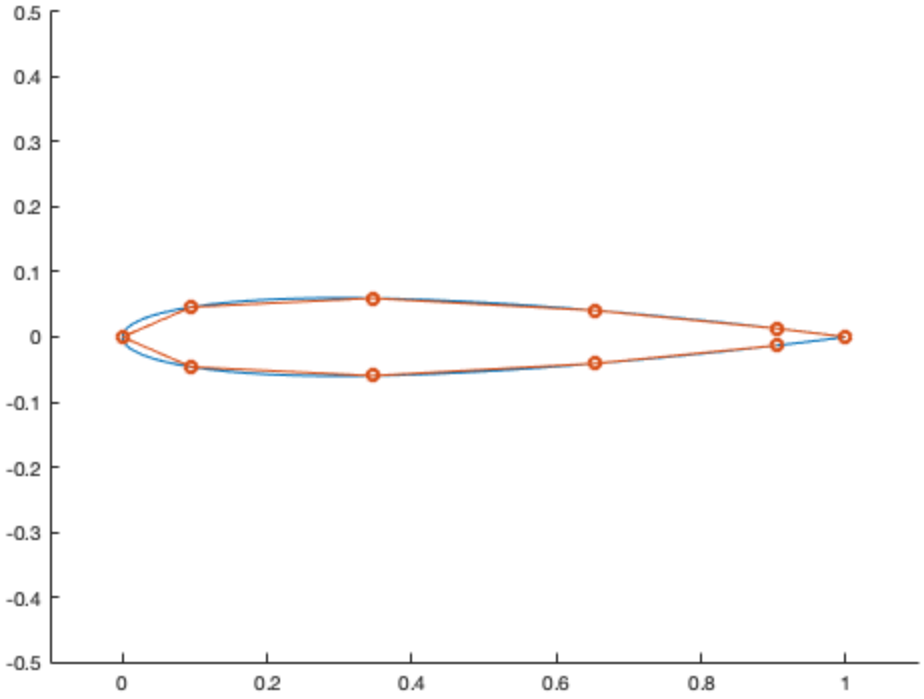
```
% Source-Panel Method on Arbitrary Airfoil
clear; clc;

% Import the airfoil from the .dat information *airfoil tools

airfoil_data = importdata('NACA0012.txt');

for i = 1:size(airfoil_data)
    x_airfoil(i) = airfoil_data(i,1);
    y_airfoil(i) = airfoil_data(i,2);
end

%figure;
%plot(x,y)
%xlim([-0.1,1.1])
%ylim([-0.5,0.5])
%grid on
%pbaspect([1 .75 1])
```

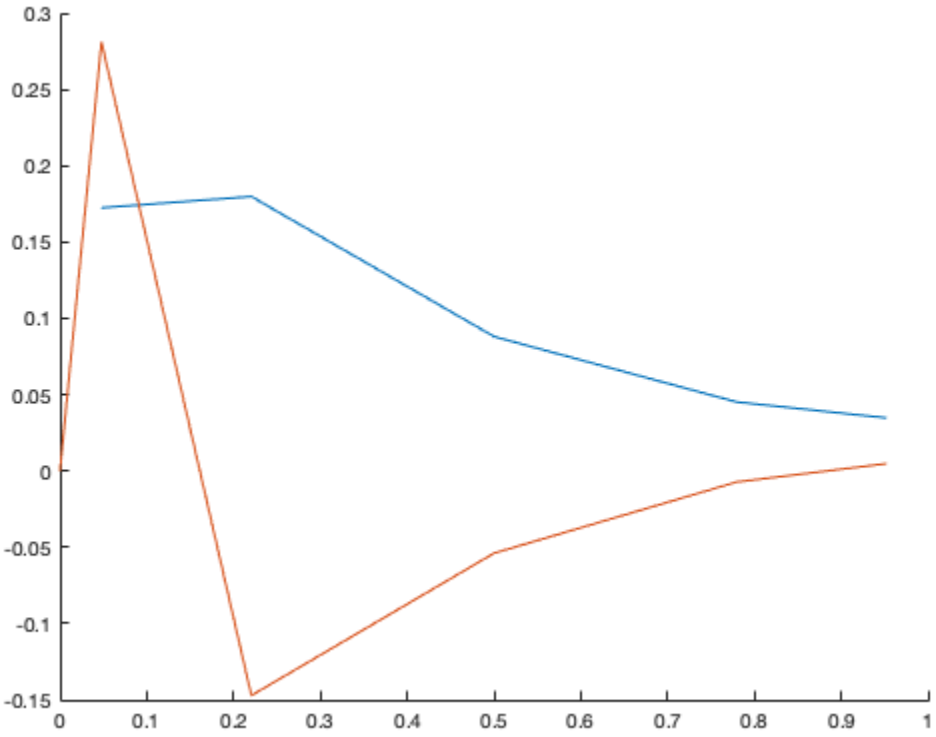
Discretize airfoil into panels

```
N = 10;
panels = definePanel(x_airfoil,y_airfoil,N);

for i = 1:N
  x_ends(i) = panels(i).xa;
  y_ends(i) = panels(i).ya;
end


figure;
hold on
plot(x_airfoil,y_airfoil)
plot(x_ends,y_ends,'marker','o')
xlim([-0.1,1.1])
ylim([-0.5,0.5])
pbaspect([1 .75 1])
hold off

% Create freestream object
u_inf = 1.0;
alpha = 0.0;

freestream = Freestream(u_inf, alpha);

% Create the linear system
A1 = buildMatrix(panels);
b = buildRHS(panels, freestream);
b1 = b(:,1);

% Solve system of linear equations
```

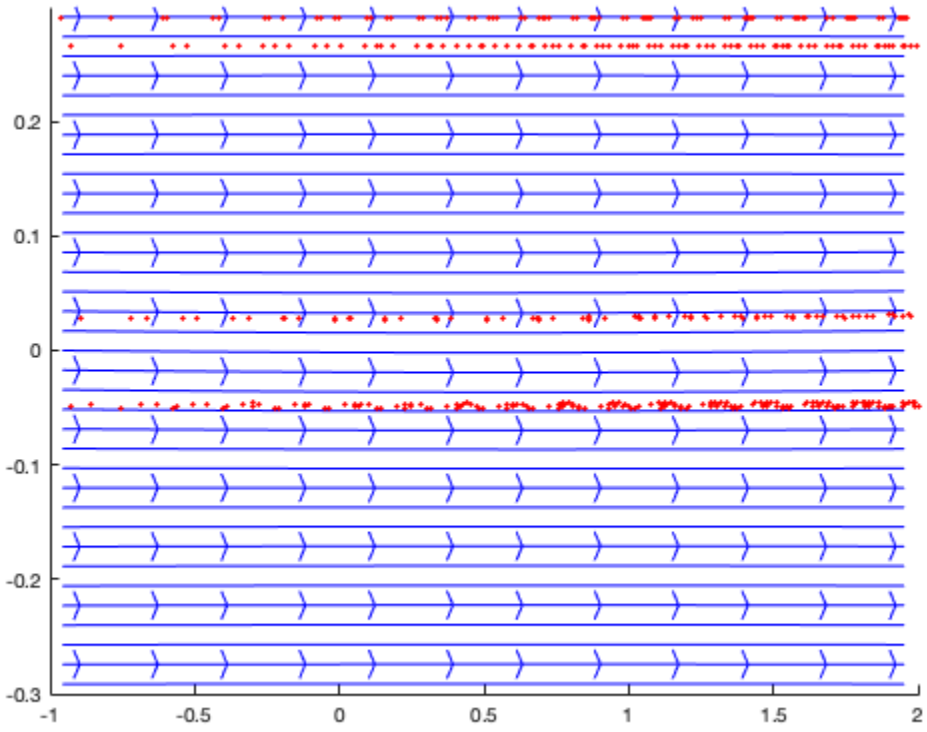
```
sigma =  inv(A1)*b1;  %  linsolve(A1,b1);

for i=1:(N)
 % panels(i).vt = tangential_velocity(i);
  %panels(i).cp = 1.0 - (panels(i).vt / freestream.u_inf).^2;
  panels(i).sigma = sigma(i);  % add sigma value to objects
end

% Calculate Tangential Velocity
panels = tangentialVelocity(panels,freestream,sigma);

% Calculate Pressure Coefficient
panels = pressureCoefficient(panels,freestream);

figure;
hold on;
for i=1:N
  if (panels(i).loc == 'upper')
    xc_u(i) = panels(i).x_center;
    cp_u(i) = panels(i).cp;
  elseif (panels(i).loc == 'lower')
    xc_l(i) = panels(i).x_center;
    cp_l(i) = panels(i).cp;
  end
end
plot(xc_u,cp_u);
plot(xc_l,cp_l);
hold off;


% Plot streamlines

% make meshgrid


x_start= -1;
x_end = 2;
y_start = -0.3;
y_end = .3;

x = linspace(x_start, x_end,N);
y = linspace(y_start, y_end,N);
[X,Y] = meshgrid(x,y);
Z = zeros(size(X));
w = ones(size(Y));

% calculate velocity field on the meshgrid

[u,v] = velocityField(panels, freestream, X, Y);

y_ends =
     0     0     0     0     0     0     0     0     0     0     0
y_ends =
  Columns 1 through 6
```

```
            0     0.013071            0            0            0
       0
  Columns 7 through 11
            0            0            0            0            0
y_ends =
  Columns 1 through 6
            0     0.013071     0.040686            0            0
       0
  Columns 7 through 11
            0            0            0            0            0
y_ends =
  Columns 1 through 6
            0     0.013071     0.040686     0.059557            0
       0
  Columns 7 through 11
            0            0            0            0            0
y_ends =
  Columns 1 through 6
            0     0.013071     0.040686     0.059557     0.046049
       0
  Columns 7 through 11
            0            0            0            0            0
y_ends =
  Columns 1 through 6
            0     0.013071     0.040686     0.059557     0.046049
       0
  Columns 7 through 11
            0            0            0            0            0
y_ends =
  Columns 1 through 6
            0     0.013071     0.040686     0.059557     0.046049
       0
  Columns 7 through 11
    -0.046049            0            0            0            0
y_ends =
  Columns 1 through 6
            0     0.013071     0.040686     0.059557     0.046049
       0
  Columns 7 through 11
    -0.046049    -0.059557            0            0            0
y_ends =
  Columns 1 through 6
            0     0.013071     0.040686     0.059557     0.046049
       0
  Columns 7 through 11
    -0.046049    -0.059557    -0.040686            0            0
y_ends =
  Columns 1 through 6
            0     0.013071     0.040686     0.059557     0.046049
       0
  Columns 7 through 11
    -0.046049    -0.059557    -0.040686    -0.013071            0
```


0.5
0.4
0.3
0.2
0.1
0
-0.1
-0.2
-0.3
-0.4
-0.5
0
0.2
0.4
0.6
0.8
1


0.3
0.25
0.2
0.15
0.1
0.05
0
-0.05
-0.1
-0.15
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1

```
figure;
xstart = min(X)+(max(X)-min(X)).*rand(round(N/2),1);
ystart = min(Y)+(max(Y)-min(Y)).*rand(round(N/2),1);
zstart = min(Z)+(max(Z)-min(Z)).*rand(round(N/2),1);
hold on
xlim([x_start x_end]);
ylim([y_start y_end]);
h=streamslice(X,Y,u,v,3);
verts = stream2(X,Y,u,v,xstart,ystart);
set(h,'color','blue')
iverts = interpstreamspeed(X,Y,Z,u,v,w,verts,.025);
streamparticles(iverts,500,'animate',10,'ParticleAlignment','off','MarkerSize',3);
hold off

% figure;
% streamline(X,Y,u,v,xstart,ystart);
```



Published with MATLAB® R2018a