
Table of Contents

.....	1
Plot 1: CFD Grid	1
Plot 2: Airfoil Geometry from 3 sources	2
Plot 3: Mach on different parts of airfoil	3
Plot 4: Pressure on different parts of airfoil	5
Plot 5: Pressure, Density, and Velocity	6
Plot 6: Density over 799 iterations	8
Plot 7: Convergence of Continuity and Energy	8

```
%Liam Hood
%Aero 300
%Lab 2

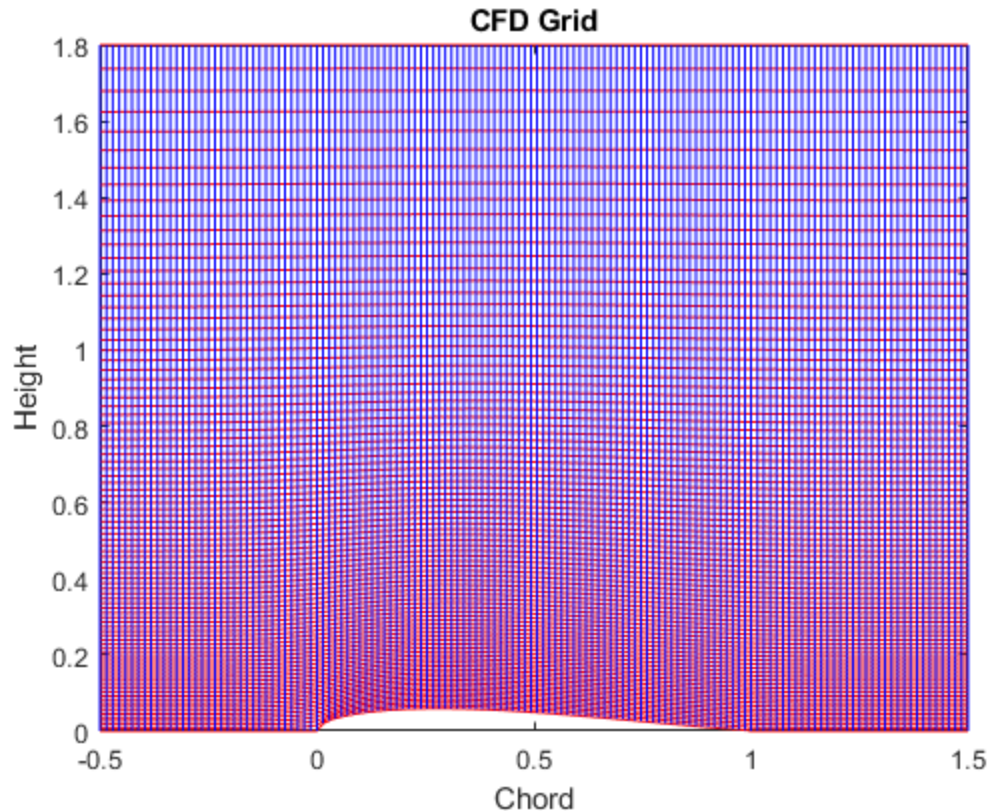
clear;
close all;
tic
% Load all the data
x = textread('x.txt');
y = textread('y.txt');
vx = textread('vx.txt');
vy = textread('vy.txt');
rho = textread('rho.txt');
pressure = textread('pressure.txt');
mach = textread('mach.txt');
load('DENSITY_iteration.mat')
con = readtable('convergence.dat');
```

Plot 1: CFD Grid

```
plot( x , y , '-r' , x.' , y.' , '-b') %Plot of CFD grid
xlabel('Chord') %x axis label
ylabel('Height') %y axis label
title('CFD Grid') %title graph

%Answers to the questions
disp('The grid boxes become smaller closer to the airfoil.')
disp('The domain has to be larger than the airfoil so that values can
    be evaluated before the airfoil takes effect')
disp('The larger the domain the more that needs to be evaluated')
```

The grid boxes become smaller closer to the airfoil.
The domain has to be larger than the airfoil so that values can be
evaluated before the airfoil takes effect
The larger the domain the more that needs to be evaluated



Plot 2: Airfoil Geometry from 3 sources

```
figure %making a new plot
hold on %allow for multiple datasources on a single plot
axis equal %making axis equal

%Airfoil shape from given x and y data
x_1 = x( 1:end , 1 ); %x coordinates of airfoil
y_1 = y( 1:end , 1 ); %y coordinates of airfoil
plot( x_1 , y_1 ) %plots the data

%Airfoil shape from NACA
ch = linspace( 0 , 1 , 100 ) ; %Creates 100 data points on the chord
length of 1 to evaluate the following at
m = 0 ; %max camber
p = 0 ; %max camber position
t = .12 ; %Thickness
%evaluates thickness at the various points along the chord
y_ti = ( ( .2969 * ch.^5 ) + ( -.1260 * ch ) + ( -.3516 * ch.^2 ) +
( .2843 * ch.^3 ) + ( -.1015 * ch.^4 ) ) ;
y_t = ( t / .2 ) .* y_ti ;
%plots thickness against chord length
plot( ch , y_t , '-r' )

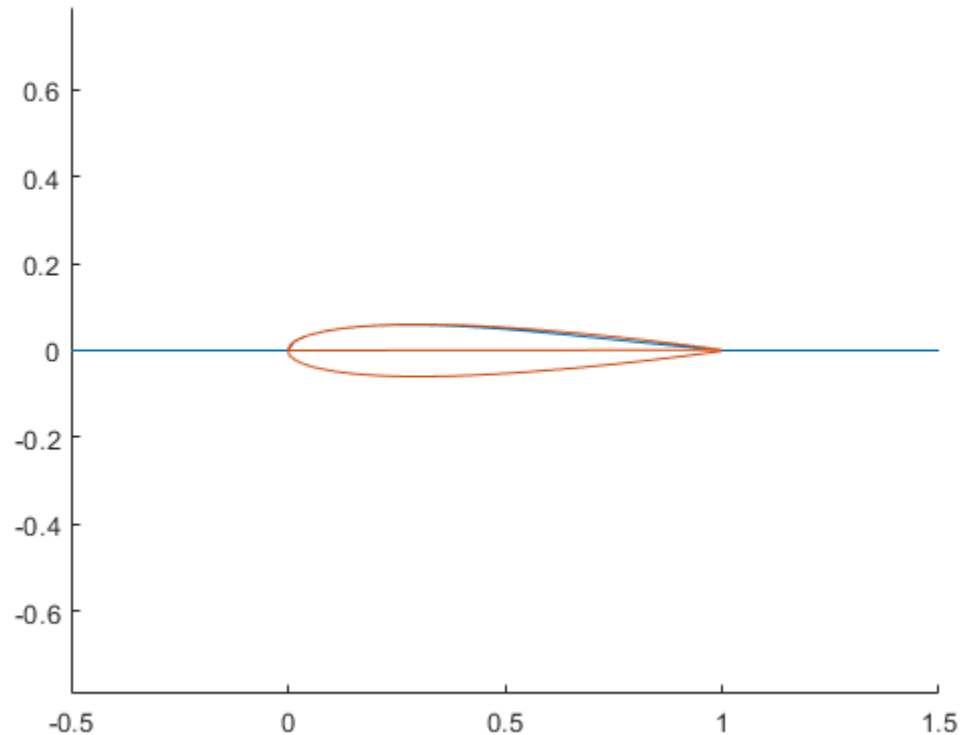
%Airfoil shape from website
```

```

geo = textread('NACA_0012.txt'); %import data
x_3 = geo( 1:end , 1 ); %x coordinates of airfoil edge
y_3 = geo( 1:end , 2 ); %y coordinates of airfoil edge
plot( x_3 , y_3 ) %plot of airfoil shape

hold off

```



Plot 3: Mach on different parts of airfoil

```

figure %Creates new figure
hold on

subplot( 2 , 2 , 1 ) %graph in upper left
contour( x , y , mach , 100 )
%defines axis
caxis([0 2])
%labels graph
xlabel('Chord')
ylabel('Height')
title('Full Domain')

subplot( 2 , 2 , 2 ) %graph in upper right
contour( x , y , mach , 100 )
%defines axis
axis([ -0.1 , 1.1 , 0 , 1.2 ])
caxis([0 2])

```

```

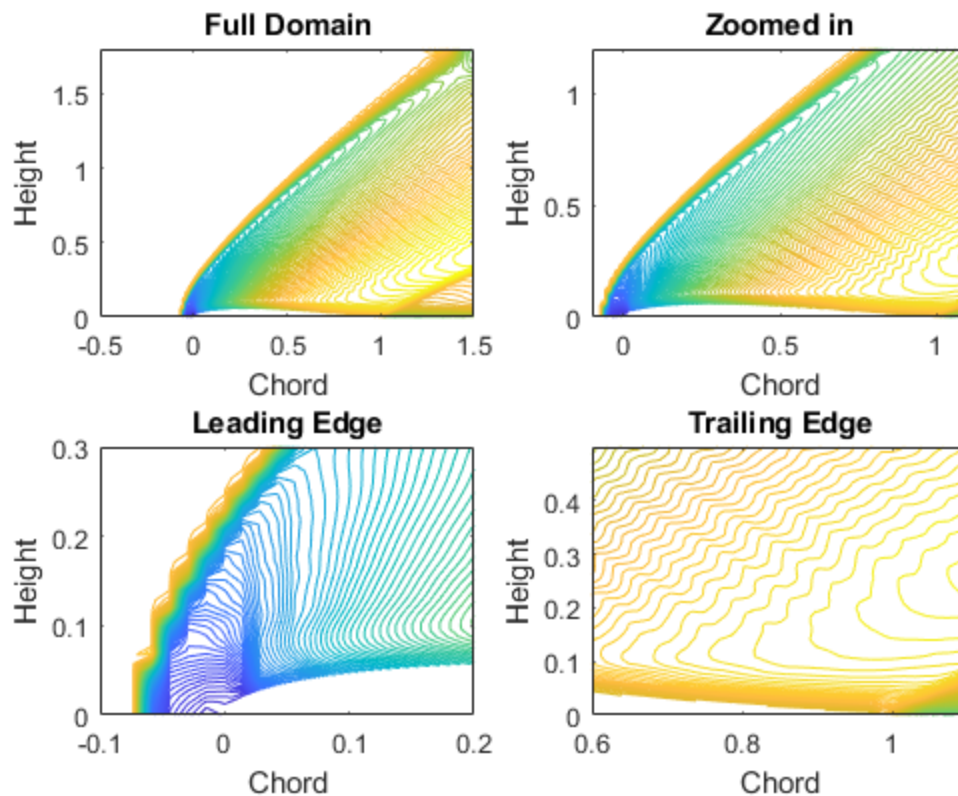
%labels graph
xlabel('Chord')
ylabel('Height')
title('Zoomed in')

subplot( 2 , 2 , 3 ) %graph in lower left
contour( x , y , mach , 100 )
%defines axis
axis([ -.1 , .2 , 0 , .3 ])
caxis([0 2])
%labels graph
xlabel('Chord')
ylabel('Height')
title('Leading Edge')

subplot( 2 , 2 , 4 ) %graph in lower right
contour( x , y , mach , 100 )
%defines axis
axis([ .6 , 1.1 , 0 , .5 ])
caxis([0 2])
%labels graph
xlabel('Chord')
ylabel('Height')
title('Trailing Edge')

hold off

```



Plot 4: Pressure on different parts of airfoil

```
figure
hold on

subplot( 2 , 2 , 1 ) %graph in upper left
s = surf( x , y , pressure ) ;
%defines axis
caxis([0 2])
%labels graph
xlabel('Chord')
ylabel('Height')
title('Full Domain')

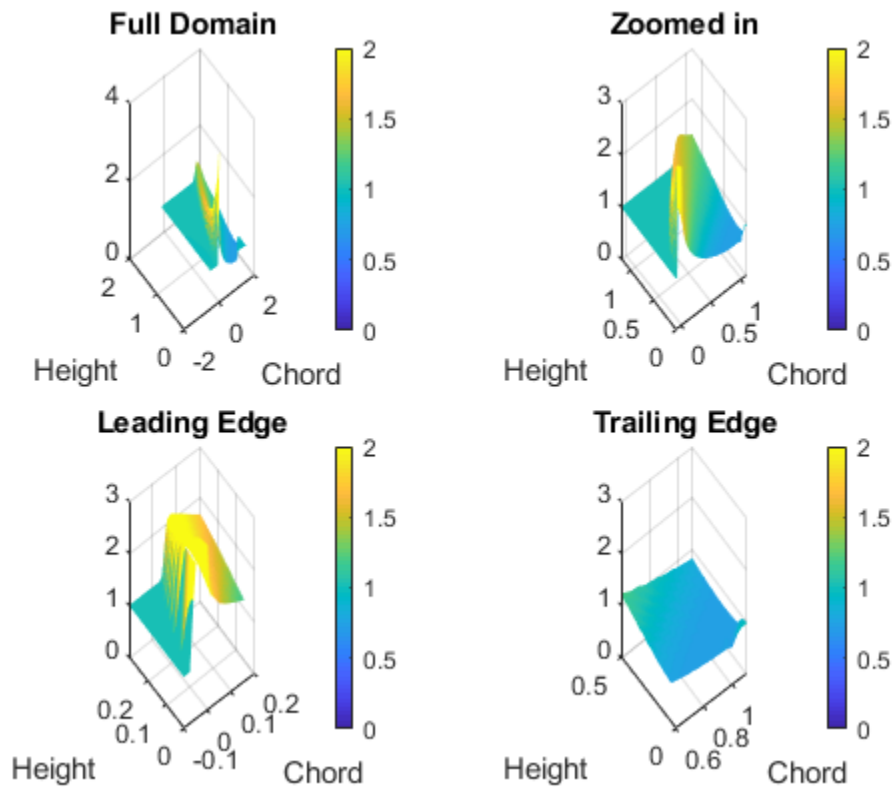
colorbar
s.EdgeColor = 'none' ;

subplot( 2 , 2 , 2 ) %graph in upper right
d = surf( x , y , pressure ) ;
%defines axis
axis([ -.1 , 1.1 , 0 , 1.2 , 0 , 3 ])
caxis([0 2])
%labels graph
xlabel('Chord')
ylabel('Height')
title('Zoomed in')
colorbar
d.EdgeColor = 'none' ;

subplot( 2 , 2 , 3 ) %graph in lower left
f = surf( x , y , pressure ) ;
%defines axis
axis([ -.1 , .2 , 0 , .3 , 0 , 3 ])
caxis([0 2])
%labels graph
xlabel('Chord')
ylabel('Height')
title('Leading Edge')
colorbar
f.EdgeColor = 'none' ;

subplot( 2 , 2 , 4 ) %graph in lower right
g = surf( x , y , pressure ) ;
%defines axis
axis([ .6 , 1.1 , 0 , .5 , 0 , 3 ])
caxis([0 2])
%labels graph
xlabel('Chord')
ylabel('Height')
title('Trailing Edge')
colorbar
g.EdgeColor = 'none' ;
```

```
hold off
```



Plot 5: Pressure, Density, and Velocity

```
figure
hold on

subplot( 2 , 2 , 1 ) %graph in upper left
ss = surf( x , y , pressure ) ;
%defines axis
caxis([0 2])
%labels graph
xlabel('Chord')
ylabel('Height')
title('Pressure')
colorbar
ss.EdgeColor = 'none' ;

subplot( 2 , 2 , 2 ) %graph in upper right
dd = surf( x , y , rho ) ;
%defines axis
caxis([0 max( max( rho ))])
%labels graph
xlabel('Chord')
ylabel('Height')
```

```
title('Density')
colorbar
dd.EdgeColor = 'none' ;

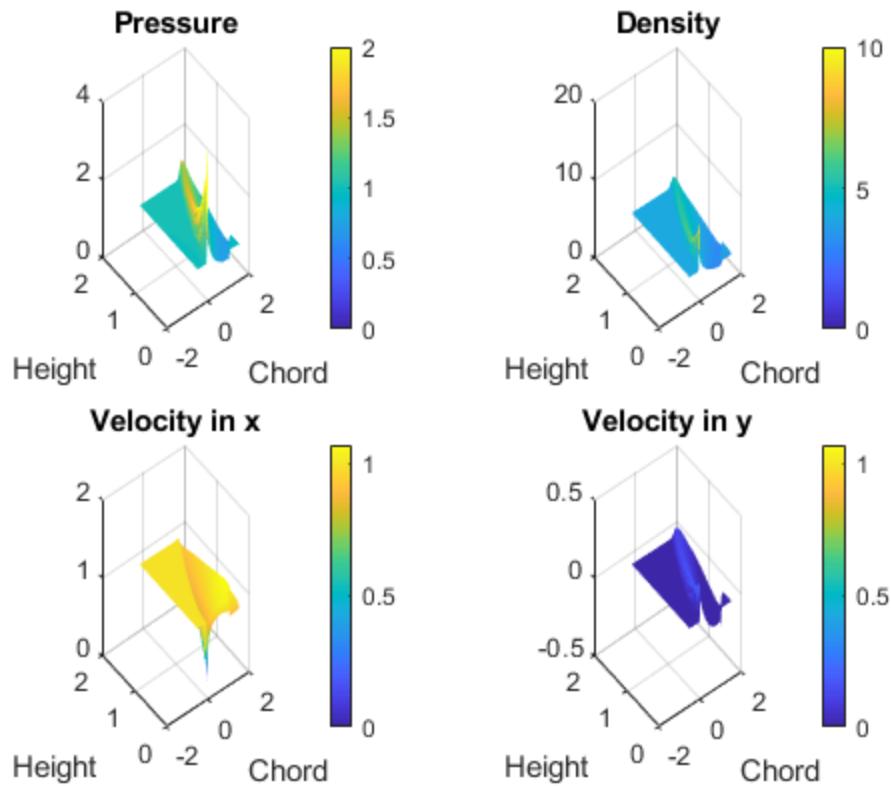
subplot( 2 , 2 , 3 ) %graph in lower left
ff = surf( x , y , vx ) ;
%defines axis
caxis([0 max( max( vx )) ])
%labels graph
xlabel('Chord')
ylabel('Height')
title('Velocity in x')
colorbar
ff.EdgeColor = 'none' ;

subplot( 2 , 2 , 4 ) %graph in lower right
gg = surf( x , y , vy ) ;
%defines axis
caxis([0 max( max( vx ))]) %keeps color axis the same as velocity in
the x direction
%labels graph
xlabel('Chord')
ylabel('Height')
title('Velocity in y')
colorbar
gg.EdgeColor = 'none' ;

hold off
t_2 = toc

t_2 =

5.1882
```



Plot 6: Density over 799 iterations

```
figure prompt = 'Do you want to start the animation? Y/N [Y]: '; %Creates prompt to run str = input(prompt);
```

```
%Creates animation if str < 1 %starts animation if a number less than one is entered tic for ii = 1:799
%Graphs all 799 iterations of density data aa = surf( x , y , C{ 1 , ii } ); %creates graph axis([ -0.5 , 1.5 , 0 ,
2 , 2 , 10 ]) %creates axis caxis([0 max( max( rho ))]) %keeps color scale same as previous density data
%Labels the graph xlabel('Chord') ylabel('Height') zlabel('Density') title(['Density after ' , num2str(ii) , '
iterations' ]) colorbar aa.EdgeColor = 'none'; %removes edge color drawnow %graphs immediately on
every iteration end t_a = toc end
```

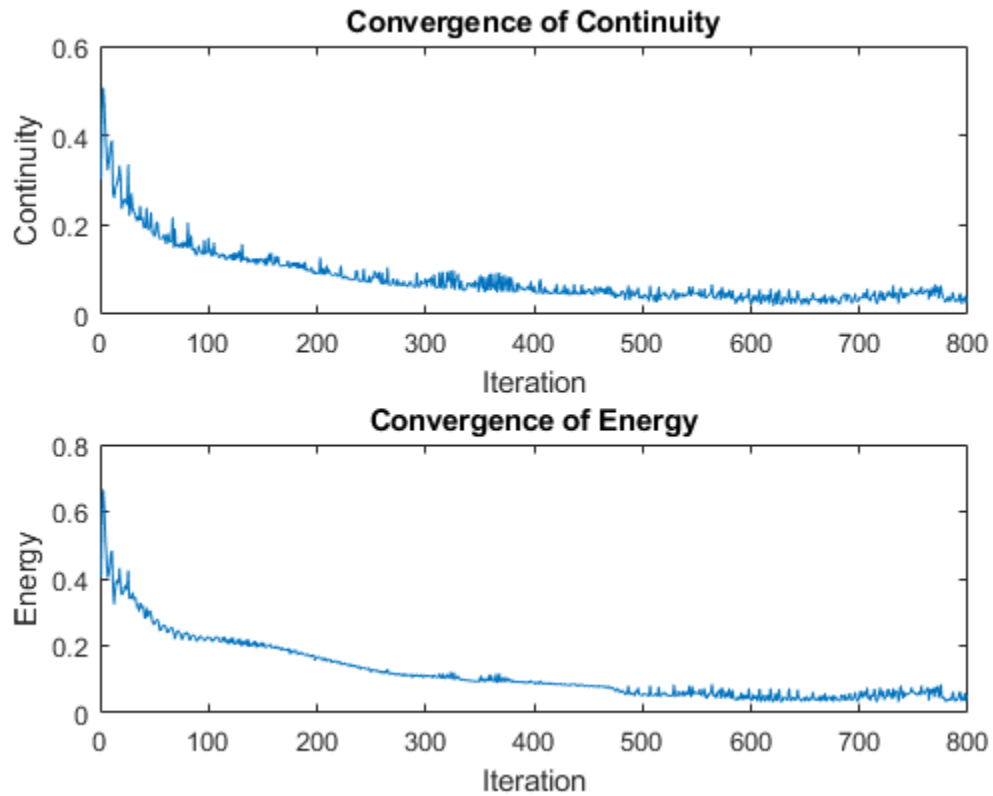
Plot 7: Convergence of Continuity and Energy

```
figure
```

```
%Converts the loaded table data to vectors
iteration = table2array( con( : , 1 ) );
continuity = table2array( con( : , 2 ) );
energy = table2array( con( : , 3 ) );

subplot( 2 , 1 , 1 ) %Graph on top
plot( iteration , continuity )
%labels graph
title( 'Convergence of Continuity' )
```

```
xlabel( 'Iteration' )  
ylabel( 'Continuity' )  
  
subplot( 2 , 1 , 2 ) %Graph below  
plot( iteration , energy )  
%labels graph  
title( 'Convergence of Energy' )  
xlabel( 'Iteration' )  
ylabel( 'Energy' )
```



Published with MATLAB® R2017b