Lifting Flows Over Arbitrary Bodies: Combination of Source and Vortex Panels

The source has zero circulation about any closed path. This zero-circulation property of source sheets has severe consequences for flow representation. Any aerodynamic model consisting only of a freestream and superimposed source sheets will have circulation zero and hence lift also becomes zero. Hence, lifting flows cannot be represented by source sheets alone. This limitation can be overcome by using the combination of source and vortex panels in a panel solution. The source panels basically simulate the airfoil thickness and vortex panels introduce circulation.

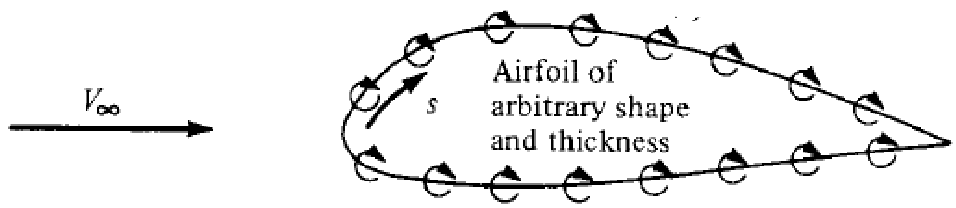

Figure 2.1 Vortex sheet on the surface of a body

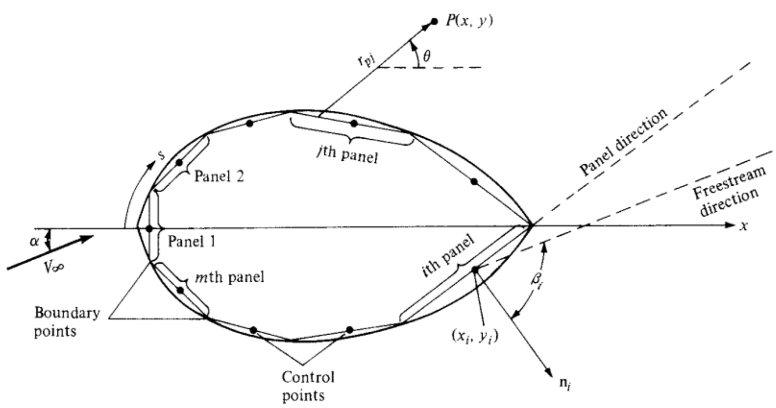

Figure 2.2 Panel distributions over the surface of a body of arbitrary shape

Let us assume the vortex sheet shown in Figure 2.1 by a series of panels, as shown in Figure 2.2. Vortex panels with the same number of source panels are added. Let the vortex strength γ be equal on all panels. Consider the point $P(x, y)$ locate in the flow and let r_{pj} be the distance from any point on the jth panel to P ,as shown in Figure 2.2. The radius r_{pj} makes the angle θ_{pj} with the x-axis. The velocity potential induced to the jth panel, $\Delta\phi_j$ is given by

$$\Delta\phi_j = -\frac{1}{2\pi}\int_j \theta_{pj}\gamma ds_j \tag{2.1}$$

Where,
$$\theta_{pj} = \tan^{-1}\frac{y-y_j}{x-x_j} \tag{2.2}$$

The velocity potential induced at P due to all panels is obtained by summing equation (2.1) over all panels:

$$\phi(P) = \sum_{j=1}^{n}\phi_j = -\frac{\gamma}{2\pi}\int_j \theta_{pj} ds_j \tag{2.3}$$

Let us put P at the control point of the $i\text{th}$ as shown in Figure 2.2. The coordinates of the control point are (x_i, y_i) . Then equations (2.2) and (2.3) become

$$\theta_{ij} = \tan^{-1}\frac{y_i-y_j}{x_i-x_j} \tag{2.4}$$

and
$$\phi(x_i, y_i) = -\frac{\gamma}{2\pi}\int_j \theta_{ij} ds_j \tag{2.5}$$

Equation (2.5) is the expression of velocity potential induced at the control point of $i\text{th}$ panel contributed by all panels.

The normal component of velocity induced at (x_i, y_i) by the vortex panels is

$$V_{vn} = \frac{\partial}{\partial n_i}\left[\phi(x_i, y_i)\right] \tag{2.6}$$

From equations (2.5) and (2.6), we have

$$V_{vn} = -\frac{\gamma}{2\pi}\sum_{\substack{j=1 \\ (j\neq 1)}}^{n}\int_j \frac{\partial\theta_{ij}}{\partial n_i} ds_j \tag{2.7}$$

Here , velocity induced due to panel itself is zero.

The normal component of the flow velocity at $i\text{th}$ control point is the sum due to the freesream (equation 1.8) ,source panels (equation 1.11) and vortex panels (2.7). Applying Neumann impermeability boundary condition, we get
$$V_{\infty,n} + V_{sn} + V_{vn} = 0 \tag{2.8}$$

$$V_\infty \cos\beta_i + \frac{\lambda_i}{2} + \sum_{\substack{j=1 \\ (j\neq 1)}}^{n} \frac{\lambda_j}{2\pi} \int_j \frac{\partial}{\partial n_i}\left(\ln r_{ij}\right) ds_j - \frac{\gamma}{2\pi} \sum_{\substack{j=1 \\ (j\neq 1)}}^{n} \int_j \frac{\partial \theta_{ij}}{\partial n_i} ds_j = 0 \tag{2.9}$$

Let $I_{i,j}$ and $Bn_{i,j}$ be the value of integral of second term and third term respectively when the control point is on the ith panel ,then

$$V_\infty \cos\beta_i + \frac{\lambda_i}{2} + \sum_{\substack{j=1 \\ (j\neq i)}}^{n} \frac{\lambda_j}{2\pi} I_{i,j} - \frac{\gamma}{2\pi} \sum_{\substack{j=1 \\ (j\neq i)}}^{n} Bn_{i,j} = 0 \tag{2.10}$$

Equation (2.10) is a linear algebraic equation with n +1 unknowns , $\lambda_1, \lambda_2, \lambda_3 ....., \lambda_j ..., \lambda_n$ and γ . When this boundary condition is applied to the control points of all the panels , $i = 1, 2, ..., n$,there will be a system of n linear algebraic equations with $n+1$ unknowns.

Similarly, the tangential velocity V_{vs} at the control point of the ith panel induced by all the vortex panels is obtained by differentiating equation (2.5) w.r.t *s*.

$$V_{vs} = \frac{\partial \phi}{\partial s} = -\frac{\gamma}{2} - \frac{\gamma}{2\pi} \sum_{\substack{j=1 \\ (j\neq i)}}^{n} \int_j \frac{\partial}{\partial s} \theta_{ij} ds_j \tag{2.11}$$

Here , velocity induced due to panel itself is $\frac{\gamma}{2}$.

The tangential component V_i of the flow velocity at ith control point is the sum due to the freestream (equation 1.15) ,source panels (equation 1.16) and vortex panels (2.11).

$$V_i = V_{\infty,n} + V_{sn} + V_{vn} = V_\infty \sin\beta_i + \sum_{\substack{j=1 \\ (j\neq i)}}^{n} \frac{\lambda_j}{2\pi} \int_j \frac{\partial}{\partial s}\left(\ln r_{ij}\right) ds_j - \frac{\gamma}{2} - \frac{\gamma}{2\pi} \sum_{\substack{j=1 \\ (j\neq i)}}^{n} \int_j \frac{\partial}{\partial s} \theta_{ij} ds_j \tag{2.12}$$

Let $J_{i,j}$ and $Bt_{i,j}$ be the value of integral of second term and third term respectively when the control point is on the ith panel ,then

$$V_i = V_\infty \sin\beta_i + \sum_{\substack{j=1 \\ (j\neq i)}}^{n} \frac{\lambda_j}{2\pi} J_{i,j} - \frac{\gamma}{2} - \frac{\gamma}{2\pi} \sum_{\substack{j=1 \\ (j\neq i)}}^{n} Bt_{i,j} \tag{2.13}$$

The second boundary condition is the Kutta condition, which states that the pressures on the lower and upper panels at the trailing edge must be equal if the flow is to leave the trailing edge smoothly. If the two pressures are not equal, then the stagnation streamline will wrap itself around the trailing edge. By

Bernoulli's equation, pressure equilibrium also implies equal velocities for incompressible flow. Since the normal velocities are taken to be zero, the boundary condition may now be stated as:

$$(V_i)_1 = -(V_i)_n \tag{2.14}$$

where the negative sign is strictly due to the adopted convention of positive tangential velocities in the direction of increasing node numbering.

$$\begin{gathered}
V_\infty \sin\beta_1 + \sum_{\substack{j=1\\(j\neq i)}}^{n} \frac{\lambda_j}{2\pi} J_{1,j} - \frac{\gamma}{2} - \frac{\gamma}{2\pi} \sum_{\substack{j=1\\(j\neq i)}}^{n} Bt_{1,j} = -\left(V_\infty \sin\beta_n + \sum_{\substack{j=1\\(j\neq i)}}^{n} \frac{\lambda_j}{2\pi} J_{n,j} - \frac{\gamma}{2} - \frac{\gamma}{2\pi} \sum_{\substack{j=1\\(j\neq i)}}^{n} Bt_{n,j} \right) \\
\sum_{\substack{j=1\\(j\neq i)}}^{n} \frac{\lambda_j}{2\pi} J_{1,j} + \sum_{\substack{j=1\\(j\neq i)}}^{n} \frac{\lambda_j}{2\pi} J_{n,j} - \frac{\gamma}{2} - \frac{\gamma}{2\pi} \sum_{\substack{j=1\\(j\neq i)}}^{n} Bt_{1,j} - \frac{\gamma}{2} - \frac{\gamma}{2\pi} \sum_{\substack{j=1\\(j\neq i)}}^{n} Bt_{n,j} = -V_\infty \left(\sin\beta_1 + \sin\beta_n \right)
\end{gathered} \tag{2.15}$$

Considering the geometry given in figure 2.3 ,the above equations can be simplified in the same way as we did previously.

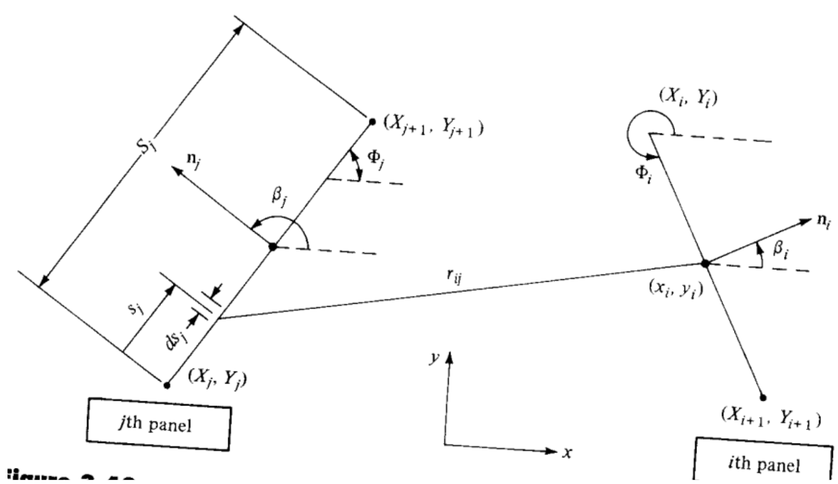

Figure 2.3 Geometry required for the evaluation of $I_{i,j}$

If α is the angle of attack then from geometry,

$$\beta_i = \Phi_i + \frac{\pi}{2} - \alpha \qquad (2.16)$$

```
for i = 1:n
      phi(i) = atan2((ya(i+1)-ya(i)),(xa(i+1)-xa(i)));
      beta(i) = phi(i)+pi/2-alpha;
 end
```

We get ,

$$I_{i,j} = \frac{C}{2}\ln\left(\frac{S_j^{\,2} + 2AS_j + B}{B}\right) + \frac{D - AC}{E}\left(\tan^{-1}\frac{S_j + A}{E} - \tan^{-1}\frac{A}{E}\right) \qquad (2.17)$$

$$J_{i,j} = \frac{D - AC}{2E}\ln\left(\frac{S_j^{\,2} + 2AS_j + B}{B}\right) - C\left(\tan^{-1}\frac{S_j + A}{E} - \tan^{-1}\frac{A}{E}\right) \qquad (2.18)$$

$$Bn_{ij} = J_{i,j} \qquad (2.19)$$

$$Bt_{ij} = -I_{i,j} \qquad (2.20)$$

Where,

$$A = -\left(x_{i-}X_j\right)\cos\Phi_j - \left(y_i - Y_j\right)\sin\Phi_j$$
$$B = \left(x_i - X_j\right)^2 + \left(y_i - Y_j\right)^2$$
$$C = \sin(\Phi_i - \Phi_j)$$
$$D = \left(y_{i-}Y_j\right)\cos\Phi_i - \left(x_i - X_j\right)\sin\Phi_i$$
$$S_j = \sqrt{\left(X_{j+1} - X_j\right)^2 + \left(Y_{j+1} - Y_j\right)^2}$$
$$E = \sqrt{B - A^2}$$

```
for i = 1:n
    for j = 1:n

            Bn(i,j) = J(i,j);
            Bt(i,j) = -I(i,j);

    end
end
```

Substituting the value of $I_{i,j}$, $J_{i,j}$, $Bn_{i,j}$ and $Bt_{i,j}$ for different *i*th panel in equations (2.10) and (2.15),we get n+1 linear algebraic equations with n+1 unknowns, $\lambda_1, \lambda_2, \lambda_3 ....., \lambda_j ..., \lambda_n$ and γ.

$$\frac{\lambda_1}{2}+\frac{\lambda_2}{2\pi}\times \mathrm{I}_{12}+\frac{\lambda_3}{2\pi}\times \mathrm{I}_{13}+.........+\frac{\lambda_n}{2\pi}\times \mathrm{I}_{1n}-\frac{\gamma}{2\pi}\times Bn_{12}-\frac{\gamma}{2\pi}\times Bn_{13}-...-\frac{\gamma}{2\pi}\times Bn_{1n}=-\mathrm{V}_\infty\cos\beta_1$$

$$\frac{\lambda_1}{2\pi}\times \mathrm{I}_{21}+\frac{\lambda_2}{2}+\frac{\lambda_3}{2\pi}\times \mathrm{I}_{23}+.........+\frac{\lambda_n}{2\pi}\times \mathrm{I}_{2n}-\frac{\gamma}{2\pi}\times Bn_{11}-\frac{\gamma}{2\pi}\times Bn_{13}-...-\frac{\gamma}{2\pi}\times Bn_{1n}=-\mathrm{V}_\infty\cos\beta_2$$

$$\frac{\lambda_1}{2\pi}\times \mathrm{I}_{31}+\frac{\lambda_2}{2\pi}\times \mathrm{I}_{32}+\frac{\lambda_3}{2}+.........+\frac{\lambda_n}{2\pi}\times \mathrm{I}_{3n}-\frac{\gamma}{2\pi}\times Bn_{11}-\frac{\gamma}{2\pi}\times Bn_{12}-...-\frac{\gamma}{2\pi}\times Bn_{1n}=-\mathrm{V}_\infty\cos\beta_3$$

$$\begin{matrix} . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . \end{matrix}$$

$$\frac{\lambda_1}{2\pi}\times \mathrm{I}_{n1}+\frac{\lambda_2}{2\pi}\times \mathrm{I}_{n2}+\frac{\lambda_3}{2\pi}\times \mathrm{I}_{n3}+.........+\frac{\lambda_n}{2\pi}-\frac{\gamma}{2\pi}\times Bn_{11}-\frac{\gamma}{2\pi}\times Bn_{12}-...-\frac{\gamma}{2\pi}\times Bn_{1n-1}=-\mathrm{V}_\infty\cos\beta_n$$

$$\frac{\lambda_2}{2\pi}\left(\mathrm{J}_{12}+\mathrm{J}_{n2}\right)+\frac{\lambda_2}{2\pi}\left(\mathrm{J}_{12}+\mathrm{J}_{n2}\right)+...+\frac{\lambda_n}{2\pi}\left(\mathrm{J}_{12}+\mathrm{J}_{n2}\right)+\frac{\lambda_n}{2\pi}\left(\mathrm{J}_{12}+\mathrm{J}_{n2}\right) \qquad =-\mathrm{V}_\infty\left(\sin\beta_1+\sin\beta_n\right)$$

$$-\frac{\gamma}{2}-\frac{\gamma}{2}-\frac{\gamma}{2\pi}\left(Bt_{1,2}+Bt_{n,2}..+Bt_{n,n-1}\right)$$

Converting above equation in matrix form, we get

$$\begin{bmatrix} a_{11} & . & . & . & . & a_{1n+1} \\ a_{21} & . & . & . & . & a_{2n+1} \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ a_{n+1,1} & a_{n+1,2} & . & . & . & a_{n+1,n+1} \end{bmatrix}\begin{bmatrix} \lambda_1 \\ \lambda_2 \\ . \\ . \\ \lambda_n \\ \gamma \end{bmatrix}=\begin{bmatrix} -V_\infty\cos\beta_1 \\ -V_\infty\cos\beta_2 \\ \\ \\ -V_\infty\cos\beta_n \\ -V_\infty(\sin\beta_1+\sin\beta_n) \end{bmatrix}$$

It can be written as

$$P\lambda = Q \tag{1.21}$$

By solving above matrix,we get values of source strength and vortex strength of each panel i.e. $\lambda_1, \lambda_2, \lambda_3 ....., \lambda_j ..., \lambda_n \text{and} \gamma$

```
for i = 1:n
    bb(i) = 0;
end
for i = 1:n
    for j = 1:n
```

```
        bb(i) = bb(i)+Bn(i,j);

    end
end



    cc = 0;


    for j = 1:n

        cc = cc+Bt(1,j)+Bt(n,j);

    end



% solving matrix equation
P = zeros(n+1,n+1);
for i = 1:n+1
    for j = 1:n+1
        if i<=n && j<=n
            P(i,j)=I(i,j);
        elseif i<=n && j==n+1
            P(i,j)=bb(i);
        elseif i==n+1 && j<=n
            P(i,j)=J(1,j)+J(n,j);
        else
            P(i,j)=cc;
        end
    end
end

Q = zeros(n+1,1);
U = 1;

for i = 1:n+1
    if i<=n
    Q(i,1) = -2*pi*U*cos(eta(i));
    else
        Q(i,1) = -2*pi*U*(sin(eta(1))+sin(eta(n)));
    end
end

lamda = zeros(n+1,1);
lamda = mldivide(P,Q);
```

Substituting the obtained values of λ and γ in equation (2.13),we get the velocities $V_1, V_2, V_3...., V_n$ .In turn, the pressure coefficients $C_{p,1}, C_{p,2}, C_{p,3}....., C_{p,n}$ are obtained directly from equation (1.19)

```
% calculating tangential velocity accordng to the equation (3.156)
for i = 1:n
b(i) = 0.0;
end
for i = 1:n
    for j = 1:n

        b(i) = b(i)+lamda(j,1)/2/pi*J(i,j);
        end
    end

for i = 1:n
    vv(i) = 0.0;
end
for i = 1:n
    for j = 1:n

        vv(i) = vv(i)+lamda(n+1,1)/2/pi*Bt(i,j);
        end
    end



  for i = 1:n
        v(i) = U*sin(eta(i))+b(i)+vv(i);
      ut = v(i)/U;
  cp(i) = 1-ut^2; % calculating cp at each control points

  end
```

The matlab program to find pressure distribution over an airfoil using combination of source and vortex is given as follows :

```
clc
clear all
close all
nacaseries = input('Enter the 4-digit naca series = ');
 c = input('Enter the chord length = ');
 n = input('Enter the number of nodes = ');
 a = input('Enter the angle of attack in degree = ');
 s = num2str(nacaseries);

 % creating points on airfoil
```

```
if numel(s)==2
    s1 = str2double(s(1));
 s2 = str2double(s(2));
 m=0;p=0;t=(10*s1+s2)*0.01;
else
    s1 = str2double(s(1));
 s2 = str2double(s(2));
 s3 = str2double(s(3));
 s4 = str2double(s(4));
    m = s1*0.01; p = s2*0.1 ; t = (10*s3+s4)*0.01;
end


for i= n:-1:1

    theta = (i-n)*2*pi/n;
    x = 0.5*c*(1+cos(theta));

if(x/c)<p
    yc(n+1-i) = m*c/p^2*(2*p*(x/c)-(x/c)^2);
    dydx(n+1-i) = (2*m/p^2)* (p-x/c);
    beta(n+1-i) = atan(dydx(n+1-i));
else
    yc(n+1-i) = m*c/(1-p)^2 * ((1-2*p)+2*p*(x/c)-(x/c)^2);
    dydx(n+1-i) = (2*m/(1-p)^2)* (p-x/c);
    beta(n+1-i) = atan(dydx(n+1-i));
end
yt=5*t*c*(0.2969*sqrt(x/c)-0.1260*(x/c)...
    -0.3516*(x/c)^2+0.2843*(x/c)^3-0.1036*(x/c)^4);

if(i<(0.5*n+1))
    xa(n+1-i)=x - yt*sin(beta(n+1-i));
    ya(n+1-i)=yc(n+1-i)+yt*cos(beta(n+1-i));
else
    xa(n+1-i)=x + yt*sin(beta(n+1-i));
    ya(n+1-i)=yc(n+1-i)-yt*cos(beta(n+1-i));
end

end
xa(n+1)= c;
ya(n+1) = 0;
yc(n+1) = 0;  % trailing edge

% computing control points and panel length
for i = 1:n
    xmid(i) = (xa(i)+xa(i+1))/2;
    ymid(i) = (ya(i)+ya(i+1))/2;
    Sj(i) = sqrt((xa(i+1)-xa(i))^2+(ya(i+1)-ya(i))^2);
end

 % Calculating angles and integrals
 alpha = a*pi/180;
 for i = 1:n
      phi(i) = atan2((ya(i+1)-ya(i)),(xa(i+1)-xa(i)));
      eta(i) = phi(i)+pi/2-alpha;
```

```
 end



for i = 1:n
    for j = 1:n
           if i~=j

        A=-(xmid(i)-xa(j))*cos(phi(j))-(ymid(i)-ya(j))*sin(phi(j));
        B = (xmid(i)-xa(j))^2+(ymid(i)-ya(j))^2;
        C = sin(phi(i)-phi(j));
        D = (ymid(i)-ya(j))*cos(phi(i))-(xmid(i)-xa(j))*sin(phi(i));
        E = sqrt(B-A^2);
        I(i,j)              =              C/2*log((Sj(j)^2+2*A*Sj(j)+B)/B)+(D-
A*C)/E*(atan2(Sj(j)+A,E)-atan2(A,E));
        J(i,j)           =           (D-A*C)/(2*E)*log((Sj(j)^2+2*A*Sj(j)+B)/B)-
C*(atan2(Sj(j)+A,E)-atan2(A,E));
         else
             I(i,j) = pi;
             J(i,j) = 0;
         end
    end

end

for i = 1:n
    for j = 1:n

             Bn(i,j) = J(i,j);
             Bt(i,j) = -I(i,j);


    end
end
for i = 1:n
    bb(i) = 0;
end
for i = 1:n
    for j = 1:n

        bb(i) = bb(i)+Bn(i,j);

    end
end




    cc = 0;



    for j = 1:n
```

```
        cc = cc+Bt(1,j)+Bt(n,j);

    end



% solving matrix equation
P = zeros(n+1,n+1);
for i = 1:n+1
    for j = 1:n+1
        if i<=n && j<=n
            P(i,j)=I(i,j);
        elseif i<=n && j==n+1
            P(i,j)=bb(i);
        elseif i==n+1 && j<=n
            P(i,j)=J(1,j)+J(n,j);
        else
            P(i,j)=cc;
        end
    end
end

Q = zeros(n+1,1);
U = 1;

for i = 1:n+1
    if i<=n
    Q(i,1) = -2*pi*U*cos(eta(i));
    else
        Q(i,1) = -2*pi*U*(sin(eta(1))+sin(eta(n)));
    end
end

lamda = zeros(n+1,1);
lamda = mldivide(P,Q);


% calculating tangential velocity accordng to the equation (3.156)
for i = 1:n
b(i) = 0.0;
end
for i = 1:n
    for j = 1:n

        b(i) = b(i)+lamda(j,1)/2/pi*J(i,j);
        end
    end

for i = 1:n
    vv(i) = 0.0;
end
for i = 1:n
    for j = 1:n
```

```
        vv(i) = vv(i)+lamda(n+1,1)/2/pi*Bt(i,j);
        end
    end



  for i = 1:n
        v(i) = U*sin(eta(i))+b(i)+vv(i);
      ut = v(i)/U;
  cp(i) = 1-ut^2; % calculating cp at each control points

  end


     plot(xa(1:n/2),cp(1:n/2),'-*r')
     set(gca,'Ydir','reverse')
     hold on
     plot(xa(n/2+1:n),cp(n/2+1:n),'-*b')

    hold on
     plot(xa,ya,'-k')
    xlabel('x')
    ylabel('cp')
    title('cp vs x')
%axis([0 1,-2 2])
```

The plot of cp vs x of NACA0012 airfoil at 9° AOA using panel method