Aerodynamics

MECH 6121

Project: Calculation of Airfoil characteristics for NACA 6412 Airfoil

For: Dr. Pierre Q. Gauthier

Submitted by-

Rahul Chug

40075138

Contents

List of Figures

Introduction

The National Advisory Committee for Aeronautics (**NACA**) identified different logical shapes as different numbers. Even though the first patented airfoil came in 1884, in 1930s, first series of number system was developed by NACA, a 4-number series such as NACA 6412. The first digit is the maximum camber in hundredths of chord, the second digit is the location of maximum camber along the chord from the leading edge in tenths of chord, and the last two digits give the maximum thickness in hundredths of chord. For the NACA 6412 airfoil, .06c is the maximum camber located at 0.4c from the leading edge, and the maximum thickness is 0.12*c*.An airfoil in which the mean camber line coincides with the chord line or in other words where camber is zero, is called a symmetric airfoil. NACA 6412 is an example of Unsymmetrical or Cambered airfoil.

At later times, more family of NACA airfoil series such as five- and six-digit series were introduced. In this project we are focused only on the 4-digit series.

Basic terminology for an airfoil is shown in Figure 1.

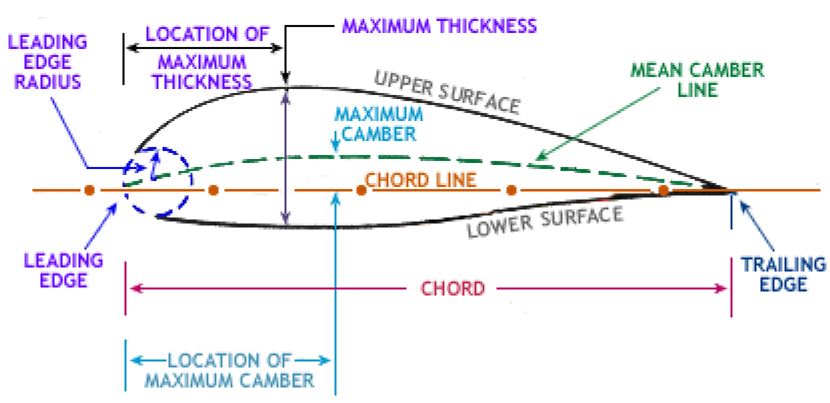

Figure 1 Airfoil Nomenclature

The airfoil data points for the project have been used from the UIUC airfoil database.[3]

Methodology

The foundation theory for calculating the required airfoil characteristics has been presented in the Model formulation. It first describes the source panel method to calculate the source panel strengths and further with the help of thin airfoil theory, airfoil characteristics such as Coefficient of lift, coefficient of moment, and center of pressure are calculated. A code has been generated on MATLAB and the outcomes are presented in the Results and discussion part.

Nomenclature

Freestream velocity – V_∞
Angle of attack (AoA)– α
Angle between the normal vector and the free stream velocity- β_i
Angle between the normal and the x-direction- δ_i
Geometry - r_{ij}
Panel source strength of panel j - λ_j
Coefficient of lift-C_L
Coefficient of moment-C_M
Panel length of j^{th} panel- S_j

Model Formulation

Source Panel Method

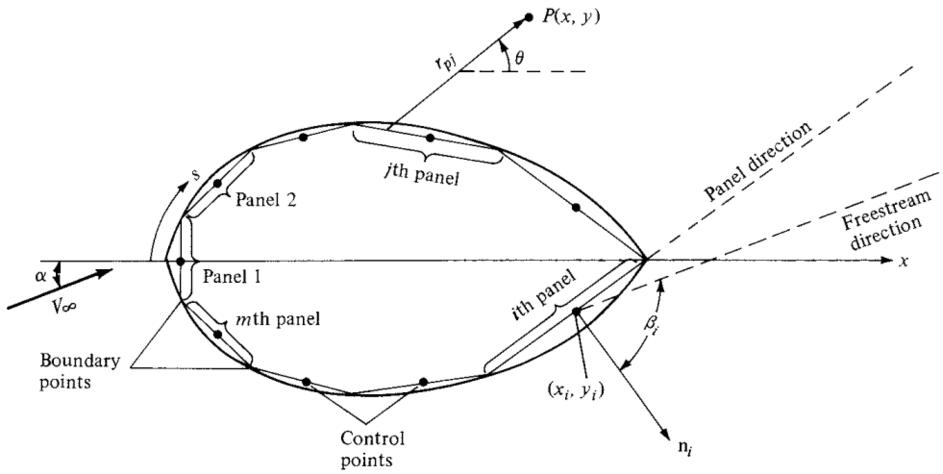

Figure 2 Source panel distribution over the surface of a body of arbitrary shape.[1]

For N flat panels, assume source strength is constant on each panel
Velocity potential at point P due to uniform flow and all the source panels,

$$\varphi_P = V_\infty \cos(\alpha)\, x + V_\infty \sin(\alpha)\, y + \sum_{j=1}^{N} \frac{\lambda_j}{2\pi} \int \ln\,(r_{Pj}) ds_j$$

Now instead of solving at point P, let's solve at the control points of the panels

$$\varphi_i = V_\infty \cos(\alpha)\, x + V_\infty \sin(\alpha)\, y + \sum_{j=1}^{N} \frac{\lambda_j}{2\pi} \int \ln\,(r_{ij}) ds_j$$

Normal or tangential velocity at point i,

Note that the Normal component is zero as there is no flow perpendicular to the panels

$$V_{n,i} = \frac{\partial \varphi_i}{\partial n_i} = V_\infty \cos(\alpha) \frac{\partial x_i}{\partial n_i} + V_\infty \sin(\alpha) \frac{\partial y_i}{\partial n_i} + \sum_{j=1}^{N} \frac{\lambda_j}{2\pi} \int \frac{\partial}{\partial n_i} \ln\,(r_{ij}) ds_j = 0$$

$$V_{t,i} = \frac{\partial \varphi_i}{\partial t_i} = V_\infty \cos(\alpha) \frac{\partial x_i}{\partial t_i} + V_\infty \sin(\alpha) \frac{\partial y_i}{\partial t_i} + \sum_{j=1}^{N} \frac{\lambda_j}{2\pi} \int \frac{\partial}{\partial t_i} \ln\,(r_{ij}) ds_j$$

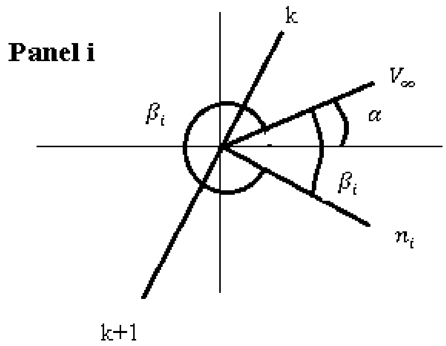

Figure 3 Free stream and Normal velocity components on Panel i

From the figure 3, the normal (or tangential) velocity due to free stream flow

$$V_{n,i} = V_\infty \cos(\beta_i)$$
$$V_{t,i} = V_\infty \sin(\beta_i)$$

Where, β_i is the angle between the free stream vector and the panel outward normal.

Therefore,

$$V_{n,i} = V_\infty \cos(\beta_i) + \sum_{j=1}^{N} \frac{\lambda_j}{2\pi} \int \frac{\partial}{\partial n_i} \ln(r_{ij}) ds_j$$
$$V_{t,i} = V_\infty \sin(\beta_i) + \sum_{j=1}^{N} \frac{\lambda_j}{2\pi} \int \frac{\partial}{\partial t_i} \ln(r_{ij}) ds_j$$

Now consider the normal velocity parameter,
Let,

$$I_{ij} = \int \frac{\partial}{\partial n_i} \ln(r_{ij}) ds_j$$

$$I_{ij} = \int \frac{1}{r_{ij}} \frac{\partial r_{ij}}{\partial n_i} ds_j$$

Where, $r_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$

By solving the differential, we will get

$$I_{ij} = \int \frac{(x_i - x_j)\frac{\partial x_i}{\partial n_i} + (y_i - y_j)\frac{\partial y_i}{\partial n_i}}{(x_i - x_j)^2 + (y_i - y_j)^2} ds_j$$

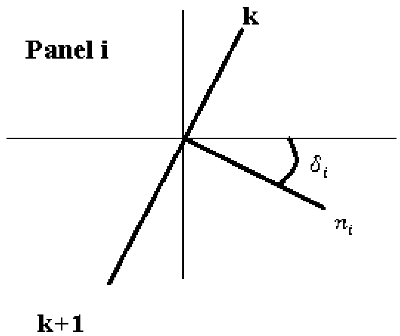

Figure 4 Panel i showing the normal vector

δ_i is the angle between the normal and the x- direction

$$\cos(\delta_i) = \frac{\partial x_i}{\partial n_i}$$

$$\sin(\delta_i) = \frac{\partial y_i}{\partial n_i}$$

$$I_{ij} = \int \frac{(x_i - x_j)\cos(\delta_i) + (y_i - y_j)\sin(\delta_i)}{(x_i - x_j)^2 + (y_i - y_j)^2} ds_j$$

We can express the terms x_j and y_j in terms of s_j

$$x_j = X_j + s_j\cos(\varphi_j)$$
$$y_j = Y_j + s_j\sin(\varphi_j)$$

Also, δ= φ +90

After putting these values and solving the integral, we get

$$I_{ij} = \int_0^{s_j} \frac{Cs_j + D}{s_j^2 + 2As_i + B} ds_j$$

$$A = -(x_i - X_j)\cos(\varphi_j) - (y_i - Y_j)\sin(\varphi_j)$$
$$B = (x_i - X_j)^2 + (y_i - Y_j)^2$$
$$C = \sin(\varphi_i - \varphi_j)$$
$$D = -(x_i - X_j)\sin(\varphi_i) + (y_i - Y_j)\cos(\varphi_i)$$

Considering the denominator of I_{ij},

$$x^2 + 2Ax + B = x^2 + 2Ax + \left(\frac{2A}{2}\right)^2 b - \left(\frac{2A}{2}\right)^2$$
$$= (x^2 + 2Ax + a^2) + (B - A^2)$$

Put $E^2 = B - A^2$

$$\int \frac{Cx + D}{(x+A)^2 + E^2} dx$$

Define u= x + A, and solving

$$\int \frac{Cu + D - AC}{u^2 + E^2} du = C \int \frac{u}{u^2 + E^2} du + (D - AC) \int \frac{1}{u^2 + E^2} du$$

After integration, we get

$$\boxed{I_{ij} = \frac{C}{2}\left[\ln\left(\frac{{s_j}^2 + 2As_j + B}{B}\right)\right] + \frac{D - AC}{E}\left[\tan^{-1}\left(\frac{s_j + A}{E}\right) - \tan^{-1}\left(\frac{A}{E}\right)\right]}$$

Now consider the tangential velocity parameter, let

$$J_{ij} = \int \frac{\partial}{\partial t_i} \ln\,(r_{ij}) ds_j$$

Like the normal velocity solution, we will get

$$\boxed{J_{ij} = \frac{C}{2}\left[\ln\left(\frac{{s_j}^2 + 2As_j + B}{B}\right)\right] + \frac{D - AC}{E}\left[\tan^{-1}\left(\frac{s_j + A}{E}\right) - \tan^{-1}\left(\frac{A}{E}\right)\right]}$$

Where

$$A = -(x_i - X_j)\cos(\varphi_j) - (y_i - Y_j)\sin\,(\varphi_j)$$
$$B = (x_i - X_j)^2 + (y_i - Y_j)^2$$
$$C = -cos\,(\varphi_i - \varphi_j)$$
$$D = (x_i - X_j)\cos(\varphi_i) + (y_i - Y_j)\sin(\varphi_i)$$
$$E = \sqrt{B - A^2}$$

For calculating the source strength of panels, we only need the normal velocity equation

$$V_{n,i} = V_\infty \cos(\beta_i) + \sum_{j=1}^{N} \frac{\lambda_j I_{ij}}{2\pi}$$

$$V_{n,i} = V_\infty \cos(\beta_i) + \frac{\lambda_i}{2} + \sum_{\substack{j=1 \\ j \neq i}}^{N} \frac{\lambda_j I_{ij}}{2\pi} = 0$$

Now writing the above equation for each panel, assume 3 panels for simplicity

$$V_{n,1} = V_\infty \cos(\beta_1) + \frac{\lambda_1}{2} + \left[\frac{\lambda_2 I_{12}}{2\pi} + \frac{\lambda_3 I_{13}}{2\pi}\right] = 0$$

$$V_{n,2} = V_\infty \cos(\beta_2) + \frac{\lambda_2}{2} + \left[\frac{\lambda_1 I_{21}}{2\pi} + \frac{\lambda_3 I_{23}}{2\pi}\right] = 0$$

$$V_{n,3} = V_\infty \cos(\beta_3) + \frac{\lambda_3}{2} + \left[\frac{\lambda_1 I_{31}}{2\pi} + \frac{\lambda_2 I_{32}}{2\pi}\right] = 0$$

In matrix form,

$$\begin{bmatrix} \frac{1}{2} & \frac{I_{12}}{2\pi} & \frac{I_{13}}{2\pi} \\ \frac{I_{21}}{2\pi} & \frac{1}{2} & \frac{I_{23}}{2\pi} \\ \frac{I_{31}}{2\pi} & \frac{I_{32}}{2\pi} & \frac{1}{2} \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \end{bmatrix} = \begin{bmatrix} -V_\infty \cos(\beta_1) \\ -V_\infty \cos(\beta_2) \\ -V_\infty \cos(\beta_3) \end{bmatrix}$$

$$AX = b$$

$$\boxed{X = A\backslash b}$$

For n panels,

$$\begin{bmatrix} \pi & I_{12} & \cdots & I_{1N} \\ I_{21} & \pi & \cdots & I_{2N} \\ \vdots & \vdots & \vdots & \vdots \\ I_{N1} & I_{N2} & \cdots & \pi \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_N \end{bmatrix} = \begin{bmatrix} -V_\infty 2\pi\cos(\beta_1) \\ -V_\infty 2\pi\cos(\beta_2) \\ \vdots \\ -V_\infty 2\pi\cos(\beta_N) \end{bmatrix}$$

From here source panel strength can be calculated which can be further used to calculate the normal velocities.

Panel velocity can be calculated by

$$\boxed{V_t = V_\infty \sin\beta + \frac{\lambda . J}{2\pi}}$$

Coefficient of pressure

$$\boxed{C_P = 1 - \left(\frac{V_t}{V_\infty}\right)^2}$$

Thin Airfoil Theory

From thin airfoil theory, for NACA 4-digit MP XX unsymmetrical airfoil

Mean camber line equation is

$$Z_c(x) = \frac{M}{P^2}(2Px - x^2) \qquad 0 < x < P$$

$$Z_c(x) = \frac{M}{(1-P)^2}[(1-2P) + 2Px - x^2] \quad P < x < C$$

$$where\ x = \frac{C}{2}(1 - \cos\theta)$$

For NACA 6412 this equation can be re-written as, considering Chord C= 1

$$Z_c(x) = \frac{0.06}{(0.4)^2}[(2 * 0.4 * x) - x^2] \qquad 0 < x < 0.4$$

$$Z_c(x) = \frac{0.06}{(1-0.4)^2}[(1 - (2 * 0.4)) + (2 * 0.4 * x) - x^2] \quad 0.4 < x < 1$$

The fundamental equation of thin airfoil theory is

$$U_\infty\left(\alpha - \frac{dZ_c}{dx}\right) = \frac{1}{2\pi}\left(\int_0^\pi \frac{\gamma(\theta)\sin\theta}{\cos\theta - \cos\theta_P}\right)$$

As 6412 is an unsymmetrical airfoil,

Assume

$$\gamma(\theta) = 2U_\infty A_0\left(1 + \frac{cos\theta}{\sin\theta}\right) + 2U_\infty \sum_{n=1}^{\infty} A_n \sin n\theta$$

Substitute this in the fundamental equation, and solve for $\frac{dZ_c}{dx}$, we get

$$\frac{dZ_c}{dx} = \alpha - A_0 + \sum_{n=1}^{\infty} A_n \cos(n\theta)$$

This term is similar to Fourier cosine series expansion for the function of dZ/dx, therefore the coefficients A_0 and A_n can be written as

$$\alpha - A_0 = \frac{1}{\pi}\int_0^{\pi} \frac{dZ_c}{dx} d\theta$$
$$A_0 = \alpha - \frac{1}{\pi}\int_0^{\pi} \frac{dZ_c}{dx} d\theta$$

$$A_n = \frac{2}{\pi}\int_0^{\pi} \frac{dZ_c}{dx} \cos{(n\theta)} d\theta$$

Coefficient of lift

$$C_l = \frac{1}{(1/2)\rho_\infty U_\infty^2 C}\int_0^C \rho_\infty U_\infty \gamma(x) dx$$

By substituting γ from the assumption we made earlier,

$$= \left\{\int_0^{\pi} A_0(1 + cos\theta) d\theta + \sum_{n=1}^{\infty} A_n \int_0^{\pi} sin\theta. \sin{(n\theta)} d\theta\right\}$$

When $n = 1, sin\theta. \sin(n\theta) = \pi/2$
When $n \neq 1, sin\theta. \sin(n\theta) = 0$

After integrating and solving, we get

$$\boxed{C_l = 2\pi\left[A_0 + \frac{A_1}{2}\right]_0}$$

Coefficient of Moment

$$C_m = \frac{-1}{(1/2)\rho_\infty U_\infty^2 C^2}\int_0^C \rho_\infty U_\infty. x. \gamma(x) dx$$

After solving, we get

$$\boxed{C_m = -\frac{\pi}{2}\left[A_0 + A_1 - \frac{A_2}{2}\right]}$$

Center of Pressure

$$\boxed{x_{CP} = \frac{C}{4}\left[\frac{2A_0 + 2A_1 - A_2}{2A_0 + A_1}\right]}$$

Results and Discussion

The plots were obtained using MATLAB. Figure 5 shows the airfoil geometry of NACA 6412 along with the boundary and control points all over the surface. As can been seen in the plot, NACA 6412 is an unsymmetrical airfoil producing a net lift.

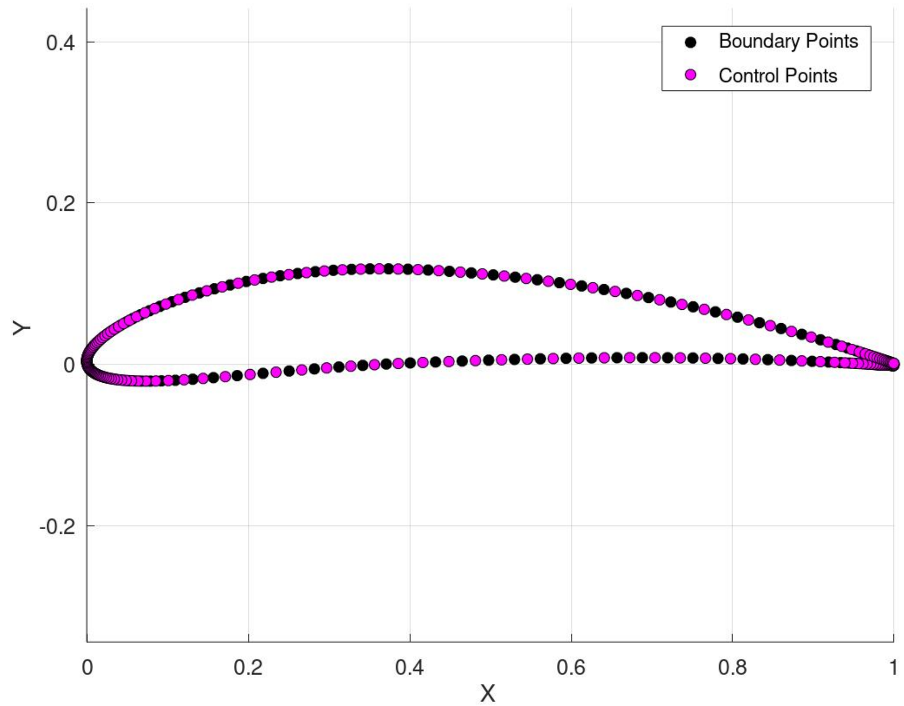

Figure 5 Boundary and Control points for NACA 6412 Airfoil

For the comparison of pressure generated on the upper and lower surface of the airfoil, different angle of attacks were considered ranging from -5^o, 0^o and 5^o. Further, Airfoil characteristics such as coefficient of lift, coefficient of moment and center of pressure were also calculated.

-5º Angle of Attack

The coefficient of pressure plots for -5º AoA have been shown in Figure6-7.

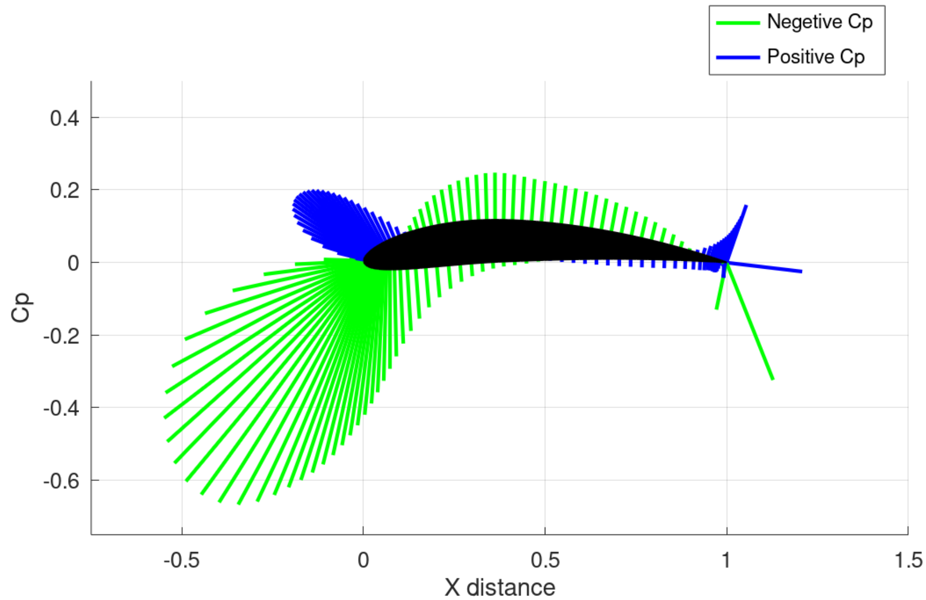

Figure 6 Coefficient of pressure values around the Airfoil Surface for -5º AOA

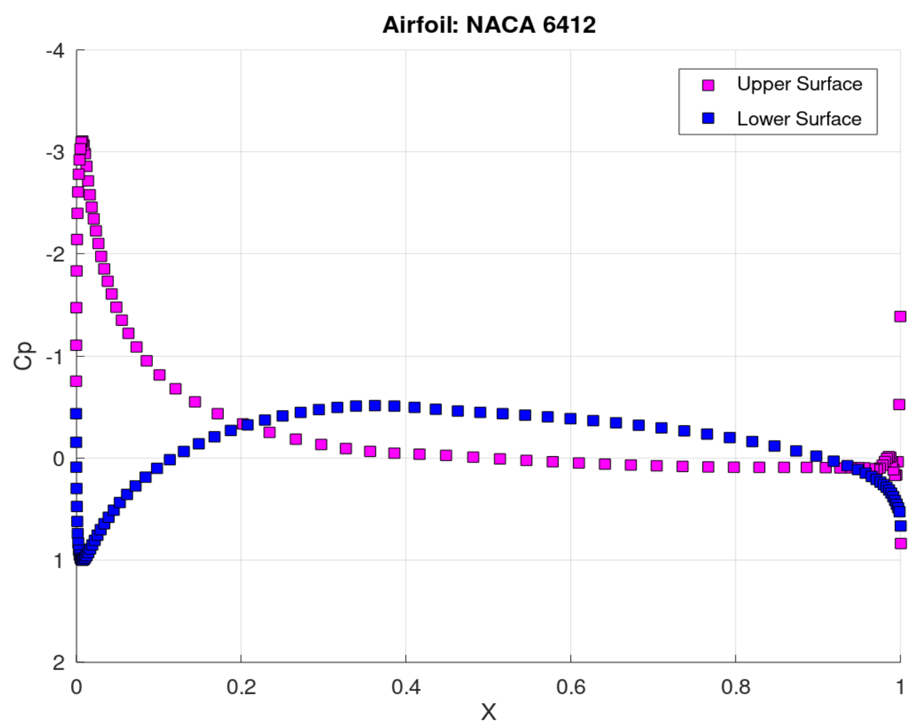

Figure 7 Coefficient of Pressure for the upper and lower surface of the airfoil for -5º AOA

Pressure distribution is simply the pressure at all points around an airfoil. Plots for the upper surface are drawn so that negative numbers are higher on the graph, as the upper surface of the airfoil will usually be farther below zero and will hence be the top line on the graph.

Upper surface pressure is below zero, however, the lower surface is positive giving rise to the difference in the pressure around the airfoil and causing lift.

The results for -5^o Angle of attack obtained from the code are:

Sum of source panel strengths: 0.00293821

Coefficient of Lift **C_L = 0.13507**

Coefficient of Moment **C_M = -0.19313**

Center of Pressure **CP = 1.4298**

Angle of attack where lift is zero AOA_L0 = -0.10876 rad

Angle of attack where lift is zero **AOA_L0_deg = -6.2317 deg**

0^o Angle of Attack

Figure 8-9 shows the C_p effect around the airfoil for 0^o angle of attack.

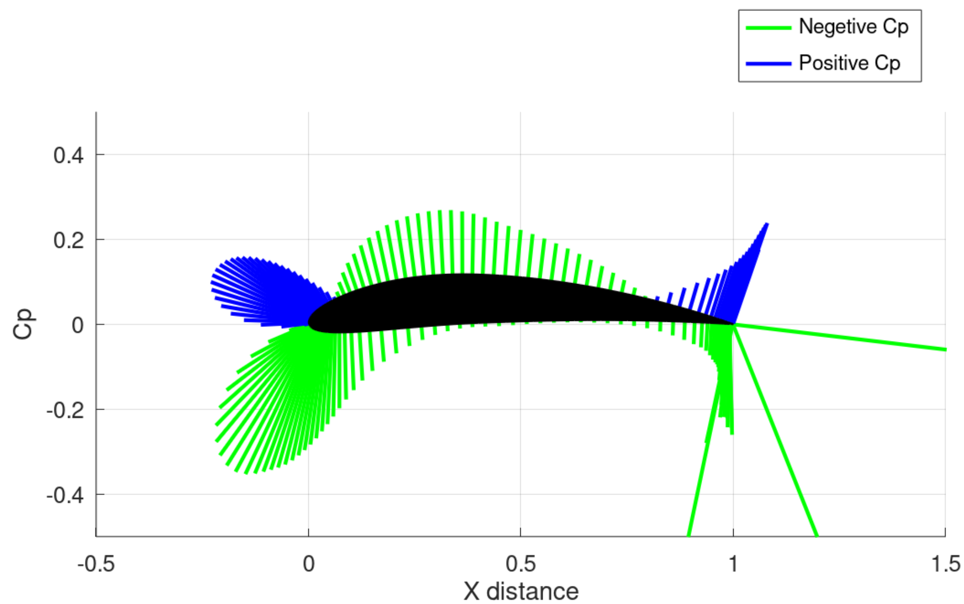

Figure 8 Coefficient of pressure values around the Airfoil Surface for 0^o AoA

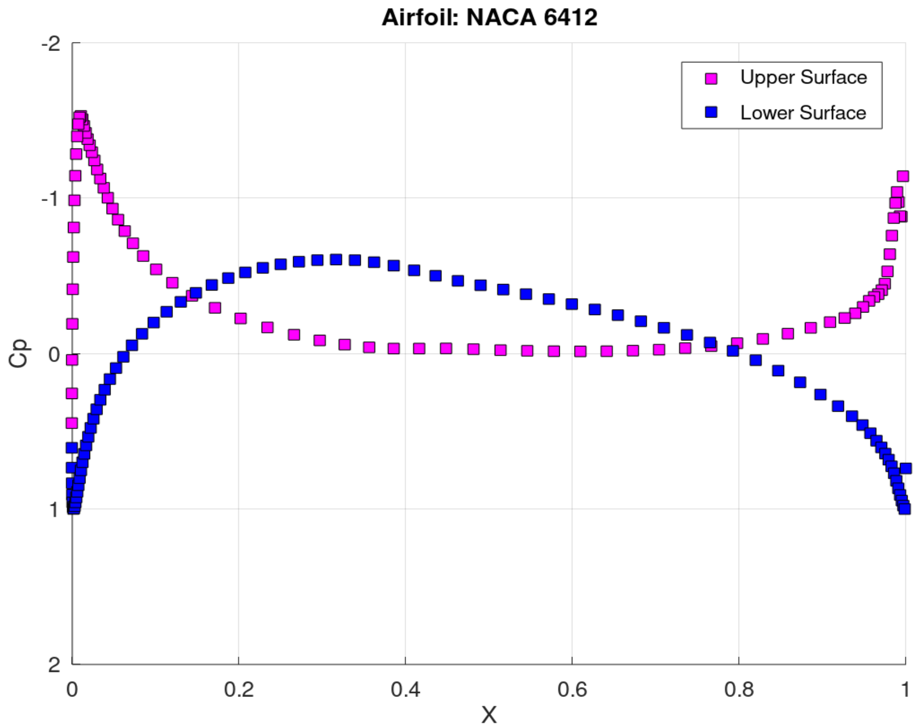

Figure 9 Coefficient of Pressure for the upper and lower surface of the airfoil for 0º AoA

Coefficient of lift for 0º AoA shows a much higher value of 0.68338 compared to 0.13507 for -5º AoA, which justifies the theory and goes along with the results seen in database of airfoil tools [4]. Source panel strengths have also been calculated for each panel and it was observed that the sum of source panel strengths was nearly zero (6.48682e-05), as for a closed polygon it must have a zero strengths sum.

Results obtained for 0º Angle of attack are -

Sum of source panels strengths: 6.48682e-05

Coefficient of Lift **C_L = 0.68338**

Coefficient of Moment **C_M = - 0.33020**

Center of Pressure **CP = 0.48319**

5º Angle of Attack

The pressure distribution for 5º AoA has been shown in Figure 10-11.

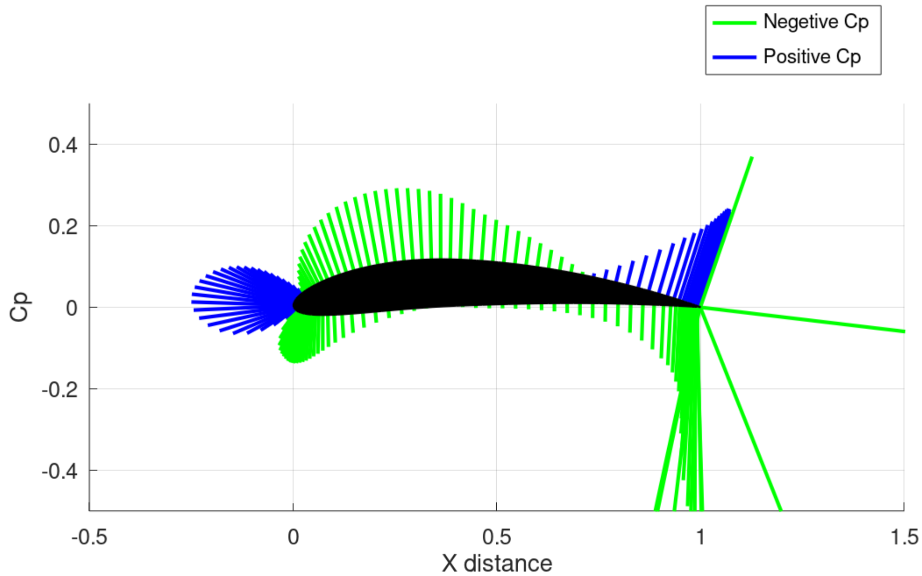

Figure 10 Coefficient of pressure values around the Airfoil Surface for 5º AoA

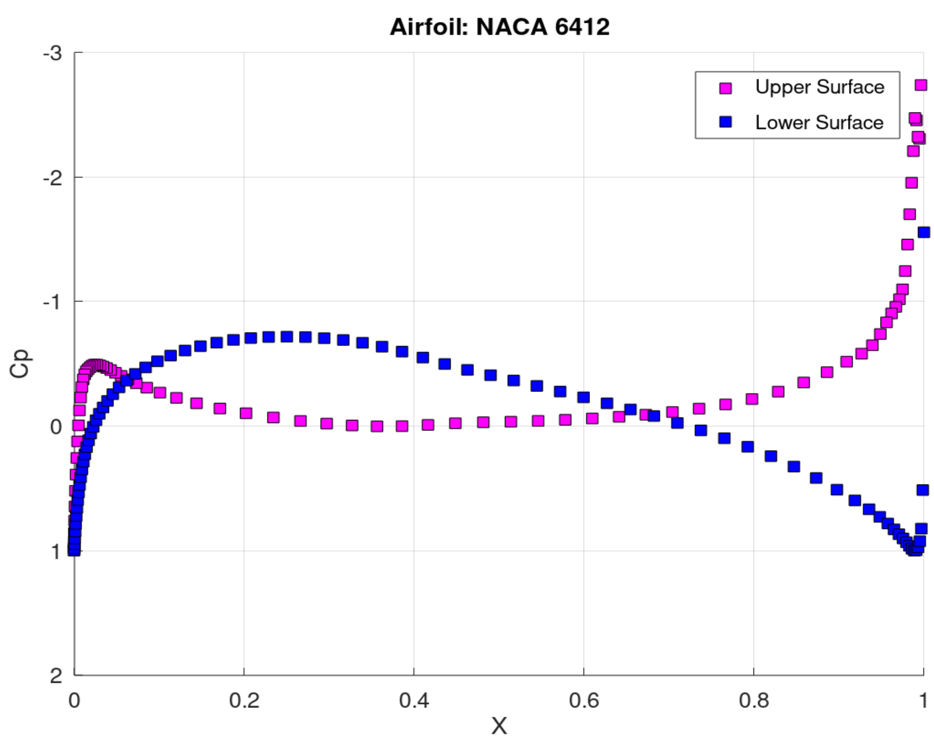

Figure 11 Coefficient of Pressure for the upper and lower surface of the airfoil for 5º AoA

Like the previous case, the C_L increased with the angle of attack. 5º AoA shows a lift coefficient of 1.2317, twice of that seen for 0º.

The results obtained for 5º Angle of attack are:

Sum of Source panel strengths: -0.00280897

Coefficient of Lift $\mathbf{C_L}$ **= 1.2317**

Coefficient of moment $\mathbf{C_M}$ **= -0.46728**

Center of pressure **CP = 0.37938**

The results for several angle of attack have been summarized in the below table along with the AoA which give rise to zero lift in the airfoil, which is a constant for all AoA. Further, plot was obtained to show the trend for lift and moment coefficient w.r.t AoA, as shown in Figure 12.

Angle of Attack	CL	CM	α for L=0
-5	0.13507	-0.1931	-6.2317
0	0.68338	-0.3302	-6.2317
5	1.2317	-0.4673	-6.2317
15	2.3283	0.74144	-6.2317

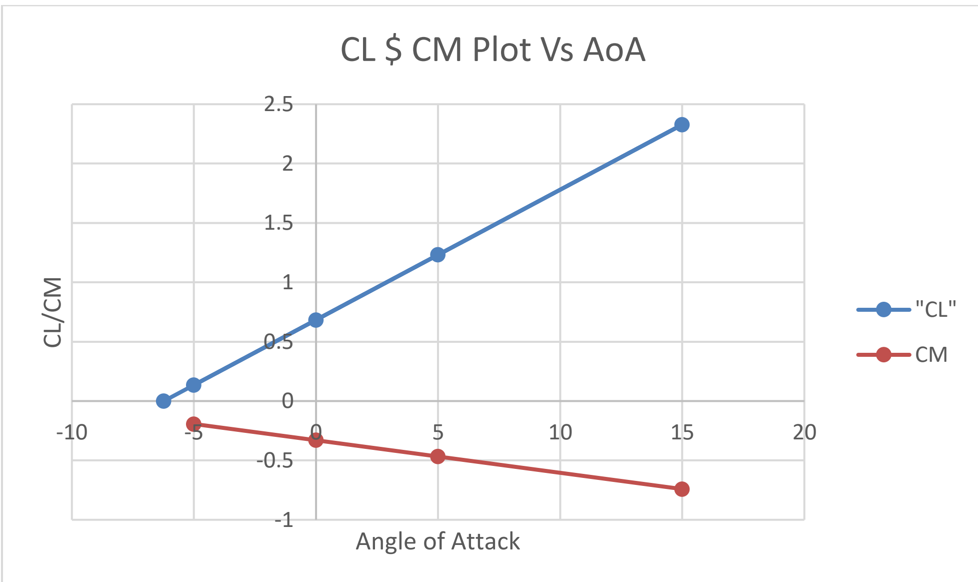

Figure 12 C_L /C_M Vs angle of attack

Conclusion

Firstly, source panel strengths were calculated, and it was verified that the sum of strengths was coming out to be zero as the airfoil is a closed polygon. Then, Airfoil characteristics were obtained for NACA 6412 for a range of angle of attacks and plots for C_L, C_P and C_M were plotted w.r.t AoA. A proportional behavior with angle of attack was observed for $C_{L,}$ however, C_M tend to decrease with increase AoA, as expected. Further, the angle of attack for zero lift was obtained, a constant irrespective of the change in AoA.

Reference

1. Anderson, J. D. (2017). *Fundamentals of aerodynamics*. New York, NY: McGraw-Hill Education.
2. Kullur, A., Khan, A., Eric, & Stetson Spencer. (n.d.). Source Panel Method. Retrieved April 20, 2020, from http://www.joshtheengineer.com/
3. Airfoil database. (n.d.). Retrieved April 19, 2020, from https://m-selig.ae.illinois.edu
4. Airfoil Tools. (n.d.). Retrieved April 15, 2020, from http://airfoiltools.com/

Appendix

Matlab code:

```
## Input
Vinf = 1;                          % Freestream velocity
AoA  = 0;                          % Angle of attack [deg]
NACA = '6412';                     % NACA airfoil to load [####]
## Convert angle of attack to radians
alpha = AoA*(pi/180);              % Angle of attack [rad]
## Number of boundary points and panels
numPts = length(XB);               % Number of boundary points
numPan = numPts - 1;
% Number of panels (control points)
%% SOURCE PANEL METHOD
% Initialize variables
XC   = zeros(numPan,1);            % X-coordinate array
YC   = zeros(numPan,1);            % Y-coordinate array
S    = zeros(numPan,1);            % panel length array
phiD = zeros(numPan,1);            % panel orientation angle array [deg]
% geometric quantities for a panel of airfoil
for i = 1:1:numPan                 % Loop over all panels
  XC(i)   = 0.5*(XB(i)+XB(i+1));  % X-value of control point
  YC(i)   = 0.5*(YB(i)+YB(i+1));  % Y-value of control point
  dx      = XB(i+1)-XB(i);        % Change in X between boundary points
  dy      = YB(i+1)-YB(i);        % Change in Y between boundary points
  S(i)    = (dx^2 + dy^2)^0.5;    % Length of the panel
        phiD(i) = atan2d(dy,dx);  % Angle of the panel (positive X-axis to inside face)
  if (phiD(i) < 0)
```

```
        phiD(i) = phiD(i) + 360;
    end
end
% Calculate angles
deltaD          = phiD + 90;                        % Angle from positive X-axis to outward normal vector
betaD           = deltaD - AoA;                     % Angle between freestream vector and outward normal
betaD(betaD > 360) = betaD(betaD > 360) - 360;      % Make sure angles aren't greater than 360 [deg]
% Convert angles from [deg] to [rad]
phi  = phiD.*(pi/180);                              % in rad
beta = betaD.*(pi/180);                             % in rad
%% SOURCE PANEL STRENGTHS CALCULATION
% Geometric integral (normal [I] and tangential [J])
[I,J] = COMPUTE_IJ_SPM(XC,YC,XB,YB,phi,S);          % Compute geometric integrals
%  Formulation of A matrix
A = zeros(numPan,numPan);                           % Initialize the A matrix
for i = 1:1:numPan                                  % Loop over all i panels
    for j = 1:1:numPan                              % Loop over all j panels
        if (i == j)                                 % If the panels are the same
            A(i,j) = pi;
        else                                        % If panels are not the same
            A(i,j) = I(i,j);                        % Set A equal to geometric integral
        end
    end
end
% formulation of b array
b = zeros(numPan,1);
for i = 1:1:numPan
```

```
    b(i) = -Vinf*2*pi*cos(beta(i));
end
% Compute source panel strengths (lambda) from system of equations
lambda  = A\b;                                          % source strength values
%  source strengths sum calculation
sumLambda = sum(lambda.*S);                             % Check sum of source panel strengths
fprintf('Sum of L: %g\n',sum(lambda.*S));               % Print sum of all source strengths
% Compute velocities
Vt = zeros(numPan,1);                                   % Initialize tangential velocity array
Cp = zeros(numPan,1);                                   % Initialize pressure coefficient array
for i = 1:1:numPan                                      % Loop over all i panels
    addVal  = 0;                                        % Reset the summation value to zero
    for j = 1:1:numPan                                  % Loop over all j panels
        addVal = addVal + (lambda(j)/(2*pi))*(J(i,j));  % Sum all tangential source panel terms
    end
    Vt(i) = Vinf*sin(beta(i)) + addVal;                 % Compute tangential velocity
    Cp(i) = 1-(Vt(i)/Vinf)^2;                           % Compute pressure coefficient
end
%% COMPUTE CL, CM, CP
M = .06;                                                % MP XX 4 digiit airfoil 6412
P = 0.4;
c = 1;                                                  %chord length
x1 = P;
theta = acos(1-((2*x1)/c))
% Calculate DZ/Dx and its integral for computing A0, A1, and A2
% A0 Calculation
fun1 = @(th) (M/(P^2))*((2*P)-(2*(c/2)*(1-cos(th))));  % Integral of DZ/Dx
```

```
Int_1 = integral(fun1,0,theta);

fun2 = @(th) (M/((1-P)^2))*((2*P)-(2*(c/2)*(1-cos(th))));  % Integral of DZ/Dx

Int_2 = integral(fun2,theta,pi);

Int = Int_1 + Int_2;                                % Integral of DZ/Dx

A0 = alpha - (Int/pi);                              %Ao formula

% A1 Calculation

fun3 = @(th) ((M/(P^2))*((2*P)-(2*(c/2)*(1-cos(th)))).*cos(th));
                                                    % Integral of {(DZ/Dx).cos(theta))

Int_3 = integral(fun3,0,theta);                     % limit is 0 to theta

fun4 = @(th) (M/((1-P)^2))*((2*P)-(2*(c/2)*(1-cos(th)))).*cos(th);
                                                    % Integral of {(DZ/Dx).cos(theta)) w.r.t theta

Int_4 = integral(fun4,theta,pi);                    % limit is from theta to pi

Int_5 = Int_3 + Int_4;                              % Integral with limit 0 to pi

A1 = (2/pi)*(Int_5)                                 % A1 formula

% A2 calculation

fun6 = @(th) (M/(P^2))*((2*P)-(2*(c/2)*(1-cos(th)))).*cos(2*th);
                                                    %Integral of {(DZ/Dx).cos(2*theta)) w.r.t theta

Int_6 = integral(fun6,0,theta);                     %limit is 0 to theta

fun7 = @(th) (M/((1-P)^2))*((2*P)-(2*(c/2)*(1-cos(th)))).*cos(2*th);
                                                    %Integral of {(DZ/Dx).cos(2*theta)) w.r.t theta

Int_7 = integral(fun7,theta,pi);                    % limit is from theta to pi

Int_8 = Int_6 + Int_7;                              % Integral limit from 0 to pi

A2 = (2/pi)*(Int_8)                                 % A2 Formula

CL = 2*pi*(A0 + (A1/2))                             % coefficient of lift

CM = (-pi/2)*(A0 + A1 - (A2/2))                     % Coefficient of Moment

CP = (c/4)*(((2*A0) + (2*A1) -A2)/((2*A0)+ A1))     % Center of pressure

AOA_L0 = ((Int/pi)-(Int_5/pi))                      % Angle of attack alpha where Lift= 0

AOA_L0_deg = AOA_L0*(180/pi)                        % in degrees
```

```
%% PLOTs
% Plotting figure of source panels-Figure 1
    figure(1);                                          % Create figure
    cla; hold on; grid on;                              % Get ready for plotting
    set(gcf,'Color','White');                           % Set color to white
    set(gca,'FontSize',12);                             % Set font size
    plot(XB,YB,'k-','LineWidth',3);                     % Plot airfoil panels
    pB = plot(XB,YB,'ko','MarkerFaceColor','k');        % Plot boundary points
    pC = plot(XC,YC,'ko','MarkerFaceColor','m');        % Plot control points
    legend([pB,pC],{'Boundary Points','Control Points'});
    xlabel('X ');                                       % Set X-label
    ylabel('Y ');                                       % Set Y-label
    xlim('auto');                                       % Set X-axis limits to auto
    ylim('auto');                                       % Set Y-axis limits to auto
    axis equal;                                         % Set axes equal
    zoom reset;                                         % Reset zoom
% Cp vectors at airfoil control points-Figure 2
    figure(2);                                          % Create figure
    cla; hold on; grid on;                              % Get ready for plotting
    set(gcf,'Color','White');                           % Set color to white
    set(gca,'FontSize',12);                             % Set font size
    Cps = abs(Cp*0.25);                                 % Scale and make positive all Cp values
    for i = 1:1:length(Cps)                             % Loop over all panels
        X(1) = XC(i);                                   % Control point X-coordinate
        X(2) = XC(i) + Cps(i)*cosd(betaD(i)+AoA);       % Ending X-value based on Cp magnitude
        Y(1) = YC(i);                                   % Control point Y-coordinate
        Y(2) = YC(i) + Cps(i)*sind(betaD(i)+AoA);       % Ending Y-value based on Cp magnitude
```

```
        if (Cp(i) < 0)                              % If pressure coefficient is negative
      p{1} = plot(X,Y,'g-','LineWidth',2);          % Plot
    elseif (Cp(i) >= 0)                             % If pressure coefficient is zero or positive
      p{2} = plot(X,Y,'b-','LineWidth',2);          % Plot
    end
  end
  fill(XB,YB,'k');                                  % Plot the airfoil as black polygon
legend([p{1},p{2}],{'Negetive Cp','Positive Cp'});  % Show legend
  xlabel('X distance');                             % Set X-label
  ylabel('Cp');                                     % Set Y-label
  xlim([-0.5 1.5]);                                 % Set X-axis limits to auto
  ylim([-0.5 0.5]);                                 % Set Y-axis limits to auto
  axis equal;                                       % Set axes equal
  zoom reset;                                       % Reset zoom
% Cp at upper and lower surface- Figure 3
  figure(3);                                        % Create figure
  cla; hold on; grid on;                            % Get ready for plotting
  set(gcf,'Color','White');                         % Set color to white
  set(gca,'FontSize',12);                           % Set font size
  midIndX = floor(length(xFoilCP)/2);               % Airfoil middle index
  midIndS = floor(length(Cp)/2);                    % Airfoil middle index for SPM data
  pSu = plot(XC(1:midIndS),Cp(1:midIndS),'ks','MarkerFaceColor','m');    % Plot Cp
  pSl = plot(XC(midIndS+1:end),Cp(midIndS+1:end),'ks', 'MarkerFaceColor','b');
legend([pSu,pSl], {'Upper Surface','Lower Surface'});
xlabel('X');                                        % Set X-label
ylabel('Cp');                                       % Set Y-label
xlim([0 1]);                                        % Set X-axis limits
```

```
ylim([-3 2]);                          % Set Y-axis limits to auto

set(gca,'Ydir','reverse')              % Reverse direction of Y-axis

title(['Airfoil: NACA 6412']);         % Set title
```