Aerodynamic Project

Flow Simulation over a Joukowski airfoil (With Ansys)

Made by:

Phelopater Ramsis Fahmy - 202001171

Under supervision:

Dr. Mohamed Madbouli

Eng. Asmaa AlaaElDeen

Fall 2024

Overview:

The goal of this study is to perform a thorough investigation of the Joukowski airfoil. In this study, we investigate the lift and moment performance, aerodynamic properties, and the behavior of the ow around the airfoil. Ansys Fluent is then used to confirm these findings. In order to better comprehend the Joukowski airfoil's performance and its uses in aviation and related sectors, the research intends to offer insightful information on its aerodynamic behavior.

Deliverables:

1. Drawing the airfoil section

2. Drawing the streamlines around the airfoil for an angle of attack of 7°

3. Drawing the velocity over the airfoil surface for an angle of attack of 7°

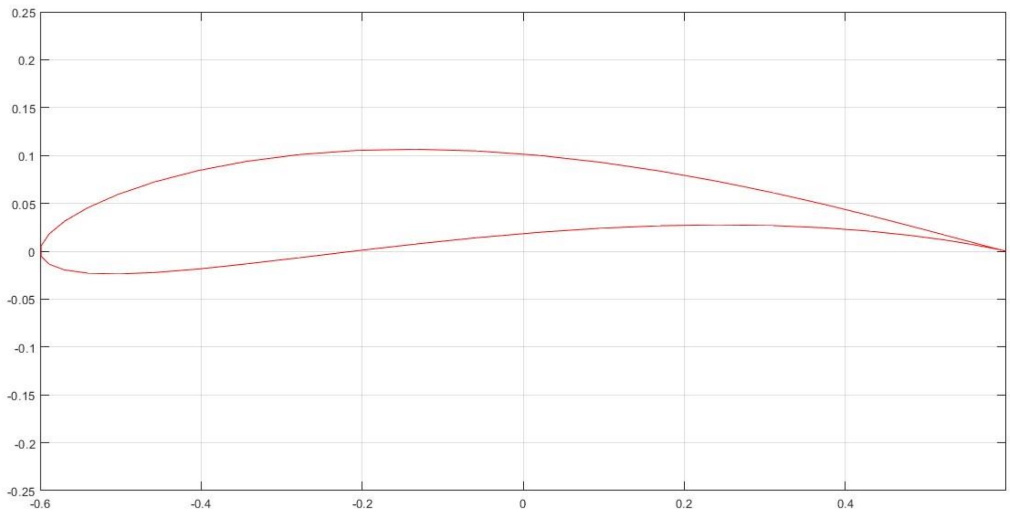
4. Drawing the Cp over the airfoil surface for an angle of attack of 7°

5. The variation of the lift coefficient and moment cofficient with the angle from −5° to 10°

6. Solving using Ansys

1. Drawing the airfoil section:

To draw the airfoil, I used the equations provided for the airfoil coordinated (x1,y1)

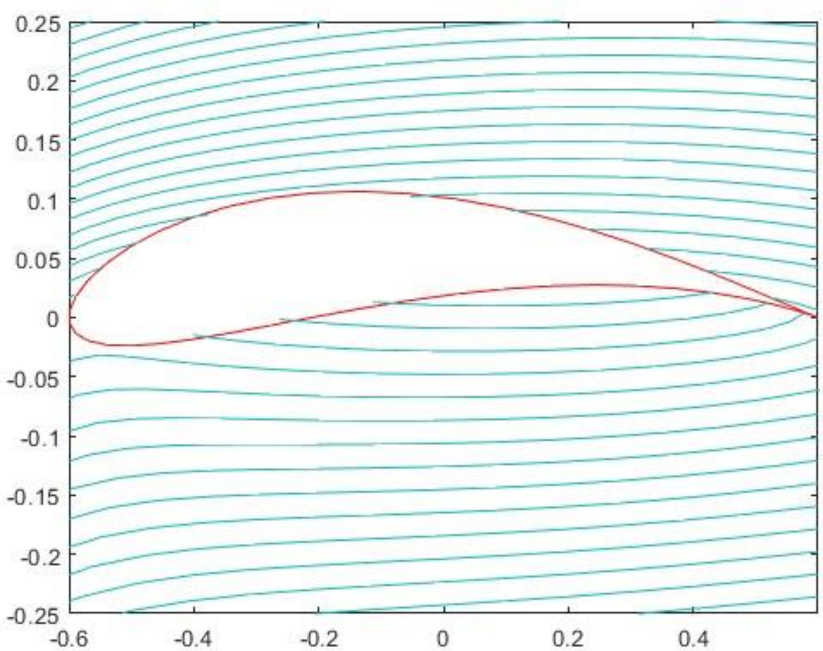


The Matlab code for this part:

```
%% Givens
vfree = 120; % freestream velocity
c = 1.2; % c length
tm_c = 0.09; % max thickness to c
Cm_c = 0.05; % Max chamber to c
alpha = 7 * pi / 180; % angle of attack;

%% Airfoil parameters
b = c / 4;
e = tm_c / 1.3;
beta = 2 * Cm_c;
a = b * (1 + e) / cos(beta);
x0 = -b * e;
y0 = a * beta;

%% Drawing the airfoil
theta1 = linspace(0,2 * pi,50);
x1 = 2 * b * cos(theta1);
y1 = 2 * b * e .* (1 - cos(theta1)) .* sin(theta1) + 2 * b * beta .* sin(theta1).^2;
figure
plot(x1, y1,'r');
grid on;
xlim([-0.6 0.6]);
ylim([-0.25 0.25]);
```

2. Drawing the streamlines around the airfoil

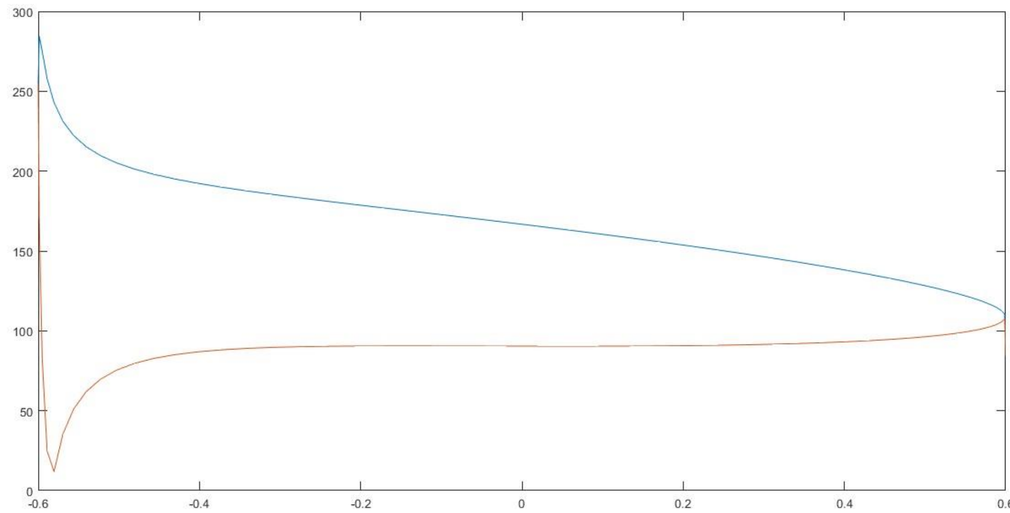
For the streamlines, I rst calculated the w function over a circular cylinder (vortex+doublet+uniform), resulting in an equation in the z plane. We know that Ψ is the imaginary part of w. Finally, to transfer the Ψ to the airfoil plane I followed the Joukowski transformation.



The Matlab code for this part:

```
%% Drawing the streamlines
rd=linspace(a,50,50);
thetad=linspace(0,2*pi,50);
Gamma=4*pi*vfree*a*sin(alpha+beta); %% vortex Strength Caculation
[Thetad,Rd]= meshgrid(thetad,rd);
Z=Rd.*exp(1i*Thetad)+x0+1i*y0; %z=z0+z'
Z1=Z+b^2./Z;%z1=z+b2/z
W = vfree .*exp(-1i.*alpha).*Z+vfree*a^2*exp( 1i *alpha)./Z+1i*log(Z)*Gamma/2/pi;
%douplet+vortex+uniform over cylinder
psi=imag(W); %% Lines of stream line
psi1=psi+b^2./psi; %% cnvorting the stream lines from circular plan z to airfoil plan z1
figure
plot(x1, y1,'r');
hold on
contour(real(Z1),imag(Z1),psi1,'levelstep',2);
xlim([-0.6 0.6]);
ylim([-0.25 0.25]);
```

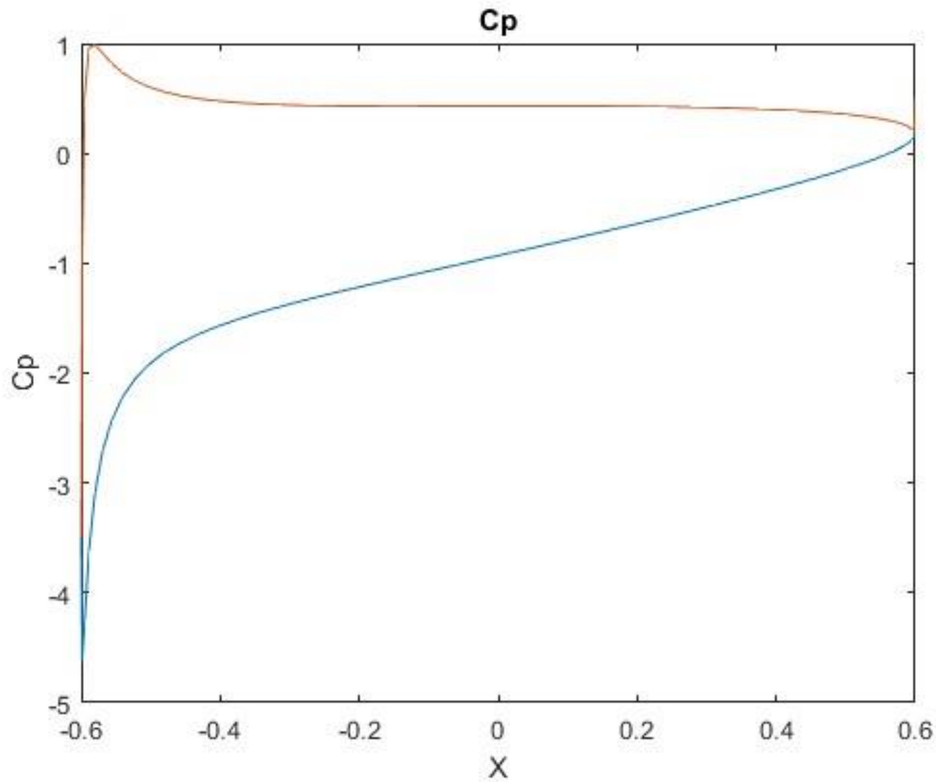
3. Drawing the velocity over the airfoil surface



The Matlab code for this part:

```
%% Drawing the velocity
%Velocity over the upper surface
thetaup =linspace(0.0001,pi,50);
x1up = 2 * b * cos(thetaup);
y1up = 2 * b * e .* (1 - cos(thetaup)) .* sin(thetaup) + 2 * b * beta .* sin(thetaup).^2;
rup=b.*(1+e.*(1-cos(thetaup))+beta.*sin(thetaup));
xup=rup.*cos(thetaup);
yup=rup.*sin(thetaup);
xdup=xup-x0;
ydup=yup-y0;
thetaupdash=atan2(ydup,xdup);
vthetaupd = -vfree .* (2 .*sin(thetaupdash- alpha) + 2 .* sin(alpha + beta));
Vup=(vthetaupd.^2./(1-2*(b^2./rup.^2).*cos(2*thetaup)+(b^4./rup.^4))).^0.5;
figure
title('Velocity vectors');
plot(x1up,Vup);
hold on

%Velocity over the lower surface
thetalow = linspace(pi,2*pi+0.0001,50);
x1low = 2 * b * cos(thetalow);
y1low = 2 * b * e .* (1 - cos(thetalow)) .* sin(thetalow) + 2 * b * beta .* sin(thetalow).^2;
rlow=b.*(1+e.*(1-cos(thetalow))+beta.*sin(thetalow));
xlow=rlow.*cos(thetalow);
ylow=rlow.*sin(thetalow);
xdlow=xlow-x0;
ydlow=ylow-y0;
thetalowdash=atan2(ydlow,xdlow);
vthetadlow = -vfree .* (2 .*sin(thetalowdash- alpha) + 2 .* sin(alpha + beta));
Vlow=(vthetadlow.^2./(1-2*(b^2./rlow.^2).*cos(2*thetalow)+(b^4./rlow.^4))).^0.5;
plot(x1low,Vlow);
```

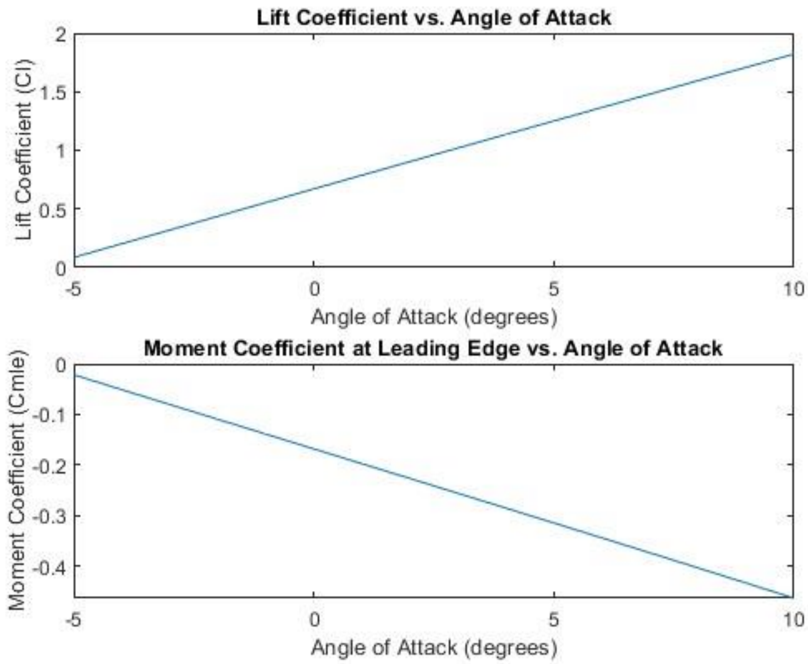
4. Drawing pressure coefcient over the airfoil surface



The Matlab code for this part:

```
%% Plotting pressure coeffecient

Cplow=1-(Vlow./vfree).^2; %Cp on the upper surface
Cpup=1-(Vup./vfree).^2; %Cp on the lower surface
figure

plot(x1up,Cpup);
hold on
plot(x1low,Cplow);

title('Cp');
xlabel('X');
ylabel('Cp');
```

5. The variation of the lift coefcient and moment coefcient with the angle from −5° to 10°

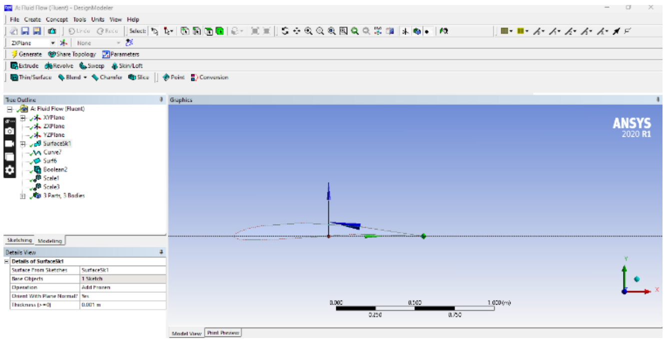


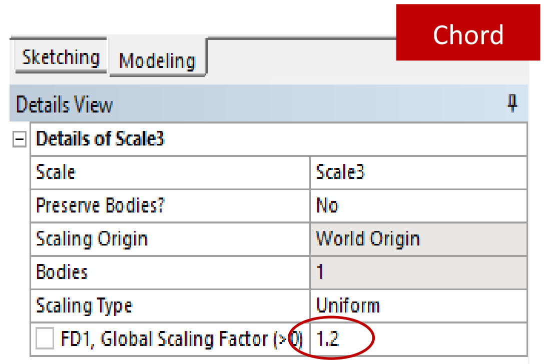
The Matlab code for this part:

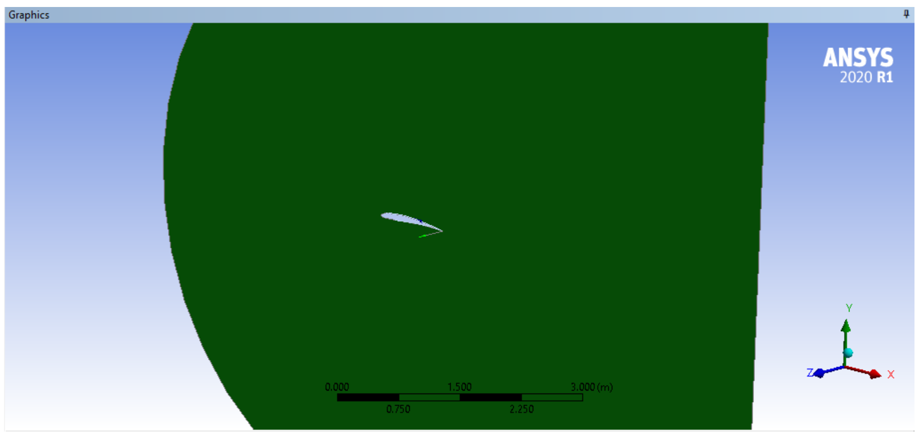
```
%% Constants and Angle of Attack in Degrees
AA_deg = -5:1:10; % Angle of Attack in degrees
beta_rad = beta; % Ensure beta is in radians

%% Lift Coefficient Calculation
Cl = 2 * pi * (1 + e) * sin((AA_deg * pi / 180) + beta_rad);

% Convert angle of attack from degrees to radians for trigonometric calculations
AA_rad = AA_deg * pi / 180;

% Calculate the moment coefficient at the leading edge
Cmle = (-1 .* Cl) ./ (4 * cos(AA_rad));

%% Plotting
figure;
subplot(2,1,1);
plot(AA_deg, Cl);
title('Lift Coefficient vs. Angle of Attack');
xlabel('Angle of Attack (degrees)');
ylabel('Lift Coefficient (Cl)');

subplot(2,1,2);
plot(AA_deg, Cmle);
title('Moment Coefficient at Leading Edge vs. Angle of Attack');
xlabel('Angle of Attack (degrees)');
ylabel('Moment Coefficient (Cmle)');
```

6. Solving using Ansys:

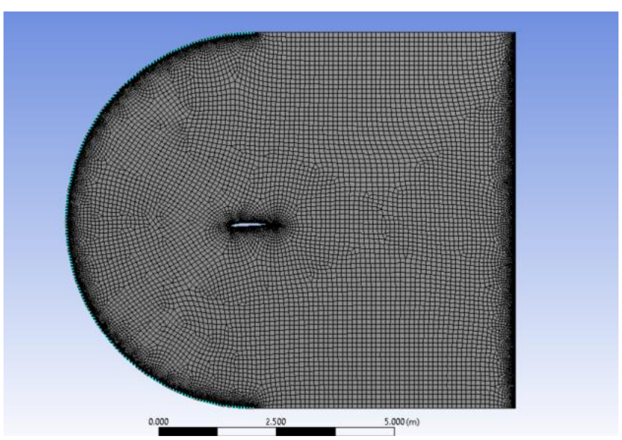
Design phase:

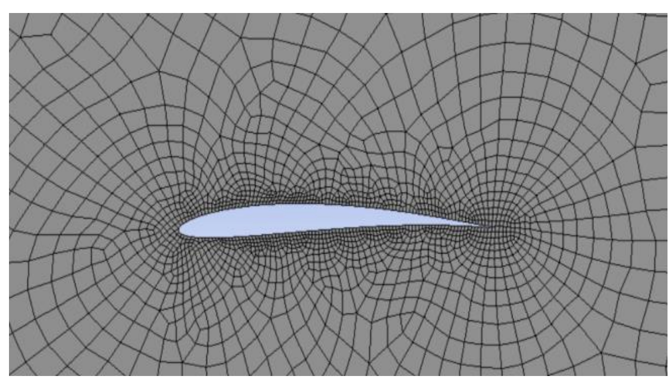




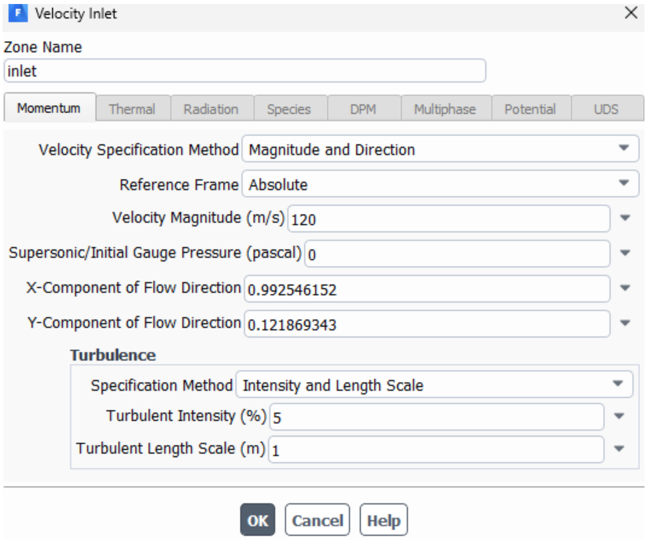


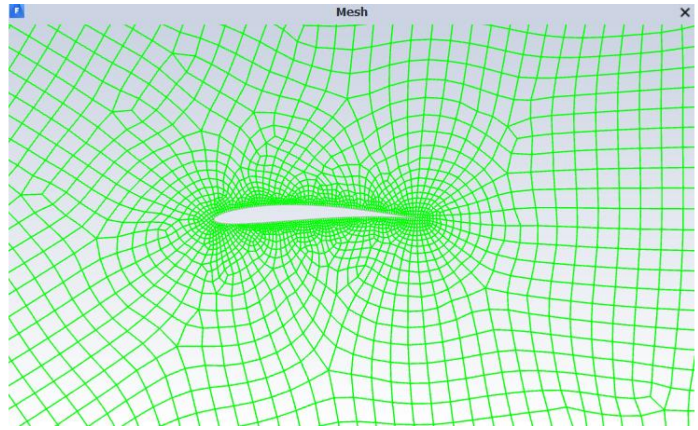
Then we will construct the mesh with smaller elements around the airfoil and at the inlet and outlet of the domain for more accuracy.





After that, we move to the setup of our problem (the velocity “120” and its direction using the given angle of attack (I used an angle of attack of 7°):

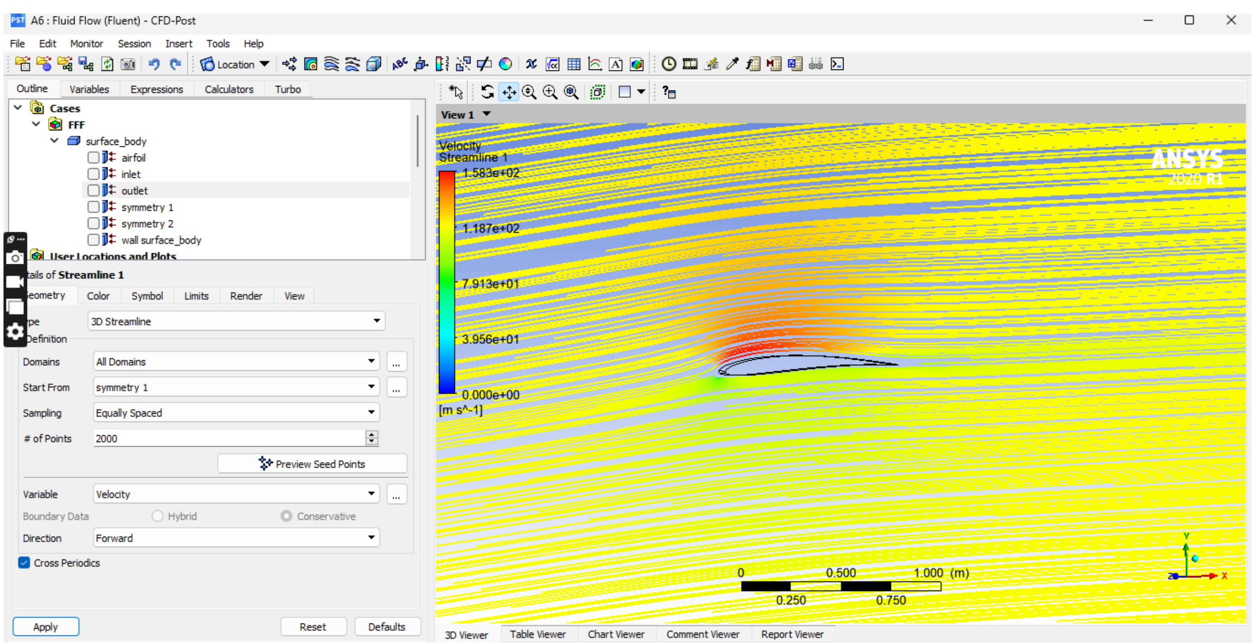


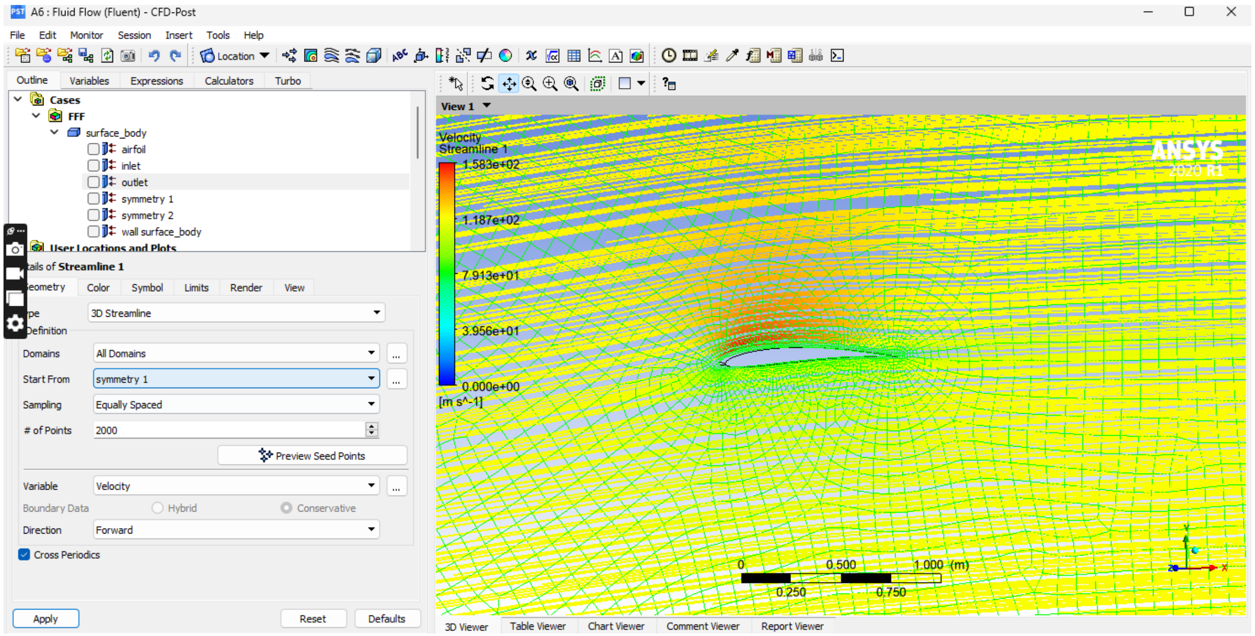


Results of the run:

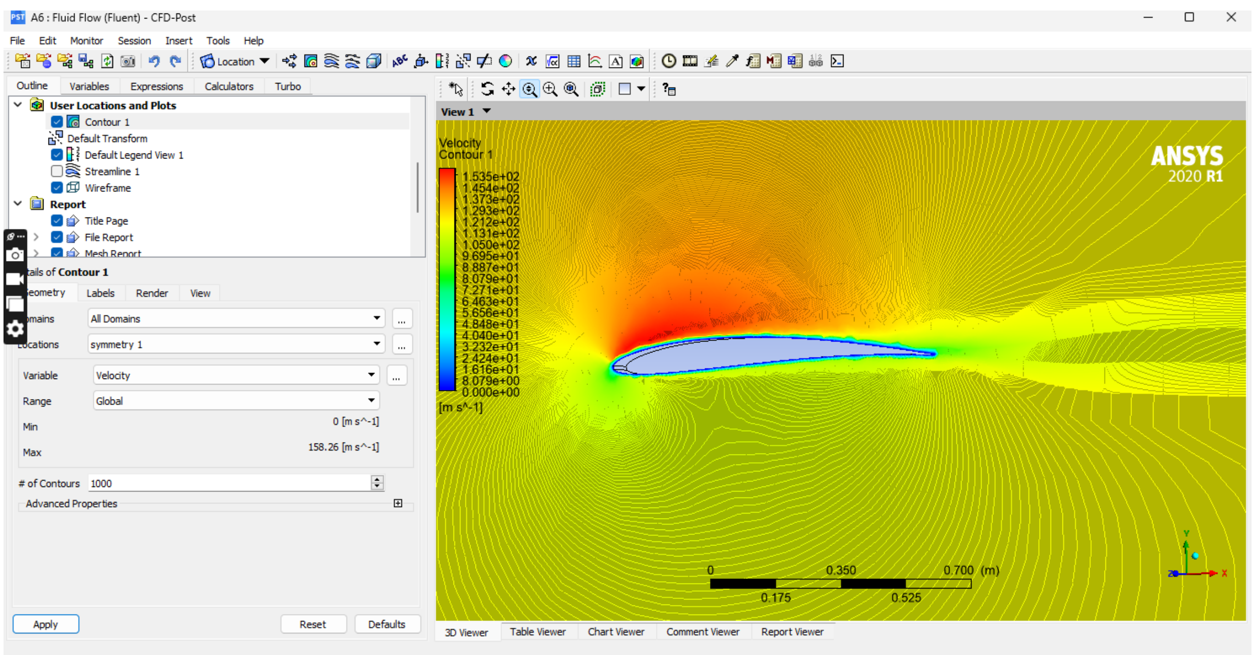
```
iter  continuity  x-velocity  y-velocity           k       omega          cl          cd          cm
1495  1.0464e-13  9.9805e-17  1.3693e-17  1.2601e-16  1.3164e-16  9.1919e-01 -3.7607e-02  9.2103e-02
```

Results of the steamlines:





Results of velocity contour:



Results of pressure contour: