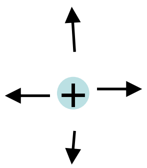
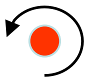
Panel Methods

Source and Vortex

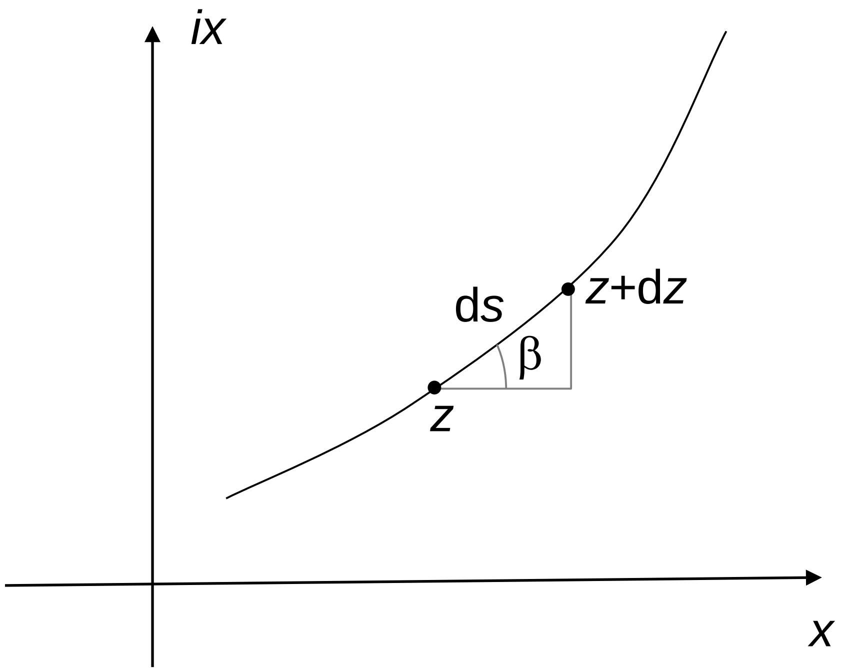
- Point Source



- Point Vortex



dz in Polar Form



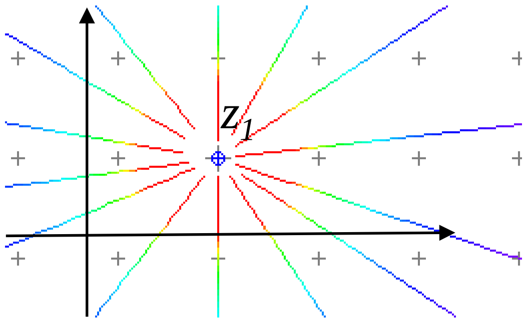
Panels

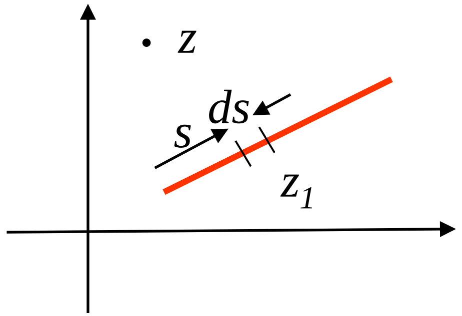
Singularity distributed along a line

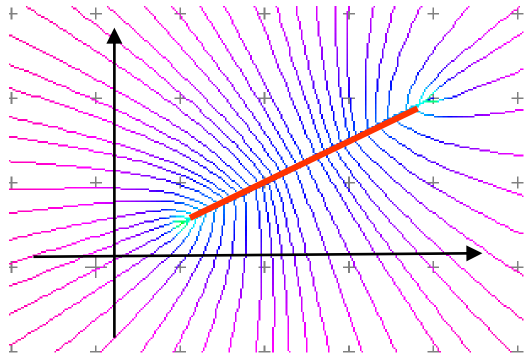
<u>*Example: The Source Panel (or Sheet)*</u>

Consider a point source ⊕ $$W(z) = \frac{q}{2\pi(z - z_1)}$$

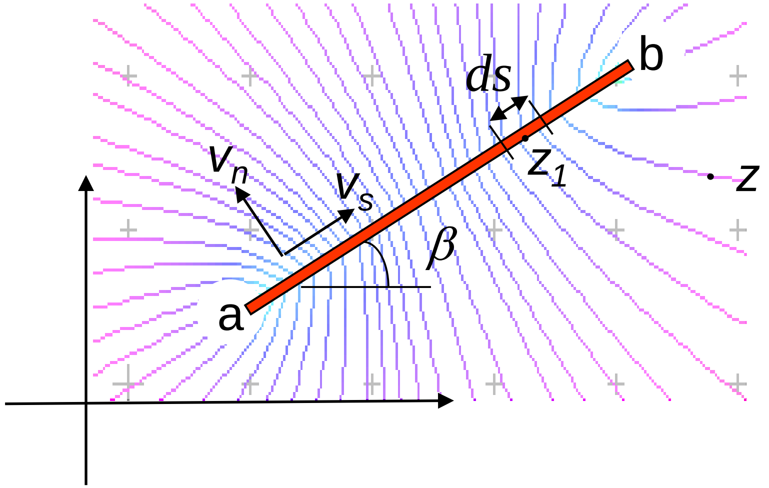
Imagine spreading the source along a line. We would then end up with a certain strength per unit length $q(s)$ that could vary with distance s along the line.







Constant Strength Source Panel



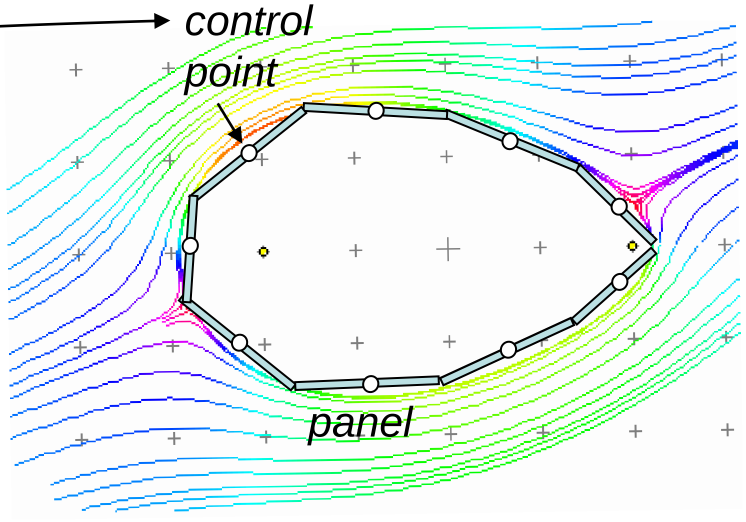
The panel is not a solid boundary to the flow. To make it behave like one you would set the strength q so that the total v_n (due to the panel and the flow) is zero

Vortex?

A Simple Source Panel Method

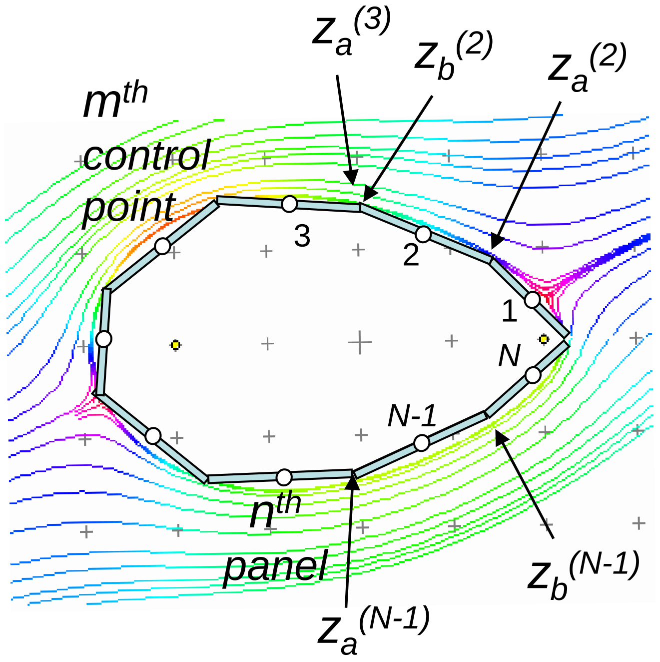
For flow past an arbitrary body

- Break up the body surface into *N* straight panels.
- Write an expression for the normal component of velocity at the middle of the panel from the sum of all the velocities produced by the panels and the free stream. Gives *N* expressions.
- Given that each expression must be equal to zero, solve the *N* equations for the *N* strengths



Defining the *N* Panels

- Number the panels anticlockwise, 1 to N
- Define N coordinates z_a that identify the start of every panel (going counter clockwise), and z_b that identify the end of every panel.



- Each panel has a slope $\dfrac{dz}{ds} = e^{i\beta} = \dfrac{z_b - z_a}{\left|z_b - z_a\right|}$

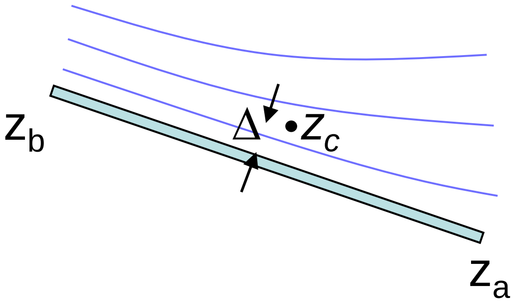
  So, if $W(z)$ is the velocity of the whole flow,

  $-\operatorname{Im}\left\{W(z)\dfrac{dz}{ds}\right\}$ is the component normal to the panel

- We pick a control point very close to the center of the panel at

$$z_c = \tfrac{1}{2}\left(z_a + z_b\right) - i\varepsilon\left(z_b - z_a\right)$$

Center point *Displacement Δ*



ε<<1 (say 0.001)

Completing the Method

Velocity produced by whole flow is

$$W(z) = W_\infty + \sum_{n=1}^{N} q^{(n)} \frac{1}{2\pi} \log_e\left(\frac{z - z_a^{(n)}}{z - z_b^{(n)}}\right) \frac{ds}{dz_1}\bigg|^{(n)} \qquad 1$$

Velocity at the control point of the m[th] panel $z_c^{(m)}$ in panel aligned components is

$$= W_\infty \frac{dz_1}{ds}\bigg|^{(m)} + \sum_{n=1}^{N} q^{(n)} \underbrace{\frac{1}{2\pi} \log_e\left(\frac{z_c^{(m)} - z_a^{(n)}}{z_c^{(m)} - z_b^{(n)}}\right) \frac{ds}{dz_1}\bigg|^{(n)} \frac{dz_1}{ds}\bigg|^{(m)}}_{C^{(m,n)}}$$

So, velocity normal to the m[th] panel is

$$= \mathrm{Im}\left\{ \frac{dz_1}{ds}\bigg|^{(m)} W_\infty \right\} + \sum_{n=1}^{N} q^{(n)} \,\mathrm{Im}\{C^{(m,n)}\}$$

Velocity parallel to the m[th] panel is

$$= \mathrm{Re}\left\{ \frac{dz_1}{ds}\bigg|^{(m)} W_\infty \right\} + \sum_{n=1}^{N} q^{(n)} \,\mathrm{Re}\{C^{(m,n)}\} \qquad 2$$

We want the normal velocity to be zero, so this is what we use to get the *q*'s

We write

$$-\mathrm{Im}\left\{ \frac{dz_1}{ds}\bigg|^{(m)} W_\infty \right\} = \sum_{n=1}^{N} q^{(n)} \,\mathrm{Im}\{C^{(m,n)}\}$$

Or 1xN result matrix (known) = 1xN matrix of strengths (unknown) x NxN matrix of ceoffs (known)

Once we have solved this for the *q*'s we can use eqn. 2 to get the velocities along the body surface, or eqn. 1 to get them anywhere else

Computational Steps

- Define coordinates of start and end of panels z_a and z_b $\boxed{z_a^{(n)}, z_b^{(n)}}$
- Compute the panel slopes $\boxed{\dfrac{dz}{ds} = e^{i\beta} = \dfrac{z_b - z_a}{|z_b - z_a|}}$
- Put the control points next to the panel centers $\boxed{z_c = \frac{1}{2}(z_a + z_b) - i\varepsilon(z_b - z_a)}$
- Determine the component of W_∞ normal to each panel $\boxed{-\operatorname{Im}\left\{ \left.\dfrac{dz_1}{ds}\right|^{(m)} W_\infty \right\}}$
- Determine the influence coefficients $\boxed{C^{(m,n)} = \dfrac{1}{2\pi} \log_e \left(\dfrac{z_c^{(m)} - z_a^{(n)}}{z_c^{(m)} - z_b^{(n)}} \right) \left.\dfrac{ds}{dz_1}\right|^{(n)} \left.\dfrac{dz_1}{ds}\right|^{(m)}}$
- Solve the matrix problem, i.e. matrix divide $\boxed{\operatorname{Im}\{C^{(m,n)}\}}$ by $\boxed{-\operatorname{Im}\left\{ \left.\dfrac{dz_1}{ds}\right|^{(m)} W_\infty \right\}}$
- Compute the flow velocities and pressures

ConstantSourcePanel.m

Matlab Code

$z_a^{(n)}, z_b^{(n)}, dz_1/ds|^{(n)}$

$z_c^{(n)}$

$C^{(m,n)}$

Matrix div.

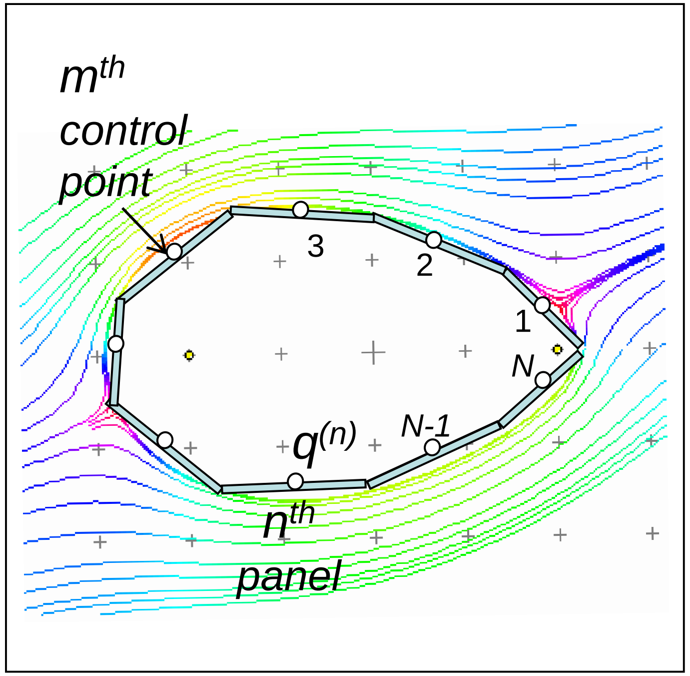
```
clear all;
%Circular cylinder example, radius 2, 35 panels
npanels=35;r=2;winf=1;
z=r*exp(i*[2*pi/npanels:2*pi/npanels:2*pi]);
a=[1:npanels];b=[2:npanels 1];
dzds=(z(b)-z(a))./abs(z(b)-z(a));

eps=0.0001;
zc=(z(a)+z(b))/2-i*eps*(z(b)-z(a)); %control points

cm=zeros(npanels);
for m=1:npanels
    cm(:,m)=log((zc(m)-z(a))./(zc(m)-z(b)))/2/pi./dzds(a)*dzds(m);
end
res=imag(-winf*dzds);
q=res/imag(cm);

ut=real(q*cm+winf*dzds);
cp=1-ut.^2/abs(winf).^2;
figure
plot(angle(zc)*180/pi,cp);
```

$-\operatorname{Im}\left\{\left.\frac{dz_1}{ds}\right|^{(m)} W_\infty\right\}$ Result matrix

$\operatorname{Re}\left\{\left.\frac{dz_1}{ds}\right|^{(m)} W_\infty\right\} + \sum_{n=1}^{N} q^{(n)} \operatorname{Re}\{C^{(m,n)}\}$ Velocities along body surface

$$-\operatorname{Im}\left\{ \left.\frac{dz_1}{ds}\right|^{(m)} W_\infty \right\} = \sum_{n=1}^{N} q^{(n)} \operatorname{Im}\{C^{(m,n)}\}$$

$$\begin{pmatrix} res(1) \\ res(2) \\ \vdots \\ res(m) \\ \vdots \\ res(N) \end{pmatrix} = \begin{pmatrix} \operatorname{Im}\{C^{(1,1)}\} & \operatorname{Im}\{C^{(1,2)}\} & \cdots & \operatorname{Im}\{C^{(1,n)}\} & \cdots & \operatorname{Im}\{C^{(1,N)}\} \\ \operatorname{Im}\{C^{(2,1)}\} & \cdot & & & & \\ \vdots & & \cdot & & & \\ \operatorname{Im}\{C^{(m,1)}\} & & & \operatorname{Im}\{C^{(m,n)}\} & & \\ \vdots & & & & \cdot & \\ \operatorname{Im}\{C^{(N,1)}\} & & & & & \operatorname{Im}\{C^{(N,N)}\} \end{pmatrix} \begin{pmatrix} q^{(1)} \\ q^{(2)} \\ \vdots \\ q^{(n)} \\ \vdots \\ q^{(N)} \end{pmatrix}$$




```
zc=(z(a)+z(b))/2-i*eps*(z(b)-z(a)); %control points

cm=zeros(npanels);
for m=1:npanels
    cm(:,m)=log((zc(m)-z(a))./(zc(m)-z(b)))/2/pi./dzds(a)*dzds(m);
end
res=imag(-winf*dzds);
q=res/imag(cm);

ut=real(q*cm+winf*dzds);
cp=1-ut.^2/abs(winf).^2;
```

ConstantSourcePanel.m

Matlab Code Ideas:

1. Non-Uniform Free Stream

E.g. Suppose free stream includes, say a doublet outside the body at a location x=5, so

$$W_\infty = 1 + \frac{10}{(z-5)^2}$$

```
clear all;
%Circular cylinder example, radius 2, 35 panels
npanels=35;r=2;winf=1;
z=r*exp(i*[2*pi/npanels:2*pi/npanels:2*pi]);
a=[1:npanels];b=[2:npanels 1];
dzds=(z(b)-z(a))./abs(z(b)-z(a));

eps=0.0001;
zc=(z(a)+z(b))/2-i*eps*(z(b)-z(a)); %control points

cm=zeros(npanels);
for m=1:npanels
    cm(:,m)=log((zc(m)-z(a))./(zc(m)-z(b)))/2/pi./dzds(a)*dzds(m);
end
res=imag(-winf*dzds);
q=res/imag(cm);

ut=real(q*cm+winf*dzds);
cp=1-ut.^2/abs(winf).^2;
figure
plot(angle(zc)*180/pi,cp);
```

ConstantSourcePanel.m

Matlab Code Ideas:

2. More than one body

E.g. Suppose we have two circles

```
clear all;
%Circular cylinder example, radius 2, 35 panels
npanels=35;r=2;winf=1;
z=r*exp(i*[2*pi/npanels:2*pi/npanels:2*pi]);
a=[1:npanels];b=[2:npanels 1];
dzds=(z(b)-z(a))./abs(z(b)-z(a));

eps=0.0001;
zc=(z(a)+z(b))/2-i*eps*(z(b)-z(a)); %control points

cm=zeros(npanels);
for m=1:npanels
    cm(:,m)=log((zc(m)-z(a))./(zc(m)-z(b)))/2/pi./dzds(a)*dzds(m);
end
res=imag(-winf*dzds);
q=res/imag(cm);

ut=real(q*cm+winf*dzds);
cp=1-ut.^2/abs(winf).^2;
figure
plot(angle(zc)*180/pi,cp);
```

ConstantSourcePanel.m

Matlab Code Ideas:

3. Use more sophisticated panels

E.g. Panels with linearly varying strength

```
clear all;
%Circular cylinder example, radius 2, 35 panels
npanels=35;r=2;winf=1;
z=r*exp(i*[2*pi/npanels:2*pi/npanels:2*pi]);
a=[1:npanels];b=[2:npanels 1];
dzds=(z(b)-z(a))./abs(z(b)-z(a));

eps=0.0001;
zc=(z(a)+z(b))/2-i*eps*(z(b)-z(a)); %control points

cm=zeros(npanels);
for m=1:npanels
    cm(:,m)=log((zc(m)-z(a))./(zc(m)-z(b)))/2/pi./dzds(a)*dzds(m);
end
res=imag(-winf*dzds);
q=res/imag(cm);

ut=real(q*cm+winf*dzds);
cp=1-ut.^2/abs(winf).^2;
figure
plot(angle(zc)*180/pi,cp);
```

LinearSourcePanel.m

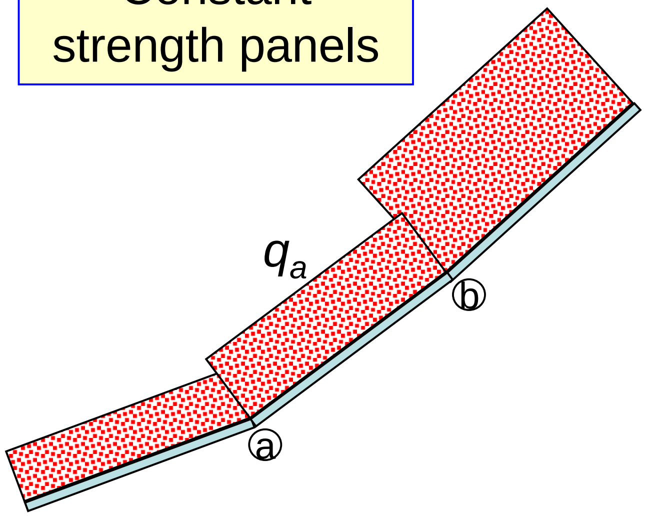
3. Linear Source Panel Method

E.g. Panels with linearly varying strength

```
clear all;
%Circular cylinder example, radius 2, 35 panels
npanels=35;r=2;winf=1;
z=r*exp(i*[2*pi/npanels:2*pi/npanels:2*pi]);
a=[1:npanels];b=[2:npanels 1];c=[3:npanels 1 2];
dzds=(z(b)-z(a))./abs(z(b)-z(a));

eps=0.0001;
zc=(z(a)+z(b))/2-i*eps*(z(b)-z(a)); %control points

cm=zeros(npanels);
for m=1:npanels
cm(:,m)=(((zc(m)-z(a))./(z(b)-z(a)).*log((zc(m)-z(a))./(zc(m)-z(b)))
        -((zc(m)-z(c))./(z(b)-z(c)).*log((zc(m)-z(c))./(zc(m)-z(b)))
end
res=imag(-winf*dzds);
q=res/imag(cm);

ut=real(q*cm+winf*dzds);
cp=1-ut.^2/abs(winf).^2;
figure
plot(angle(zc)*180/pi,cp);
```

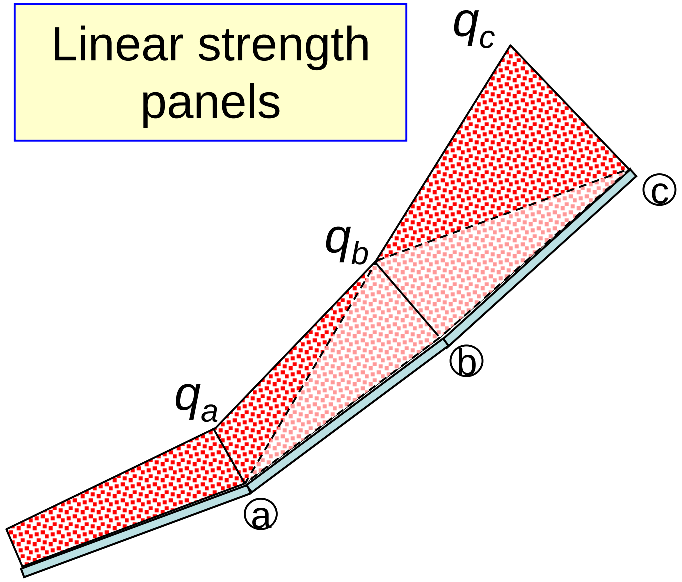
3. Linear Source Panels

Constant strength panels



Influence of q_a depends only panel ab

Linear strength panels



Influence of q_b depends on panel ab and panel bc

LinearSourcePanel.m

3. Linear Source Panel Method

E.g. Panels with linearly varying strength

```
clear all;
%Circular cylinder example, radius 2, 35 panels
npanels=35;r=2;winf=1;
z=r*exp(i*[2*pi/npanels:2*pi/npanels:2*pi]);
a=[1:npanels];b=[2:npanels 1];c=[3:npanels 1 2];
dzds=(z(b)-z(a))./abs(z(b)-z(a));

eps=0.0001;
zc=(z(a)+z(b))/2-i*eps*(z(b)-z(a)); %control points

cm=zeros(npanels);
for m=1:npanels
cm(:,m)=(((zc(m)-z(a))./(z(b)-z(a)).*log((zc(m)-z(a))./(zc(m)-z(b)))
        -((zc(m)-z(c))./(z(b)-z(c)).*log((zc(m)-z(c))./(zc(m)-z(b)))
end
res=imag(-winf*dzds);
q=res/imag(cm);

ut=real(q*cm+winf*dzds);
cp=1-ut.^2/abs(winf).^2;
figure
plot(angle(zc)*180/pi,cp);
```

LinearSourcePanel.m

Matlab Code Ideas:

4.Change to vortex panel method

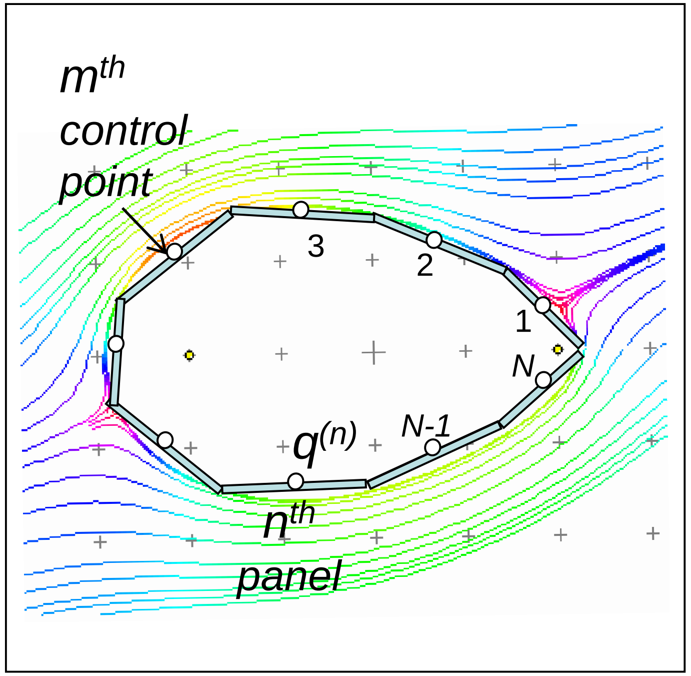
$$W(z) = \frac{q}{2\pi(z - z_1)}$$

$$W(z) = \frac{-i\Gamma}{2\pi(z - z_1)}$$

```
clear all;
%Circular cylinder example, radius 2, 35 panels
npanels=35;r=2;winf=1;
z=r*exp(i*[2*pi/npanels:2*pi/npanels:2*pi]);
a=[1:npanels];b=[2:npanels 1];c=[3:npanels 1 2];
dzds=(z(b)-z(a))./abs(z(b)-z(a));

eps=0.0001;
zc=(z(a)+z(b))/2-i*eps*(z(b)-z(a)); %control points

cm=zeros(npanels);
for m=1:npanels
cm(:,m)=(((zc(m)-z(a))./(z(b)-z(a)).*log((zc(m)-z(a))./(zc(m)-z(b)))
        -((zc(m)-z(c))./(z(b)-z(c)).*log((zc(m)-z(c))./(zc(m)-z(b)))
end
res=imag(-winf*dzds);
q=res/imag(cm);

ut=real(q*cm+winf*dzds);
cp=1-ut.^2/abs(winf).^2;
figure
plot(angle(zc)*180/pi,cp);
```

$$-\operatorname{Im}\left\{ \left.\frac{dz_1}{ds}\right|^{(m)} W_\infty \right\} = \sum_{n=1}^{N} q^{(n)} \operatorname{Im}\{C^{(m,n)}\}$$

$$\begin{pmatrix} res(1) \\ res(2) \\ \vdots \\ res(m) \\ \vdots \\ res(N) \end{pmatrix} = \begin{pmatrix} \operatorname{Im}\{C^{(1,1)}\} & \operatorname{Im}\{C^{(1,2)}\} & \cdots & \operatorname{Im}\{C^{(1,n)}\} & \cdots & \operatorname{Im}\{C^{(1,N)}\} \\ \operatorname{Im}\{C^{(2,1)}\} & \cdot & & & & \\ \vdots & & \cdot & & & \\ \operatorname{Im}\{C^{(m,1)}\} & & & \operatorname{Im}\{C^{(m,n)}\} & & \\ \vdots & & & & \cdot & \\ \operatorname{Im}\{C^{(N,1)}\} & & & & & \operatorname{Im}\{C^{(N,N)}\} \end{pmatrix} \begin{pmatrix} q^{(1)} \\ q^{(2)} \\ \vdots \\ q^{(n)} \\ \vdots \\ q^{(N)} \end{pmatrix}$$



LinearVortexPanel.m
(see also ConstantVortexPanel.m)

Matlab Code Ideas:
4. Vortex panel method

```
clear all;
%Circular cylinder example, radius 2, 35 panels
npanels=35;r=2;winf=1;gamma=-10;
z=r*exp(i*[2*pi/npanels:2*pi/npanels:2*pi]);
a=[1:npanels];b=[2:npanels 1];c=[3:npanels 1 2];
dzds=(z(b)-z(a))./abs(z(b)-z(a));

eps=0.0001;
zc=(z(a)+z(b))/2-i*eps*(z(b)-z(a)); %control points

cm=zeros(npanels);
for m=1:npanels
cm(:,m)=-i*(((zc(m)-z(a))./(z(b)-z(a)).*log((zc(m)-z(a))./(zc(m)-z(b
           -((zc(m)-z(c))./(z(b)-z(c)).*log((zc(m)-z(c))./(zc(m)-z(b
end
res=imag(-winf*dzds);
cm1=imag(cm);cm1(:,end)=0.5*(abs(z(b)-z(a))+abs(z(c)-z(b)));
res(end)=gamma;
q=res/cm1;

ut=real(q*cm+winf*dzds);
cp=1-ut.^2/abs(winf).^2;
figure
```

LinearVortexPanel.m
(see also ConstantVortexPanel.m)

Matlab Code Ideas:
5. Set a Kutta Condition

Kutta condition requires that surface vorticity at trailing edge is zero.

```
clear all;
%Circular cylinder example, radius 2, 35 panels
npanels=35;r=2;winf=1;gamma=-10;
z=r*exp(i*[2*pi/npanels:2*pi/npanels:2*pi]);
a=[1:npanels];b=[2:npanels 1];c=[3:npanels 1 2];
dzds=(z(b)-z(a))./abs(z(b)-z(a));

eps=0.0001;
zc=(z(a)+z(b))/2-i*eps*(z(b)-z(a)); %control points

cm=zeros(npanels);
for m=1:npanels
cm(:,m)=-i*(((zc(m)-z(a))./(z(b)-z(a)).*log((zc(m)-z(a))./(zc(m)-z(b
           -((zc(m)-z(c))./(z(b)-z(c)).*log((zc(m)-z(c))./(zc(m)-z(b
end
res=imag(-winf*dzds);
cm1=imag(cm);cm1(:,end)=0.5*(abs(z(b)-z(a))+abs(z(c)-z(b)));
res(end)=gamma;
q=res/cm1;

ut=real(q*cm+winf*dzds);
cp=1-ut.^2/abs(winf).^2;
figure
```

LinearVortexPanelKutta.m

Matlab Code Ideas:

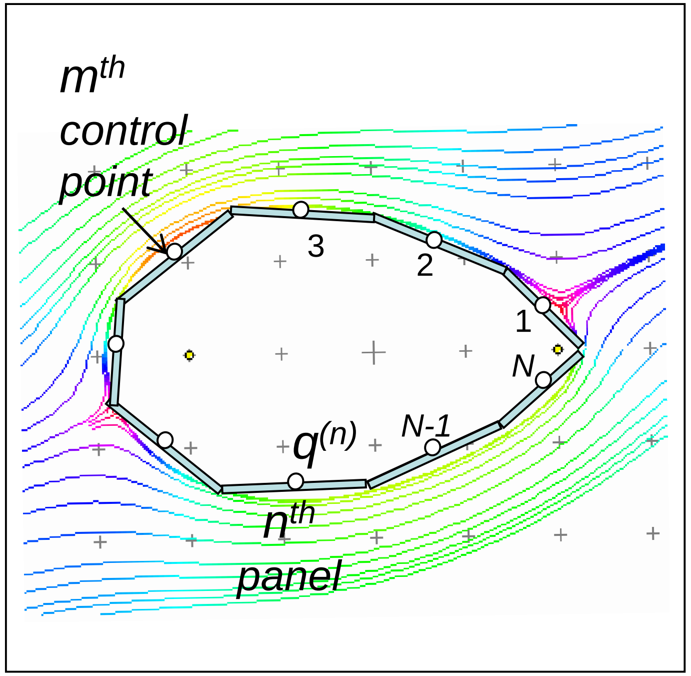
5. Kutta Condition Code

Kutta condition requires that surface vorticity at trailing edge is zero.

```
clear all;
%Circular cylinder example, radius 2, 35 panels
npanels=35;r=2;winf=1;k=npanels-1;
z=r*exp(i*[2*pi/npanels:2*pi/npanels:2*pi]);
a=[1:npanels];b=[2:npanels 1];c=[3:npanels 1 2];
dzds=(z(b)-z(a))./abs(z(b)-z(a));

eps=0.0001;
zc=(z(a)+z(b))/2-i*eps*(z(b)-z(a)); %control points

cm=zeros(npanels);
for m=1:npanels
cm(:,m)=-i*(((zc(m)-z(a))./(z(b)-z(a)).*log((zc(m)-z(a))./(zc(m)-z(k
           -((zc(m)-z(c))./(z(b)-z(c)).*log((zc(m)-z(c))./(zc(m)-z(k
end
res=imag(-winf*dzds);
cm1=imag(cm);cm1(:,end)=0;cm1(k,end)=1;
res(end)=0;
q=res/cm1;

ut=real(q*cm+winf*dzds);
cp=1-ut.^2/abs(winf).^2;
figure
```

$$-\operatorname{Im}\left\{ \left.\frac{dz_1}{ds}\right|^{(m)} W_\infty \right\} = \sum_{n=1}^{N} q^{(n)} \operatorname{Im}\{C^{(m,n)}\}$$

$$\begin{pmatrix} res(1) \\ res(2) \\ \vdots \\ res(m) \\ \vdots \\ res(N) \end{pmatrix} = \begin{pmatrix} \operatorname{Im}\{C^{(1,1)}\} & \operatorname{Im}\{C^{(1,2)}\} & \cdots & \operatorname{Im}\{C^{(1,n)}\} & \cdots & \operatorname{Im}\{C^{(1,N)}\} \\ \operatorname{Im}\{C^{(2,1)}\} & \cdot & & & & \\ \vdots & & \cdot & & & \\ \operatorname{Im}\{C^{(m,1)}\} & & & \operatorname{Im}\{C^{(m,n)}\} & & \\ \vdots & & & & \cdot & \\ \operatorname{Im}\{C^{(N,1)}\} & & & & & \operatorname{Im}\{C^{(N,N)}\} \end{pmatrix} \begin{pmatrix} q^{(1)} \\ q^{(2)} \\ \vdots \\ q^{(n)} \\ \vdots \\ q^{(N)} \end{pmatrix}$$



Something to watch out for…

- The control-point equation

$$z_c = \underbrace{\tfrac{1}{2}\left(z_a + z_b\right)}_{\textit{Center point}} - \underbrace{i\varepsilon(z_b - z_a)}_{\textit{Displacement } \Delta}$$

  assumes that as you move from a to b you are progressing counter-clockwise around the body surface. (For clockwise you need to reverse the sign before i).