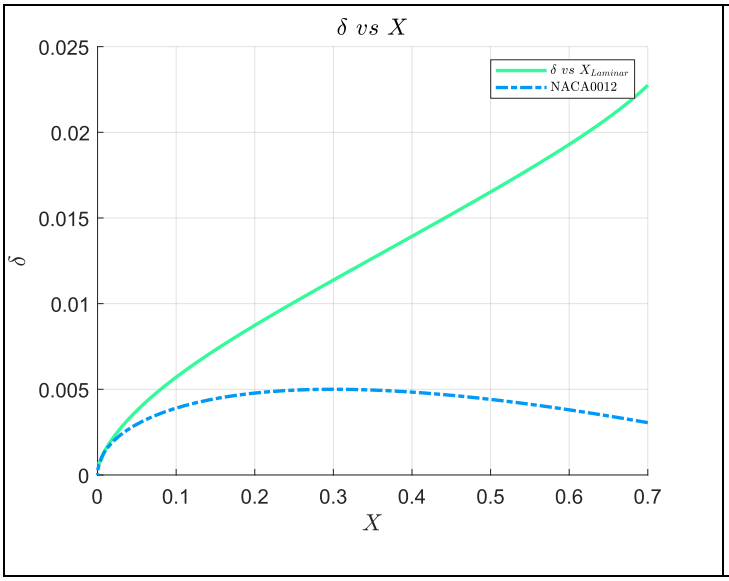
Shear Stress at the Wall (τ_w).

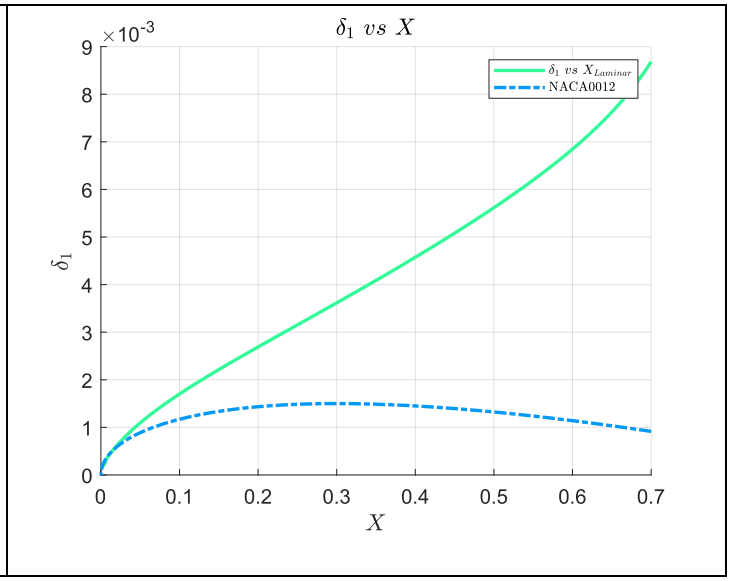
Solution Steps:

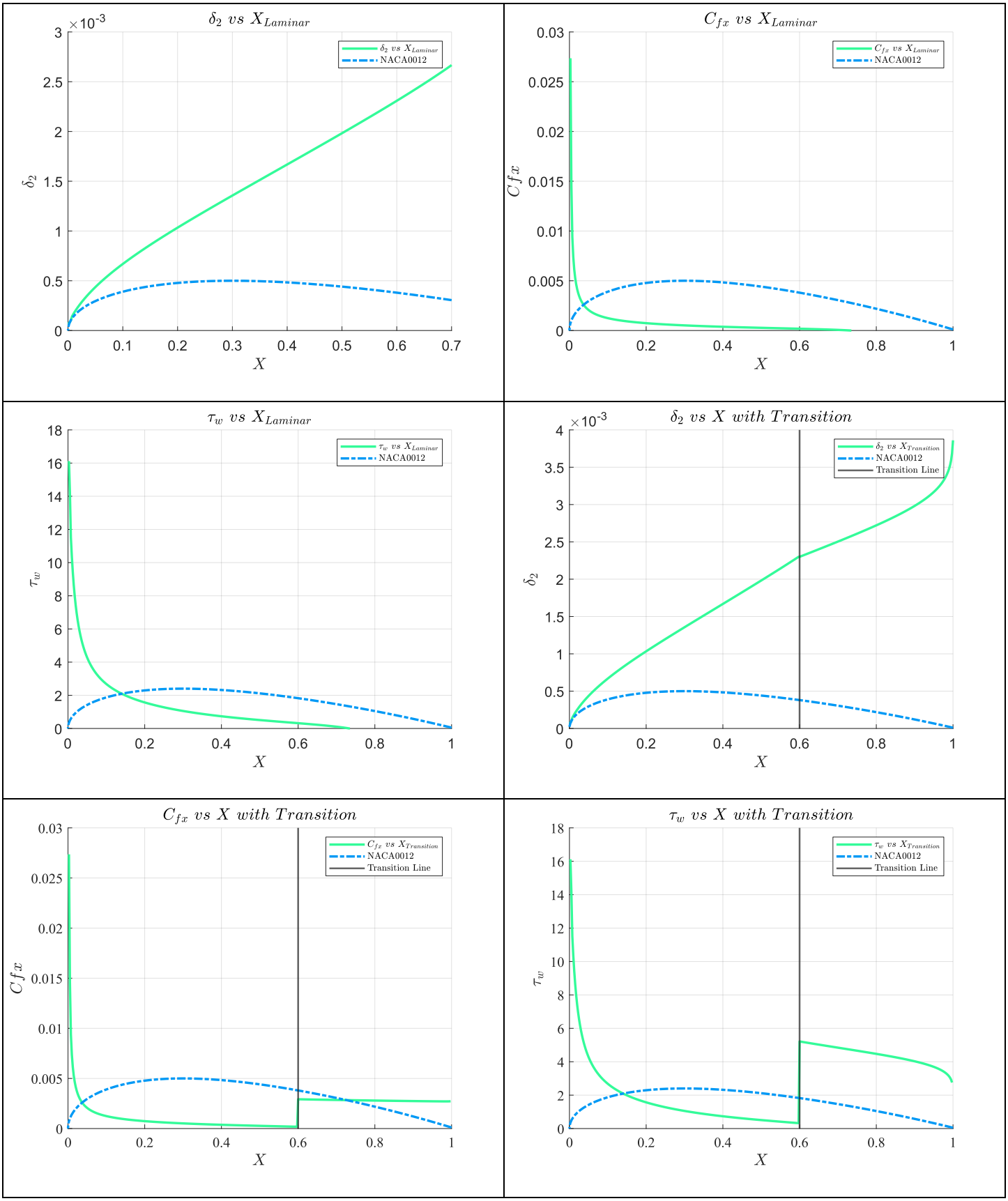
- obtain the velocity distribution for each section on the airfoil surface using the "Panel method"
- obtain U', U'' of the velocity distribution
- start from the leading edge stagnation point where (U=0, x=0)
- $\Lambda_0 = 7.052$
- Get $\bar{\delta}_1 = \frac{3}{10} - \frac{\Lambda}{120}, \bar{\delta}_2 = \frac{37}{315} - \frac{1}{945}\Lambda - \frac{1}{9072}\Lambda^2$
- Get $K(\Lambda), \mathrm{f}_1(\Lambda), \mathrm{f}_2(\Lambda), \mathrm{F}(\Lambda)$
- Get $Z_0 = \frac{K_0}{U'_0}, \left[\frac{F(\Lambda)}{\bar{U}}\right]_0 = -0.0652857 U''_0 / U'^2_0$
- Get $Z1 = Z_0 + \left(\frac{F}{U}\right)_0 \Delta x$
- Get $\delta = \sqrt{\frac{\Lambda \mu_\infty}{U' \rho_\infty}}, \delta_1 = \delta\bar{\delta}_1, \delta_2 = \delta\bar{\delta}_2$
- Using Euler Forward Scheme to get the Z and therefore populate the table
- For transition consideration the Smith model will be used

Results

X	U	U'	Z	K	Λ	f_1	f_2	δ	δ_1	δ_2	C_{fx}	τ_w
0.1	59.3203872	5.055731076	1.53101968	0.00309617	0.225345565	2.54335126	0.238834879	0.005704987	0.001700783	0.000668717	0.001243485	2.680121975
0.2	58.890057	-9.23910122	3.65797006	-0.013518542	-0.964681014	2.60215194	0.217724202	0.008731707	0.002689706	0.001033647	0.000738723	1.569176614
0.3	57.7587327	-12.1793237	6.36997496	-0.031032795	-2.182440168	2.66835227	0.195115413	0.011438833	0.003639688	0.001364021	0.000511496	1.045161947
0.4	56.5303452	-12.8313883	9.52944043	-0.04891038	-3.408665916	2.74158798	0.171521141	0.01392764	0.004573914	0.001668345	0.000375612	0.735205612
0.5	55.2446432	-12.839084	13.4543234	-0.069096476	-4.798366384	2.83321989	0.144032672	0.016519687	0.005616469	0.001982362	0.00027163	0.507765837
0.6	53.960016	-12.9231039	18.2413222	-0.094293801	-6.586756024	2.9661119	0.107947387	0.026905411	0.006846487	0.002299825	0.002924006	0.319226741
0.7	52.5982022	-13.8873823	24.5217206	-0.136217003	-9.934950434	3.26910746	0.040300649	0.022855714	0.008748967	0.002509194	0.002852427	0.10019607
0.8	51.1159275	-17.1166277	0	0	0	0	0	0	0	0.00272135	0.002794014	NaN
0.9	48.9575885	-28.8153498	0	0	0	0	0	0	0	0.002982003	0.002742685	NaN
1	38.1694035	-979.821481	0	0	0	0	0	0	0	0.003860408	NaN	NaN






δ_2 vs $X_{Laminar}$
$\times 10^{-3}$
δ_2
X
δ_2 vs $X_{Laminar}$
NACA0012
C_{fx} vs $X_{Laminar}$
Cfx
X
C_{fx} vs $X_{Laminar}$
NACA0012
τ_w vs $X_{Laminar}$
τ_w
X
τ_w vs $X_{Laminar}$
NACA0012
δ_2 vs X with Transition
$\times 10^{-3}$
δ_2
X
δ_2 vs $X_{Transition}$
NACA0012
Transition Line
C_{fx} vs X with Transition
Cfx
X
C_{fx} vs $X_{Transition}$
NACA0012
Transition Line
τ_w vs X with Transition
τ_w
X
τ_w vs $X_{Transition}$
NACA0012
Transition Line

Appendix A [MATLAB Codes]

```
clc
clear
close all
```

Importing Airfoil Data

```
path="AirfoilData.xlsx";
path1="n0012-il.xlsx";
[X,U]=Import_Script(path);
[xfoil,yfoil]=Import_Script_airfoil(path1);
```

Givens

```
U_inf=50;
mu = 1.789*10^-5;
rho = 1.225;
```

Setting up The Pohlhausen Method Functions

$U', U'', \delta_1, \delta_2, \bar{\delta}_1, \bar{\delta}_2, K(\Lambda), f_1(\Lambda), f_2(\Lambda), Z$

```
U_d = zeros(1,length(U));
length_U = length(U);
%Velocity gradients
U_d=gradient(U,X);                                    %U'
U_dd=gradient(U_d,X);                                 %U"
%The Paulhausen Method Functions as Polynomials Coefficients
del1=[-120^-1 3/10];                                  %delta1 [1st degree poly]
del2=[-1/9072 -1/945 37/315];                         %delta2 [2nd degree poly]
k_p=conv(conv(del2,del2),[1 0]);                      %K(Lambda) [5th degree poly]
f2_p=conv([1/6 2],del2);                              %f2(Lambda) [3rd degree]
F_del2=(conv((2*[0 0 f2_p]-4*k_p),del2)-[0 conv(2*k_p,del1)]);      %F(Lambda)*delta2 [7th degree poly]
lamda_s=roots(F_del2);                                %Solving for Lambda
a=real(lamda_s)<12 & real(lamda_s)>-12;               %Physical limits for Lambda
ඞ ඞ ඞ ඞ ඞ ඞ ඞ ඞ ඞ ඞ ඞ ඞ ඞ ඞ ඞ ඞ ඞ ඞ ඞ ඞ ඞ ඞ ඞ ඞ ඞ ඞ ඞ ඞ ඞ ඞ ඞ ඞ ඞ ඞ ඞ ඞ ඞ ඞ
lamda_itr = zeros(1, length_U);
lamda_itr(1) = lamda_s(a);
delta1_ = zeros(1, length_U);
delta2_ = zeros(1, length_U);
delta1_(1) = 0.3-lamda_itr(1)/120;
delta2_(1) = 37/315-lamda_itr(1)/945-lamda_itr(1)^2/9072;
K = zeros(1, length_U);
K(1) = delta2_(1)^2*lamda_itr(1);
z = zeros(1, length_U);
z(1) = K(1)/U_d(1);
F = zeros(1, length_U);
f1 = zeros(1, length_U);
```

```
f2 = zeros(1, length_U);
f1(1) = delta1_(1)/delta2_(1);
f2(1) = (2+lamda_itr(1)/6)*delta2_(1);
F(1) = 2*f2(1)-4*K(1)-2*K(1)*f1(1);
F0_U0 = -0.0652*U_dd(1)/U_d(1)^2;
z(2) = z(1)+F0_U0*( X(2) - X(1) );
delta = zeros(1, length_U);
delta1 = zeros(1, length_U);
delta2 = zeros(1, length_U);
delta(1) = sqrt(lamda_itr(1)*mu/rho/U_d(1));
delta1(1) = delta(1)*delta1_(1);
delta2(1) = delta(1)*delta2_(1);
```

Populating The Pohlhausen Method Functions

```
for i = 2:length(U_d)
   K(i) = z(i)*U_d(i);
    P1=[0 0 0 0 0 -K(i)]+k_p;
    Lamda_itr_s=roots(P1);
    Lamda_itr_s=Lamda_itr_s(imag(Lamda_itr_s)==0);
   b=real(Lamda_itr_s)<lamda_itr(1) & real(Lamda_itr_s)>-12;
   if b==0
      break
   end
   lamda_itr(i) = Lamda_itr_s(b);
   delta1_(i) = 0.3-lamda_itr(i)/120;
   delta2_(i) = 37/315-lamda_itr(i)/945-lamda_itr(i)^2/9072;
   delta(i) = sqrt(lamda_itr(i)*mu/rho/U_d(i));
   delta1(i) = delta(i)*delta1_(i);
   delta2(i) = delta(i)*delta2_(i);
   f1(i) = delta1_(i)/delta2_(i);
   f2(i) = (2+lamda_itr(i)/6)*delta2_(i);
   F(i) = 2*f2(i)-4*K(i)-2*K(i)*f1(i);
   z(i+1) = z(i)+F(i)/U(i)*(X(i+1)-X(i));
end
```

Plots

$\delta \; vs \; X_{Laminar}$

```
figure(1)
cla; hold on; grid on;
set(gcf,'Color','White');
set(gca,'FontSize',12);
plot(X, delta,'Color',[0.2 0.984 0.6],'LineWidth',2)
xlim([0 0.7])
xlabel('$X$','interpreter','latex','FontSize',14);
ylabel('$\delta$','interpreter','latex','FontSize',14);
title('$\delta$ $vs$ $X$','interpreter','latex','FontSize',14);
hold on
```

```
plot(xfoil(1:ceil(length(xfoil)/2)),yfoil(1:ceil(length(yfoil)/2))/12,'-.','Color',[0 0.6 0.96],'LineWidth',2)
legend('$\delta$ $vs$ $X_{Laminar}$','NACA0012','interpreter','latex','FontSize',8)
```

$\delta_1 \; vs \; X_{Laminar}$

```
figure(2)
cla; hold on; grid on;
set(gcf,'Color','White');
set(gca,'FontSize',12);
plot(X, delta1,'Color',[0.2 0.984 0.6],'LineWidth',2)
xlim([0 0.7])
xlabel('$X$','interpreter','latex','FontSize',14);
ylabel('$\delta_{1}$','interpreter','latex','FontSize',14);
title('$\delta_{1}$ $vs$ $X$','interpreter','latex','FontSize',14);
hold on
plot(xfoil(1:ceil(length(xfoil)/2)),yfoil(1:ceil(length(yfoil)/2))/40,'-.','Color',[0 0.6 0.96],'LineWidth',2)
legend('$\delta_1$ $vs$ $X_{Laminar}$','NACA0012','interpreter','latex','FontSize',8)
```

$\delta_2 \; vs \; X_{Laminar}$

```
figure(3)
cla; hold on; grid on;
set(gcf,'Color','White');
set(gca,'FontSize',12);
plot(X, delta2,'Color',[0.2 0.984 0.6],'LineWidth',2)
xlim([0 0.7])
xlabel('$X$','interpreter','latex','FontSize',14);
ylabel('$\delta_{2}$','interpreter','latex','FontSize',14);
title('$\delta_{2}$ $vs$ $X_{Laminar}$','interpreter','latex','FontSize',14);
hold on
plot(xfoil(1:ceil(length(xfoil)/2)),yfoil(1:ceil(length(yfoil)/2))/120,'-.','Color',[0 0.6 0.96],'LineWidth',2)
legend('$\delta_2$ $vs$ $X_{Laminar}$','NACA0012','interpreter','latex','FontSize',8)
```

$C_{fx} \; vs \; X_{Laminar}$

```
figure(4)
cla; hold on; grid on;
set(gcf,'Color','White');
set(gca,'FontSize',12);
cfx = ( 2.*f2)./(sqrt(rho*U_inf/mu).* U.* sqrt(z));
plot(X,cfx,'Color',[0.2 0.984 0.6],'LineWidth',2)
xlabel('$X$','interpreter','latex','FontSize',14);
ylabel('$C{fx}$','interpreter','latex','FontSize',14);
title('$C_{fx}$ $vs$ $X_{Laminar}$','interpreter','latex','FontSize',14);
```

```
hold on
plot(xfoil(1:ceil(length(xfoil)/2)),yfoil(1:ceil(length(yfoil)/2))/12,'-.','Color',[0 0.6 0.96],'LineWidth',2)
legend('$C_{fx}$ $vs$ $X_{Laminar}$','NACA0012','interpreter','latex','FontSize',8)
```

$\tau_w \; vs \; X_{Laminar}$

```
figure(5)
cla; hold on; grid on;
set(gcf,'Color','White');
set(gca,'FontSize',12);
% tau_w = cfx * 0.5 * rho;
tau_w_notbar = 0.5*rho*cfx.*U.^2*U_inf^2;
plot(X,tau_w_notbar,'Color',[0.2 0.984 0.6],'LineWidth',2)
xlabel('$X$','interpreter','latex','FontSize',14);
ylabel('$\tau_{w}$','interpreter','latex','FontSize',14);
title('$\tau_{w}$ $vs$ $X_{Laminar}$','interpreter','latex','FontSize',14);
hold on
plot(xfoil(1:ceil(length(xfoil)/2)),yfoil(1:ceil(length(yfoil)/2))*40,'-.','Color',[0 0.6 0.96],'LineWidth',2)
legend('$\tau_w$ $vs$ $X_{Laminar}$','NACA0012','interpreter','latex','FontSize',8)
```

Accounting for Transition Using Smith model to locate the point of transition.

```
H=f1;
log_Rex=log10(rho.*U.*X./mu);
F_H=-40.4557+64.8066*H-26.7538*H.^2+3.3819*H.^3;
d=find(abs(real(log_Rex-F_H))<0.01);
x_s=X(d);
f=f1(d);
delta2(d) = delta2(d-1);
delta(d) = 1.4*delta(d-1);

syms x
eqn= (-delta(d)/delta2(d)+x+3.3+1.5501*(x-0.6778)^-3.064==0);
H_(d) = double(vpasolve(eqn,x,1));
H_11 = double(vpasolve(1.1+0.86*(x-3.3)^-0.777==H_(d),x,1));

ඪ ඪ ඪ ඪ ඪ ඪ ඪ ඪ ඪ ඪ ඪ ඪ ඪ ඪ ඪ ඪ ඪ ඪ ඪ ඪ ඪ ඪ ඪ ඪ ඪ ඪ ඪ ඪ ඪ ඪ ඪ ඪ ඪ ඪ ඪ ඪ ඪ ඪ ඪ ඪ ඪ ඪ ඪ ඪ ඪ
H_1(d)=H_11;
for i = d+1:length(U_d)
   if H_(i-1) > 1.8 && H_(i-1) < 2.8
      break
   end

   Re_delta2_(i-1) = U(i-1)*U_inf .* delta2(i-1) * (rho/mu);
   cfx(i-1) = 0.246*10^(-0.678*H_(i-1)) * (Re_delta2_(i-1))^-0.268;
   delta2_by_dx = cfx(i-1)/2 - (delta2(i-1) / U(i-1)) * U_d(i-1);
```

```
    delta2(i) = delta2(i-1) + delta2_by_dx * (X(i) - X(i-1));

    H_1(i) = (0.306e-1 * U(i) * ((H_1(i - 1) - 3) ^ (-0.6169e0)) + (U(i) * delta2(i) * H_1(i - 1) / (X(i) - X(i - 1)))) / ((U(i) * delta2(i) - U(i - 1) * delta2(i - 1)) / (X(i) - X(i - 1)) + U(i) * delta2(i) / (X(i) - X(i - 1)));

    if (H_1(i) <= 3.3)
        H_(i) = 3;
    elseif (H_1(i) > 5.3)
        H_(i) = 1.1+0.86*(H_1(i)-3.3)^-0.777;
    else
        H_(i) = 0.6778 + 1.1536*(H_1(i) - 3.3 )^-0.326;
    end
end
```

$\delta_2 \; vs \; X_{Transition}$

```
figure(6)

cla; hold on; grid on;
set(gcf,'Color','White');
set(gca,'FontSize',12);
plot(X, delta2,'Color',[0.2 0.984 0.6],'LineWidth',2)
xlim([0 1])
xlabel('$X$','interpreter','latex','FontSize',14);
ylabel('$\delta_{2}$','interpreter','latex','FontSize',14);
title('$\delta_{2}$ $vs$ $X$ $with$ $Transition$','interpreter','latex','FontSize',14);
hold on
plot(xfoil(1:ceil(length(xfoil)/2)),yfoil(1:ceil(length(yfoil)/2))/120,'-.','Color',[0 0.6 0.96],'LineWidth',2)
xline(x_s,'LineWidth',1.5)
legend('$\delta_2$ $vs$ $X_{Transition}$','NACA0012','Transition Line','interpreter','latex','FontSize',8)
```

$C_{fx} \; vs \; X_{Transition}$

```
figure(7)
cla; hold on; grid on;
set(gcf,'Color','White');
set(gca,'FontSize',12,'fontname','times');
plot(X,cfx,'Color',[0.2 0.984 0.6],'LineWidth',2)
hold on
plot(xfoil(1:ceil(length(xfoil)/2)),yfoil(1:ceil(length(yfoil)/2))/12,'-.','Color',[0 0.6 0.96],'LineWidth',2)
xline(x_s,'LineWidth',1.5)
legend('$C_{fx}$ $vs$ $X_{Transition}$','NACA0012','Transition Line','interpreter','latex','FontSize',8)
xlabel('$X$','interpreter','latex','FontSize',14);
ylabel('$C{fx}$','interpreter','latex','FontSize',14);
title('$C_{fx}$ $vs$ $X$ $with$ $Transition$','interpreter','latex','FontSize',14);
```

τ_w vs $X_{Transition}$

```
figure(8)
cla; hold on; grid on;
set(gcf,'Color','White');
set(gca,'FontSize',12,'fontname','times');
tau_w = cfx * 0.5 * rho.*U.^2*U_inf^2;
plot(X,tau_w,'Color',[0.2 0.984 0.6],'LineWidth',2)
hold on
plot(xfoil(1:ceil(length(xfoil)/2)),yfoil(1:ceil(length(yfoil)/2))*40,'-.','Color',[0 0.6 0.96],'LineWidth',2)
xline(x_s,'LineWidth',1.5)
legend('$\tau_{w}$ $vs$ $X_{Transition}$','NACA0012','Transition Line','interpreter','latex','FontSize',8)
xlabel('$X$','interpreter','latex','FontSize',14);
ylabel('$\tau_{w}$','interpreter','latex','FontSize',14);
title('$\tau_{w}$ $vs$ $X$ $with$ $Transition$','interpreter','latex','FontSize',14);
```

```
X_req=linspace(0.1,1,10);
index=zeros(1,length(X_req));
U_req=zeros(1,length(X_req));
U_d_req=zeros(1,length(X_req));
Z_req=zeros(1,length(X_req));
K_req=zeros(1,length(X_req));
lamda_itr_req=zeros(1,length(X_req));
f1_req=zeros(1,length(X_req));
f2_req=zeros(1,length(X_req));
F_req=zeros(1,length(X_req));
delta_req=zeros(1,length(X_req));
delta1_req=zeros(1,length(X_req));
delta2_req=zeros(1,length(X_req));
cfx_req=zeros(1,length(X_req));
tau_w_notbar_req=zeros(1,length(X_req));
for i = 1:1:length(X_req)
index(i)=find(X>=X_req(i),1,'first' );
U_req(i)=U(index(i))*U_inf;
U_d_req(i)=U_d(index(i))*U_inf;
Z_req(i)=z(index(i));
K_req(i)=K(index(i));
lamda_itr_req(i)=lamda_itr(index(i));
f1_req(i)=f1(index(i));
f2_req(i)=f2(index(i));
F_req(i)=F(index(i));
delta_req(i)=delta(index(i));
delta1_req(i)=delta1(index(i));
delta2_req(i)=delta2(index(i));
cfx_req(i)=cfx(index(i));
tau_w_notbar_req(i)=tau_w_notbar(index(i));
end
```

Getting Required Data to Fill the Table

```
fprintf('======= Sussy results =======\n');
Data = [X_req',U_req',U_d_req',Z_req',K_req',lamda_itr_req',f1_req',f2_req',delta_req',delta1_req',delta2_req',cfx_req',tau_w_notbar_req'];
VarNames = {'X','U','U_dash','Z','K','Lambda','f1','f2','delta','delta 1','delta 2','Cfx','tau_w'};
T = table(Data(:,1),Data(:,2),Data(:,3),Data(:,4),Data(:,5),Data(:,6),Data(:,7),Data(:,8),Data(:,9),Data(:,10),Data(:,11),Data(:,12),Data(:,13),'VariableNames',VarNames)
  warning('off','MATLAB:xlswrite:AddSheet'); %optional
  writematrix(Data,'test.xlsx','Sheet',1);
```

```
======= Sussy results =======
```

X	U	U_dash	Z	K	Lambda	f1	f2	delta	delta 1	delta 2	Cfx	tau_w
0.1	59.32	5.0557	1.531	0.0030962	0.22535	2.5434	0.23883	0.005705	0.0017008	0.00066872	0.0012435	2.6801
0.2	58.89	-9.2391	3.658	-0.013519	-0.96468	2.6022	0.21772	0.0087317	0.0026897	0.0010336	0.00073872	1.5692
0.3	57.759	-12.179	6.37	-0.031033	-2.1824	2.6684	0.19512	0.011439	0.0036397	0.001364	0.0005115	1.0452
0.4	56.53	-12.831	9.5294	-0.04891	-3.4087	2.7416	0.17152	0.013928	0.0045739	0.0016683	0.00037561	0.73521
0.5	55.245	-12.839	13.454	-0.069096	-4.7984	2.8332	0.14403	0.01652	0.0056165	0.0019824	0.00027163	0.50777
0.6	53.96	-12.923	18.241	-0.094294	-6.5868	2.9661	0.10795	0.026905	0.0068465	0.0022998	0.002924	0.31923
0.7	52.598	-13.887	24.522	-0.13622	-9.935	3.2691	0.040301	0.022856	0.008749	0.0025092	0.0028524	0.1002
0.8	51.116	-17.117	0	0	0	0	0	0	0	0.0027213	0.002794	NaN
0.9	48.958	-28.815	0	0	0	0	0	0	0	0.002982	0.0027427	NaN
1	38.169	-979.82	0	0	0	0	0	0	0	0.0038604	NaN	NaN

Published with MATLAB® R2021a

References

[1] D. H. E. Banna, Lecture Handouts, 2022.

[2] P. C. S. H. C. E.L. Houghton, Aerodynamics for Engineering Students, Elsevier Ltd., 2017.

[3] MT, "Approximate Method Based on the Momentum Equation: Pohlhausen's Method.," [Online]. Available: http://web.mit.edu/fluids-modules/www/highspeed_flows/ver2/bl_Chap2/node12.html. [Accessed 17 5 2022].