MAE 579 Wind Energy Fall 2021

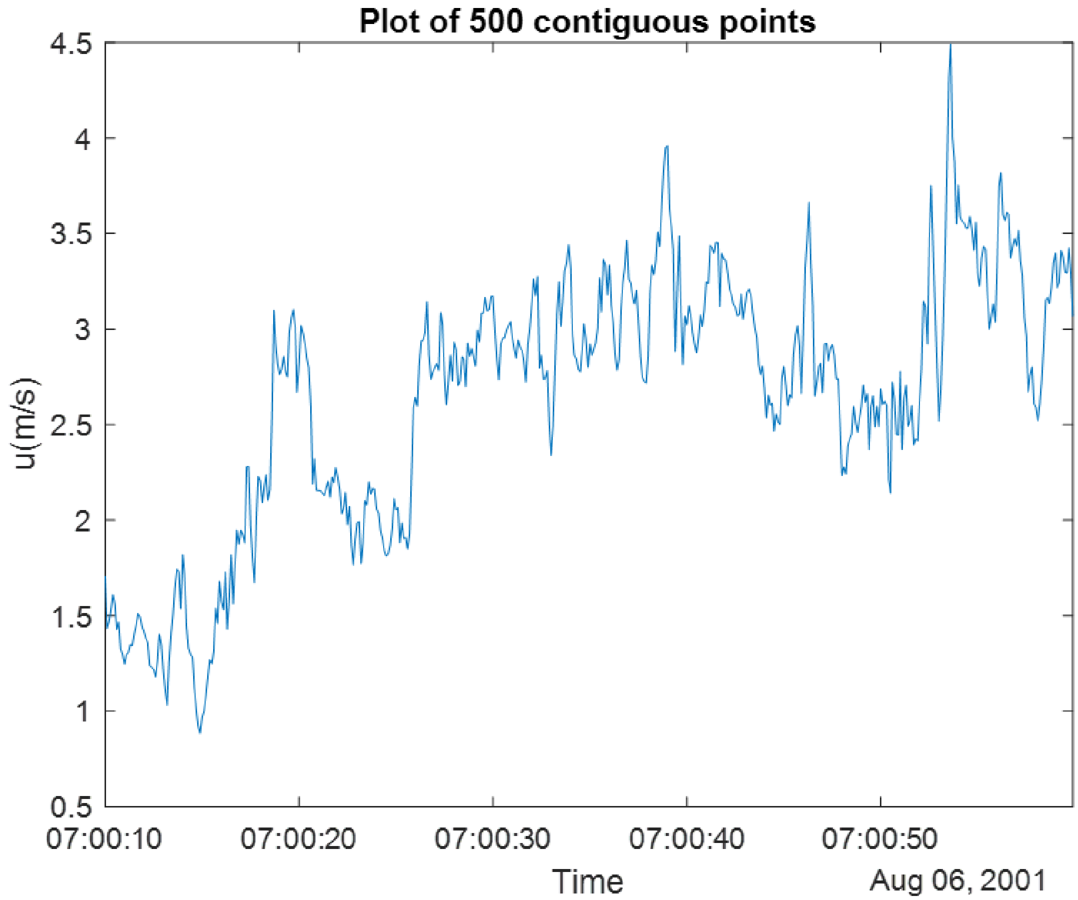
HW4

Wind Analysts: Rajesh Ramesh, Pranav Pawar

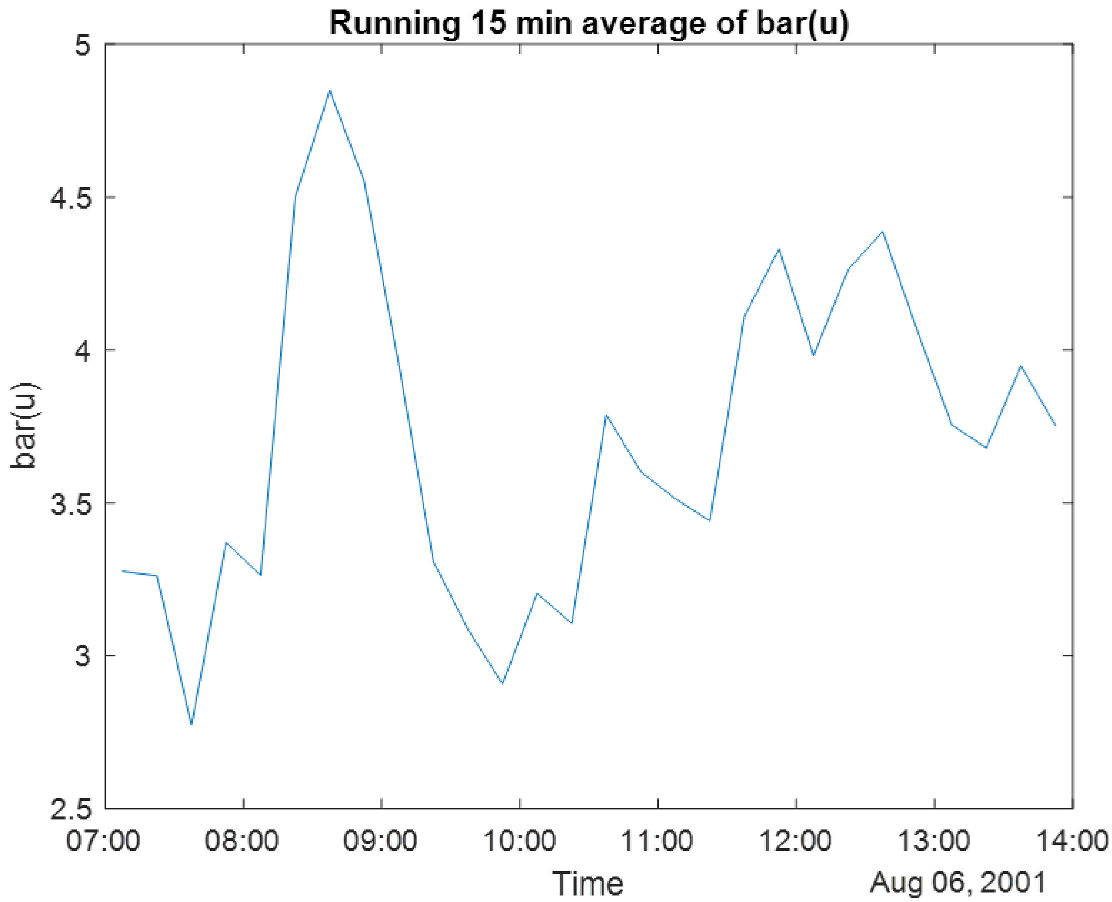
Arizona State University

2.a) Time is in units of day (Last time – first time = 0.291=7/24). Index of time array increments by 0.1s (since 7 hrs = 25200s and there are 252000 elements)

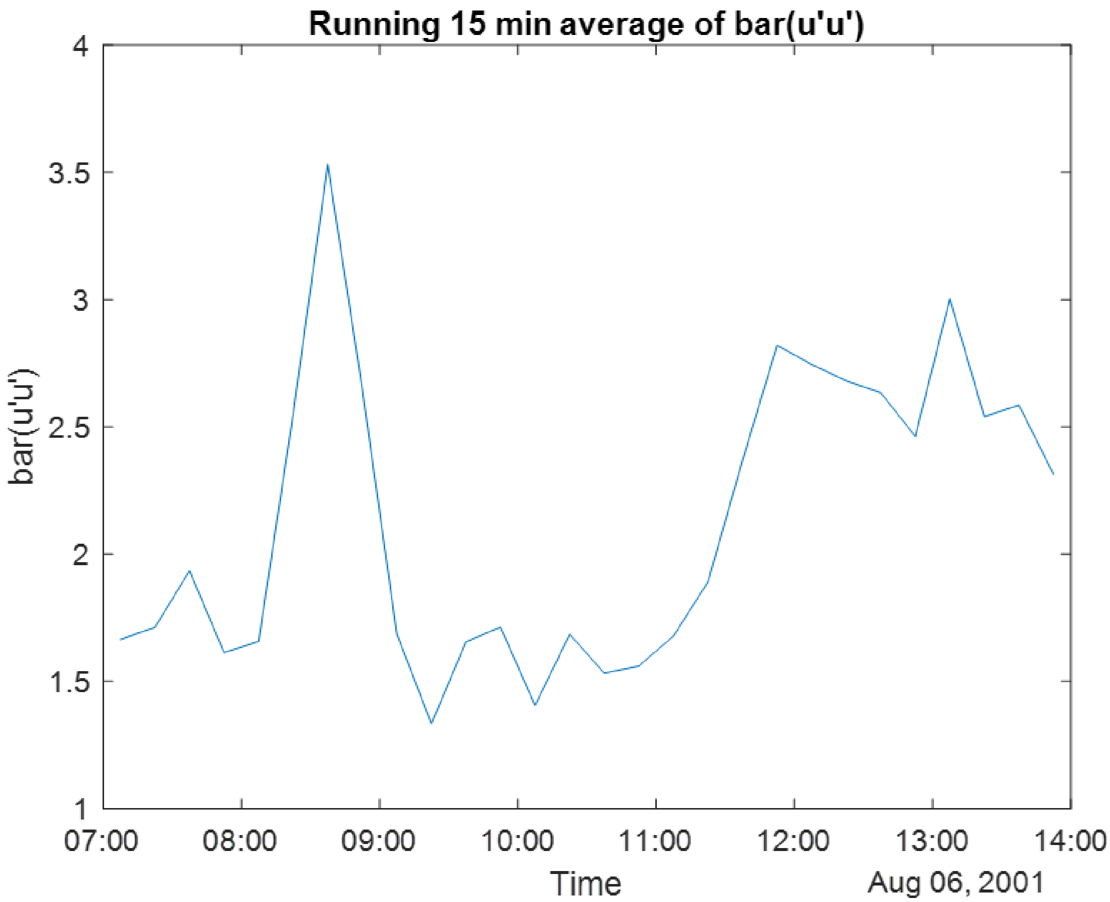
2.b)



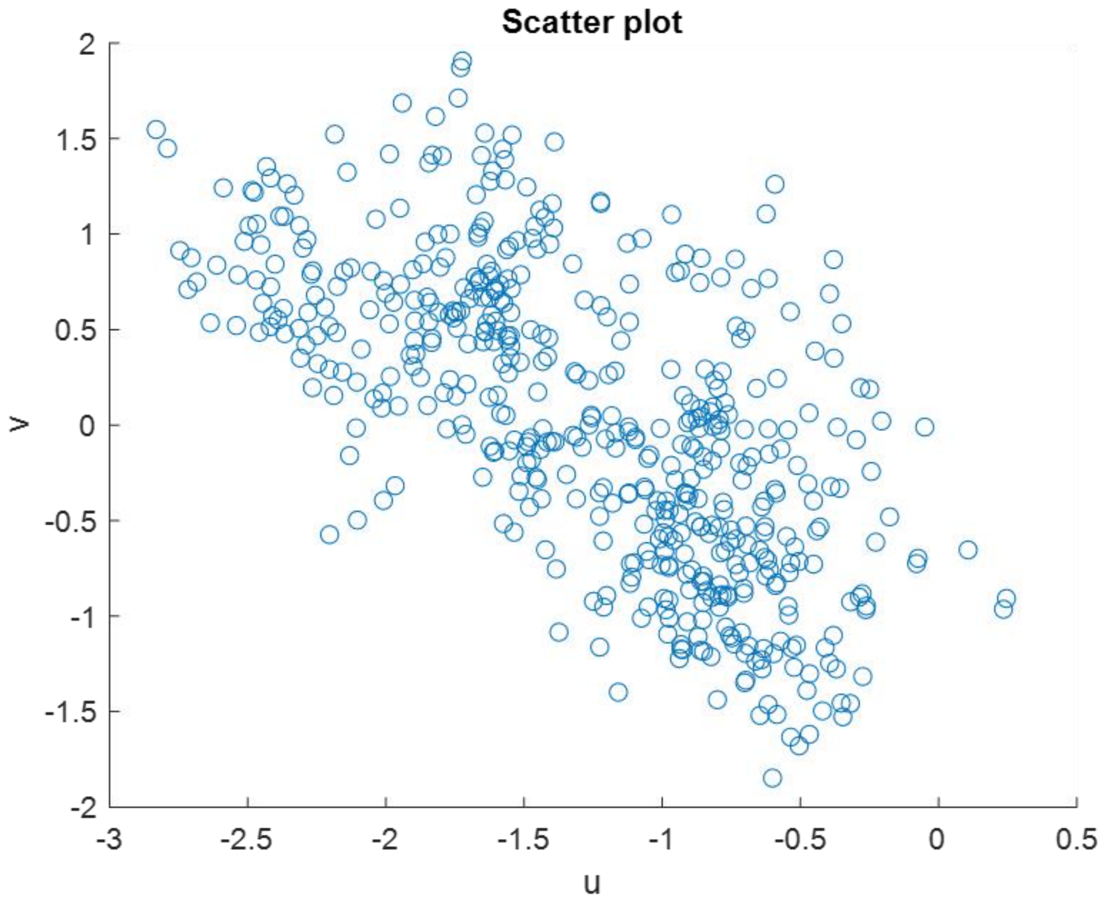
2.c) The average wind speed over 15 min varies with time. Therefore, this is a case of a non-stationary process.



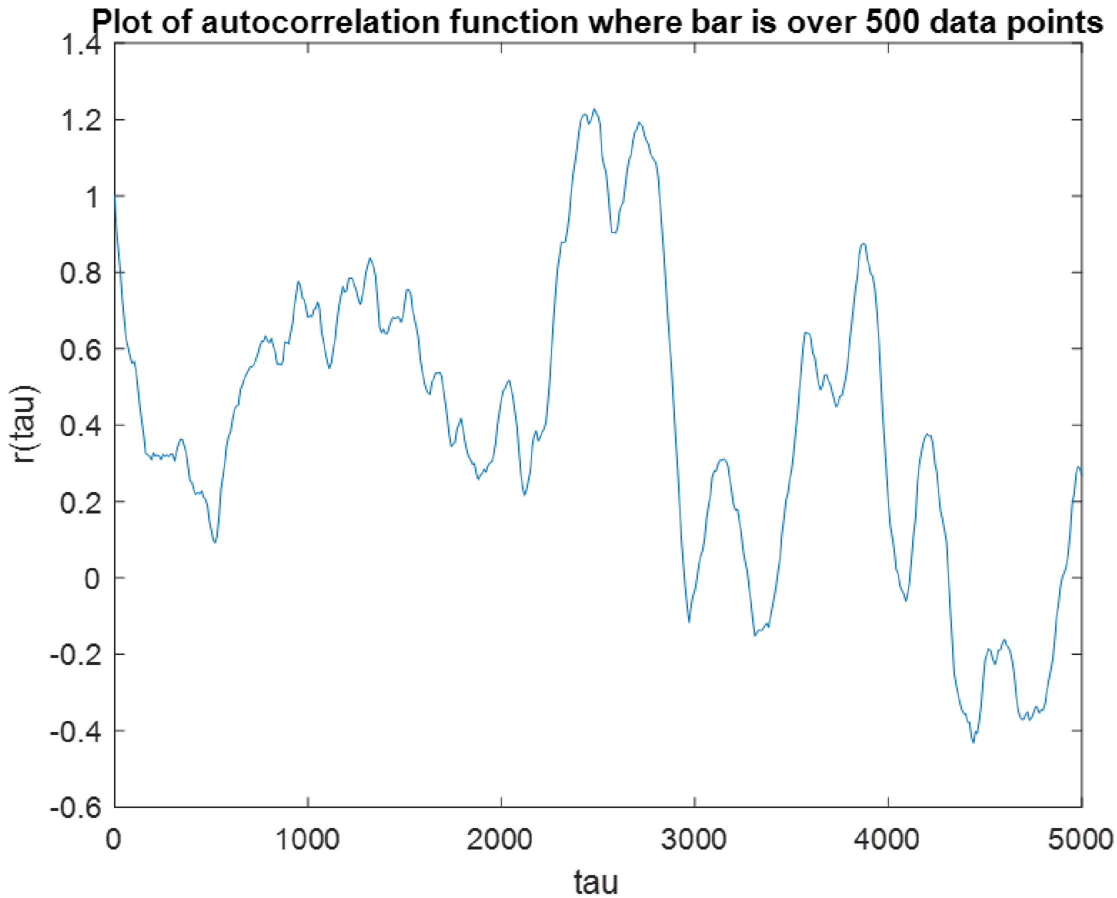
2.d) bar(u'u') = 2.1290 where bar is over all time

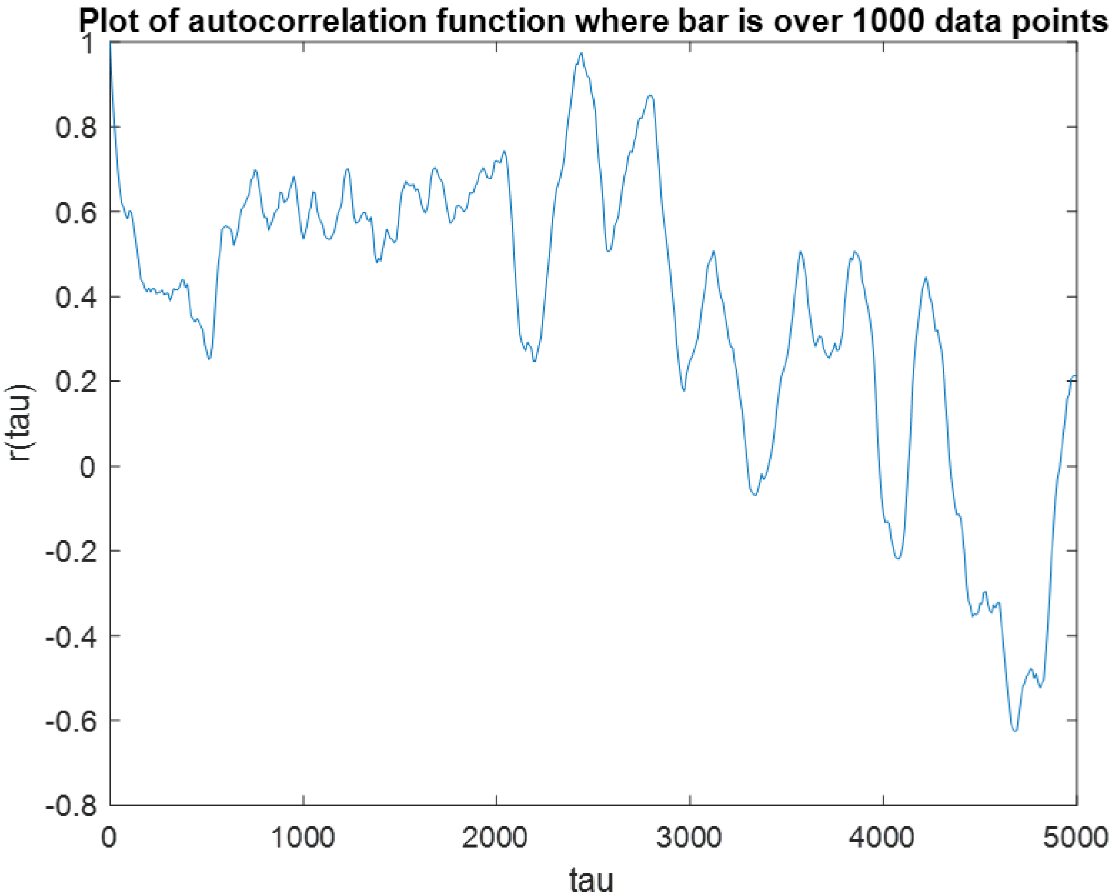


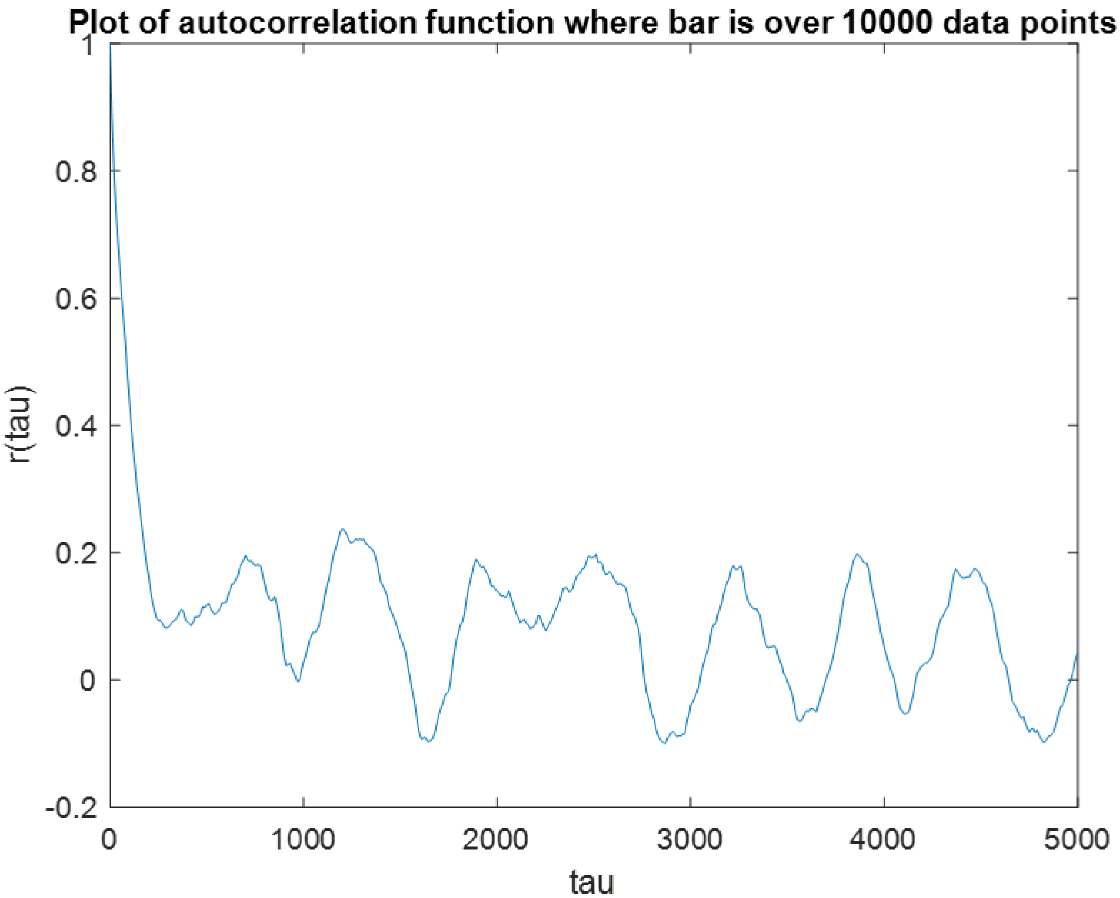
2.e)

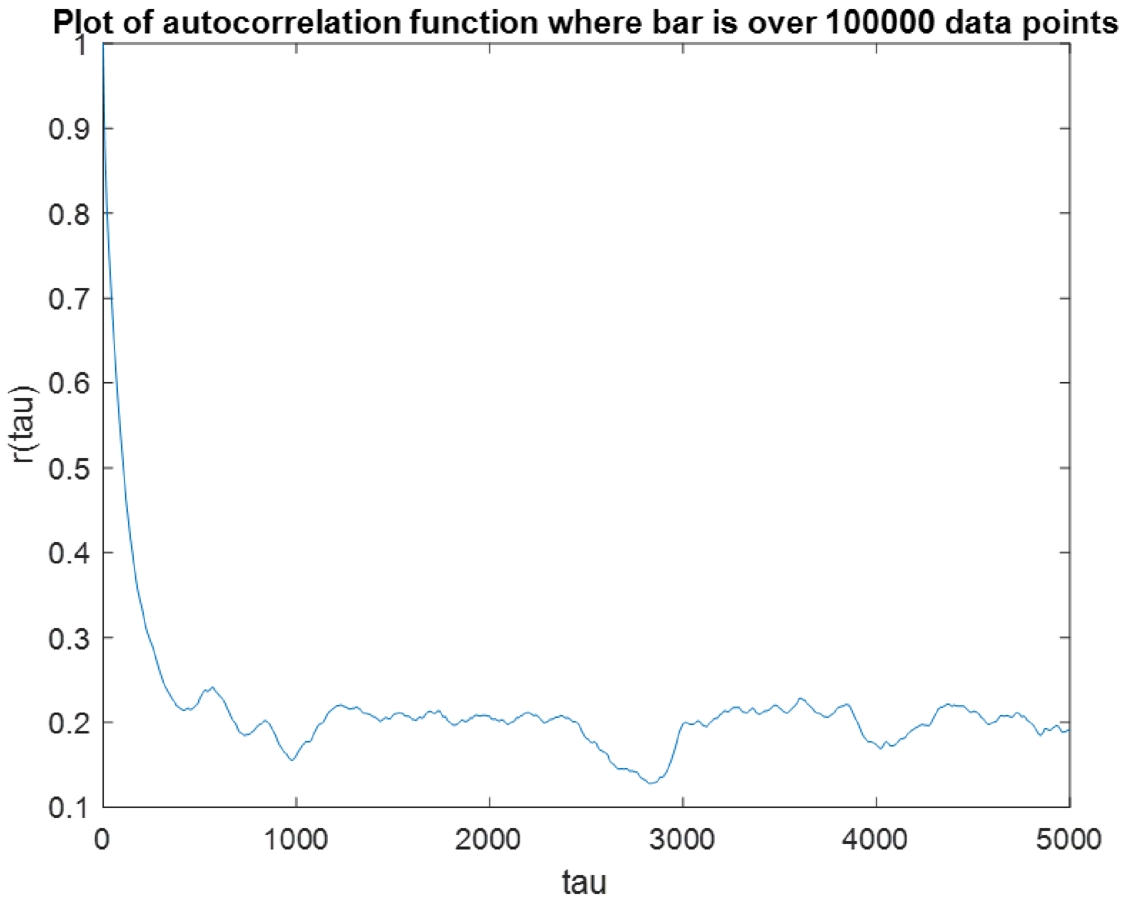


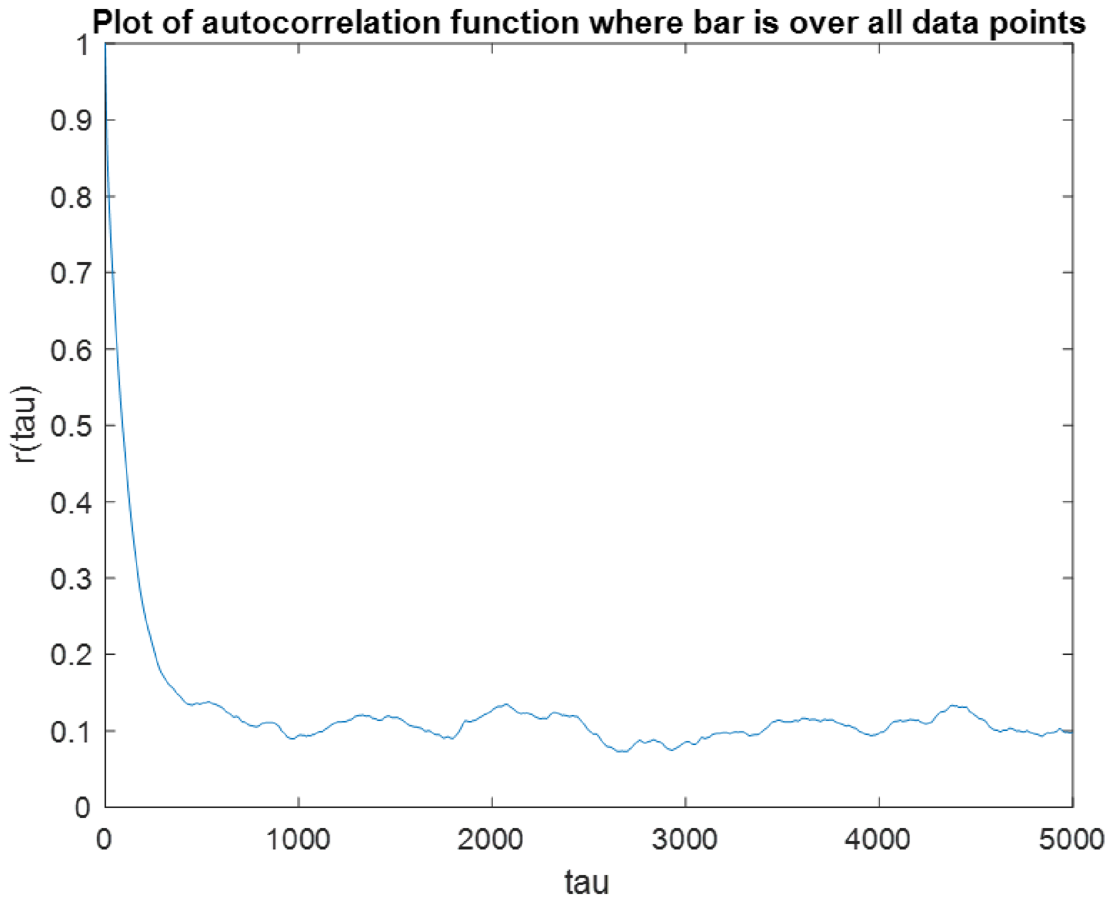
2.f)


Plot of autocorrelation function where bar is over 500 data points
1.4
1.2
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
r(tau)
0
1000
2000
3000
4000
5000
tau


Plot of autocorrelation function where bar is over 1000 data points
r(tau)
tau
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
0
1000
2000
3000
4000
5000


Plot of autocorrelation function where bar is over 10000 data points
r(tau)
tau
1
0.8
0.6
0.4
0.2
0
-0.2
0
1000
2000
3000
4000
5000


Plot of autocorrelation function where bar is over 100000 data points
r(tau)
tau
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
1000
2000
3000
4000
5000



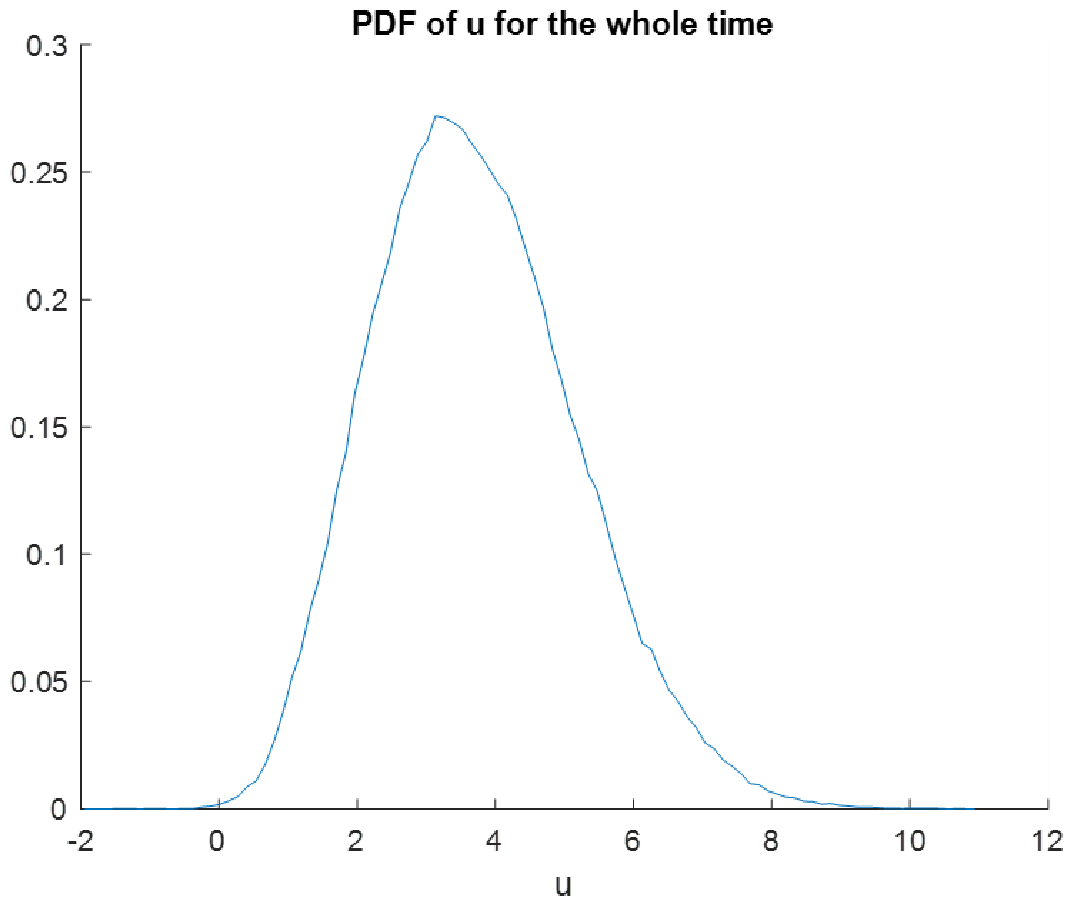
For all graphs, initially the u' and u' at del t are autocorrelated as the value of R is large and positive. However, with time progression, the value of R diminishes as the fluctuation points become uncorrelated to each other. This is because for larger values of tau the velocities at t and t+tau are not in the same coherent eddy structure. For shorter values of tau, the velocities at t and t+tau are in the same coherent eddy structure, so they have more correlation.

Interestingly, for graph plots for 500, 1000 bar mean averages, large negative values are encountered as u1(t) is auto-correlated to opposite signs of u2(t).

For higher bar averaging, there is no auto-correlation among the two u' time intervals. Yet, you see larger R auto correlation for lower bar averaging values.

In conclusion, autocorrelation is observed over smaller intervals among u' and u'del t and diminishes over higher time averaging.

2.g)




```
>> pdf_integral=trapz(v,f1)

pdf_integral =

     1

>> pdf_u_integral=trapz(v,f1.*v)

pdf_u_integral =

    3.7149
```

3. Betz Derivation (included in the last section)

4. QBLADE: Testing Implementation of BEM Theory using QBLADE (BEM Code)

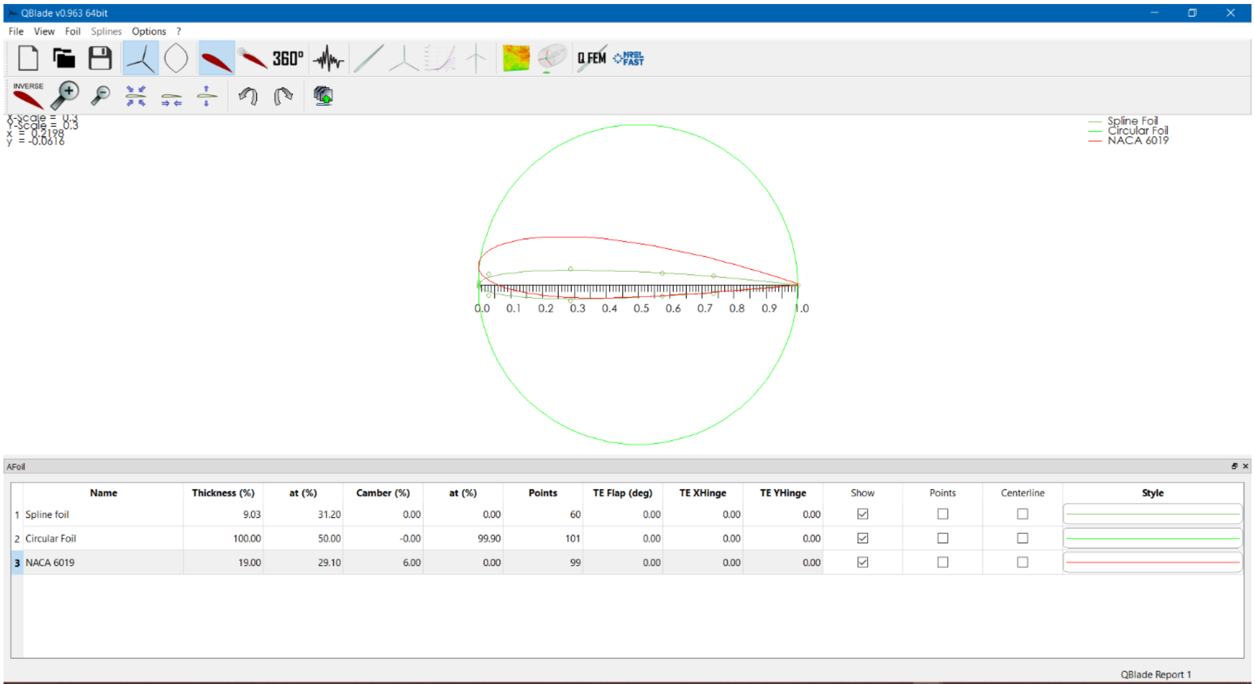


Figure 1: Importing “canned” NACA foils in QBlade

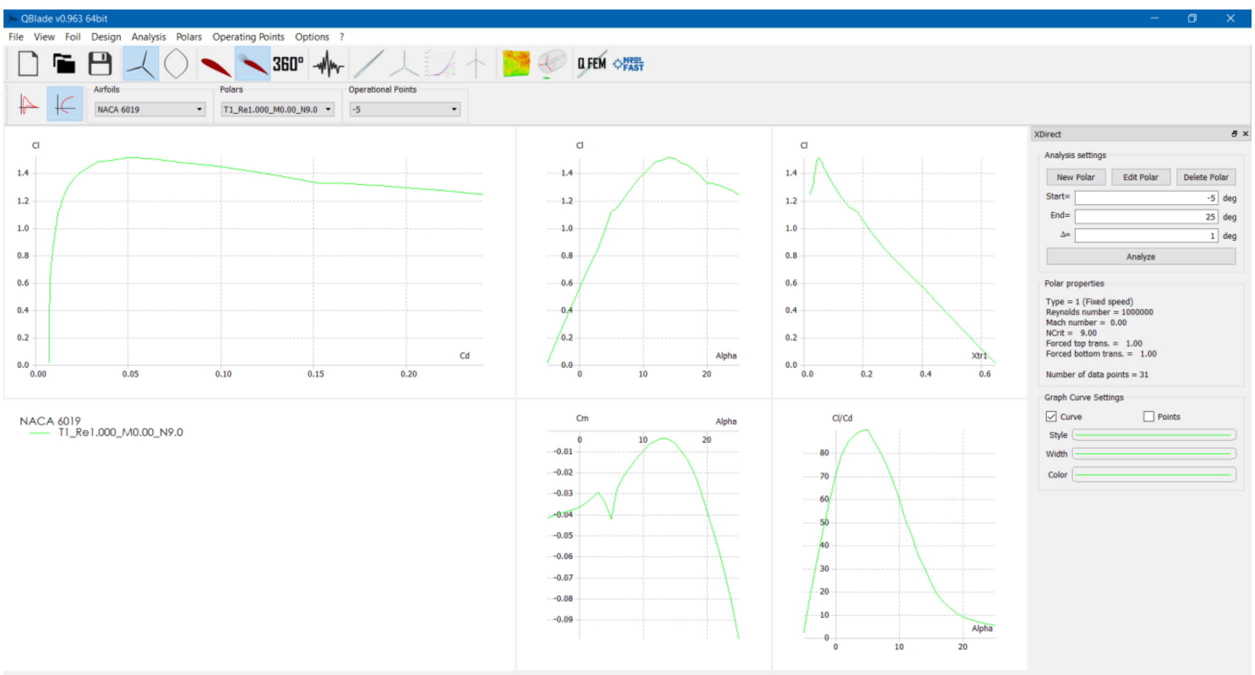


Figure 2: XFOIL Direct Analysis module to create polar to simulate flow around the airfoil

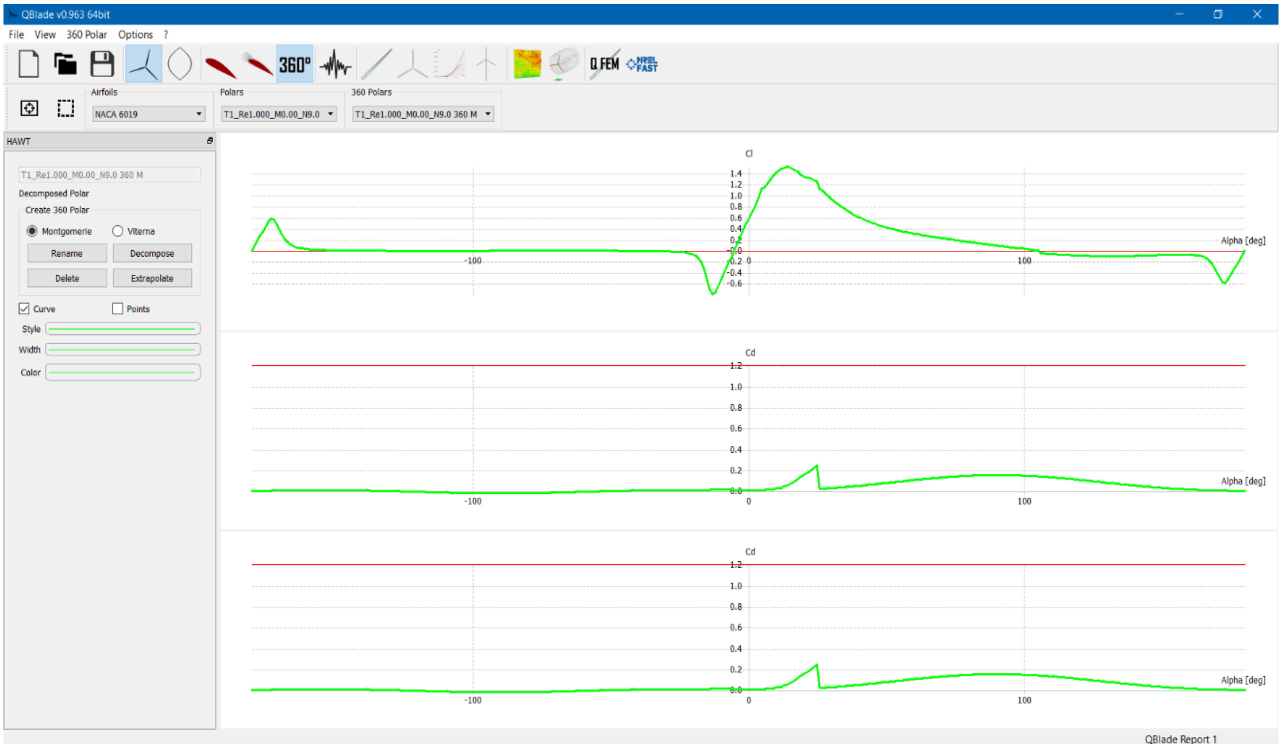


Figure 3: 360° Polar extrapolation module

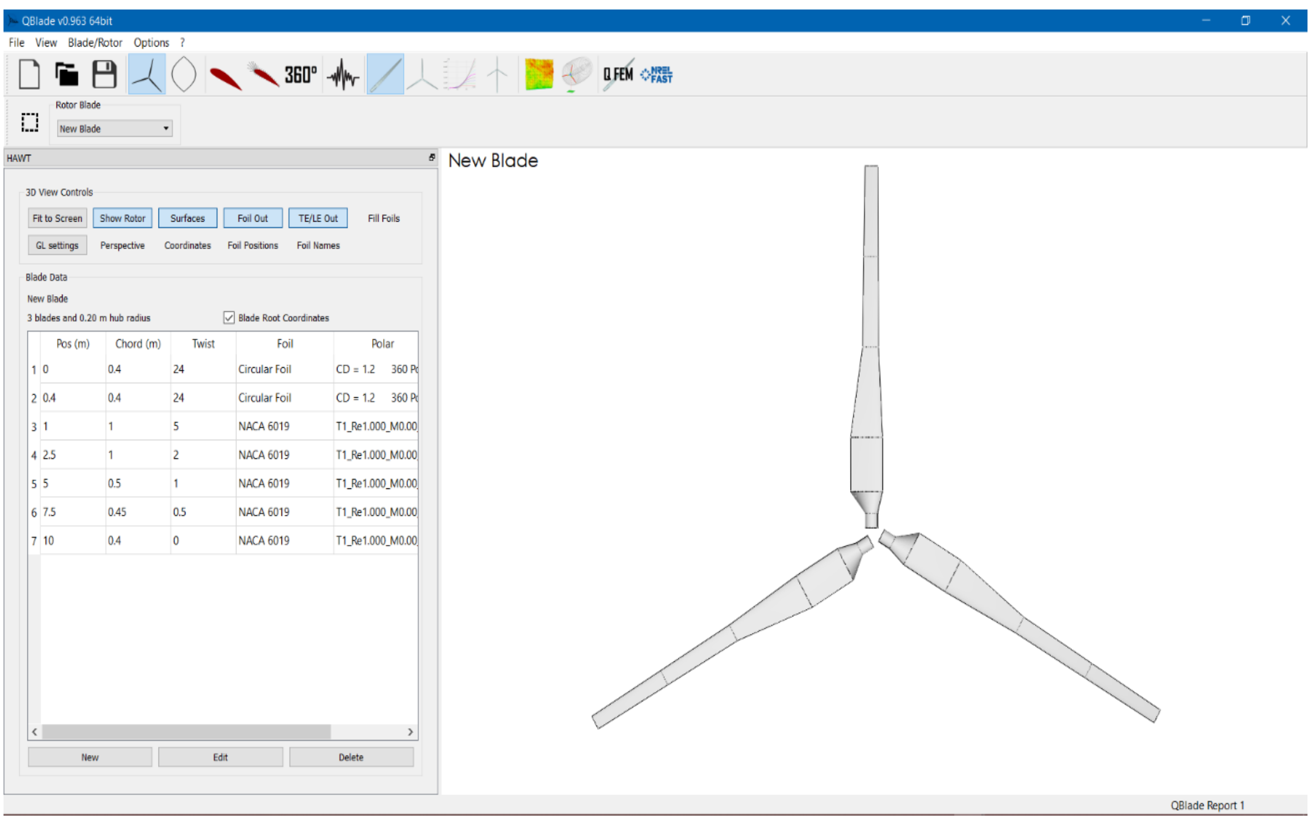


Figure 4: HAWT blade design and optimization submodule

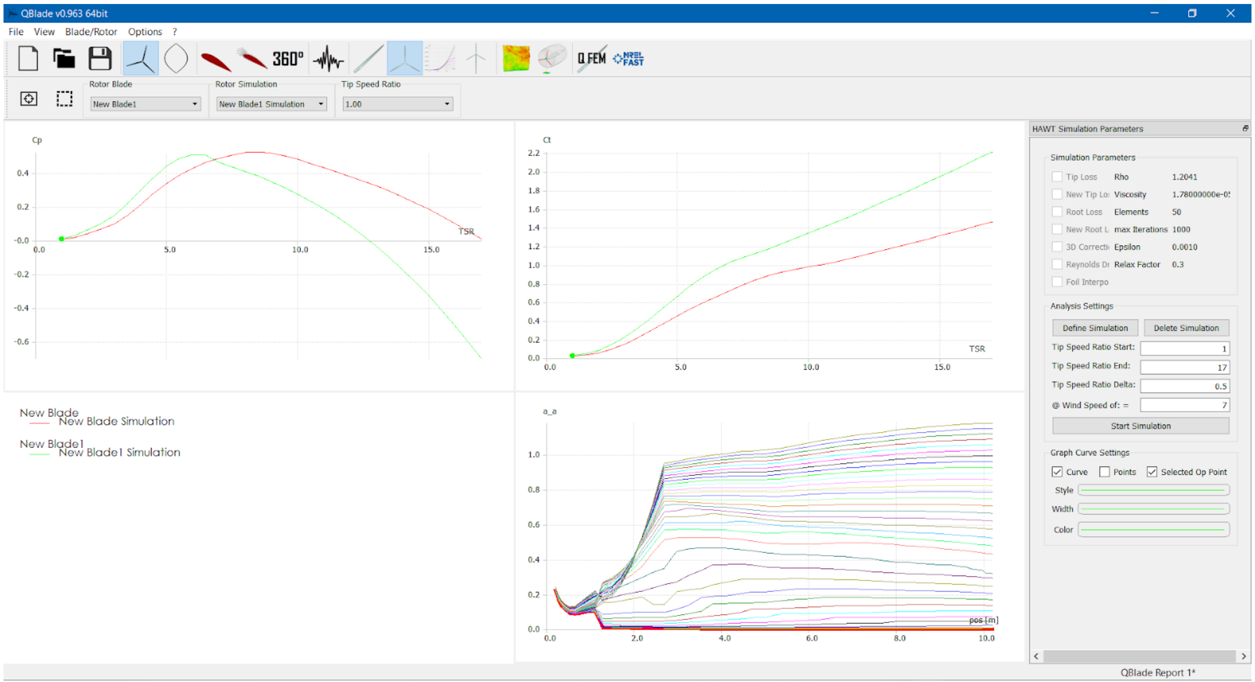


Figure 5: HAWT rotor simulation submodule

5. Reynold's Decomposition in diffusion term of Navier Stokes Equation

steps to build wind farm : Gather Data

$u_i = U_i + u_i'$

9) $\nu \dfrac{\partial^2 u_i}{\partial x_j^2} = \nu \dfrac{\partial^2 (U_i + u_i')}{\partial x_j \partial x_j} = \nu \left[\dfrac{\partial^2 U_i}{\partial x_j \partial x_j} + \dfrac{\partial^2 u_i'}{\partial x_j \partial x_j} \right]$

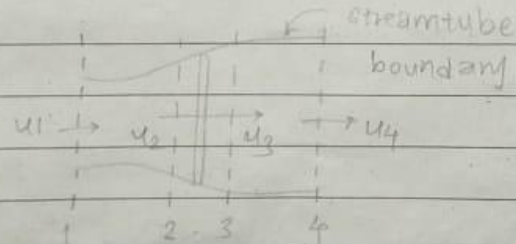
Avg $= \nu \left[\overline{\dfrac{\partial^2 U_i}{\partial x_j \partial x_j}} + \overline{\dfrac{\partial^2 u_i'}{\partial x_j \partial x_j}} \right]$ $u_i = U_i + u_i'$ mean + fluctuation

$= \nu \left[\dfrac{\partial^2 U_i}{\partial x_j \partial x_j} + \overline{\dfrac{\partial^2 u_i'}{\partial x_j \partial x_j}} \right]$ Apply Reynold's decomposition

$= \nu \left[\dfrac{\partial^2 U_i}{\partial x_j \partial x_j} + \dfrac{\partial^2 \overline{u}_i'}{\partial x_j \partial x_j} \right]$ $\overline{u}_i' = 0$ ∴ commutation rule

$= \nu \dfrac{\partial^2 U_i}{\partial x_j \partial x_j}$

3)



Assumption:
1) homogenous, incompressible, steady state fluid flow
2) No frictional drag
3) non-rotating wake

conservation of linear momentum, thrust equal & opposite to rate of change of momentum of air stream.

$T = u_1(\rho A u)_1 - u_4(\rho A u)_4$

$T = \dot{m}(u_1 - u_4)$ ... steady state flow — (A)

Bernoulli's at 1,2 & 3,4,

$p_1 + \frac{1}{2}\rho u_1^2 = p_2 + \frac{1}{2}\rho u_2^2$, — (1)

$p_3 + \frac{1}{2}\rho u_3^2 = p_4 + \frac{1}{2}\rho u_4^2$ — (2)

Assume, $p_1 = p_4$, $u_2 = u_3$

Thrust, $T = A_2(p_2 - p_3)$ — (3)

eq^n (1) − (2) & substitute in (3)

$$P_3 + \frac{1}{2}\rho u_1^2 = P_2 + \frac{1}{2}\rho u_4^2$$

$$P_2 - P_3 = \frac{1}{2}\rho(u_1^2 - u_4^2) \quad -(4)$$

Put (4) in (3)

$$T = A_2 \cdot \frac{1}{2}\rho(u_1^2 - u_4^2) \quad -(5)$$

equating thrust eqn from (5) & (A) (Bernoulli's & linear momentum

$$\dot{m}(u_1 - u_4) = A_2 \cdot \frac{1}{2}\rho(u_1^2 - u_4^2) \qquad \dot{m} = (\rho A u)_2 = (\rho A u)_4$$

$$u_2 = \frac{u_1 + u_4}{2} \quad -(6)$$

a. fractional decrease in wind velocity freestream & rotorplan

$$a = \frac{u_1 - u_2}{u_1}$$

$$u_2 = u_1(1-a) \quad -(7)$$

$$u_4 = u_1(1-2a) \quad \text{from (6)} \quad -(8)$$

from eqn (7) (8) (5)

$$T = \frac{1}{2} A_2 \rho (u_1^2 - u_4^2)$$

$$T = \frac{1}{2} A_2 \rho (u_1 + u_4)(u_1 - u_4)$$

$$T = \frac{1}{2} A_2 \rho (u_1 + u_1 - 2a u_1)(u_1 - u_1 + 2a u_1)$$

$$T = \frac{1}{2} A_2 \rho (2u_1 - 2a u_1)(+2a u_1)$$

$$T = \frac{1}{2}\rho A_2 (4a u_1^2 - 4a^2 u_1^2)$$

$$\boxed{T = \frac{1}{2}\rho A u^2 (4a(1-a))} \quad \ldots u_1 = u, \; A_2 = A$$

$C_T = \frac{\text{Thrust force}}{\text{Dynamic force}} = \frac{T}{\frac{1}{2}\rho U^2 A}$

$\boxed{C_T = 4a(1-a)}$ ... Thrust coefficient

C_T max, $\frac{dC_T}{da} = 0$

$C_T = 4a - 4a^2$

$\frac{dC_T}{da} = 4 - 8a = 0$

$\boxed{a = \frac{1}{2}}$ $\boxed{T_{max} = \frac{1}{2}\rho U^2 A}$ ... thrust at Betz Power limit

$C_{T_{max}} = 4\left(\frac{1}{2}\right) - 4\left(\frac{1}{4}\right)$

$= 2 - 1$

$\boxed{C_{T_{max}} = 1}$ ... max. thrust coefficient limit.

(ii) thrust at Betz power limit ($a = 1/3$)

$C_T = 4\left(\frac{1}{3}\right)\left[1 - \frac{1}{3}\right]$

$\boxed{C_{T_{\text{betz limit}}} = \frac{8}{9}}$ ... thrust at Betz limit.

Matlab Code:

Main file

```
close all
clear
%load sonic_segment.mat
load MAE_577_sonic_segment.mat

%b
figure(1)
formatted_time=datetime(2001,8,6,7,0,0)+seconds((time-time(1))*24*60*60);
plot(formatted_time(12001:12501),u(12001:12501))
xlim([formatted_time(12001) formatted_time(12501)])
xlabel('Time')
ylabel('u(m/s)')
title('Plot of 500 contiguous points')
%c running 15 min average of bar(u)
i=1;
j=1;
while i<252000
    u_mean(j)=mean(u(i:i+8999));
    t_mean(j)=(2*i+9000)/2;
    t_mean(j)=t_mean(j)/(10*60);
    i=i+9000;
    j=j+1;
end

formatted_time_mean=datetime(2001,8,6,7,0,0)+minutes(t_mean);
figure(2)
plot(formatted_time_mean,u_mean);
title('Running 15 min average of bar(u)')
xlabel('Time')
ylabel('bar(u)')

%d running 15 minute average of bar(u'u')
u_mean=mean(u);
for i=1:252000
    u_prime(i)=u(i)-u_mean;
    u_prime_squared(i)=u_prime(i)*u_prime(i);
end

j=1;
i=1;
while i<252000
    u_prime_squared_mean(j)=mean(u_prime_squared(i:i+8999));
    t_mean(j)=time(i+4500);
    i=i+9000;
    j=j+1;
end

u_prime_squared_mean_all_time=mean(u_prime_squared);

figure(3)
plot(formatted_time_mean,u_prime_squared_mean)
title("Running 15 min average of bar(u'u')")
xlabel('Time')
ylabel("bar(u'u')")

%e scatter plot of u and v
figure(4)
scatter(u_prime(1:500),u_prime(3401:3900))
```

```
xlabel('u')
ylabel('v')
title('Scatter plot')

%f autocorrelation
tau=[0:10:5000];
T0=100;
auto_correlation=zeros(5,length(tau));
auto_correlation_normalized=zeros(5,length(tau));
self_correlation=zeros(5,length(tau));
self_correlation_normalized=zeros(5,length(tau));

T=[500 1000 10000 100000 252000];
for j=1:length(tau)
for i=1:5
    for t=T0:T0+T(i)-1

auto_correlation(i,j)=auto_correlation(i,j)+(u_prime(wrap(t))*u_prime(wrap(t+tau(j))))
;
        self_correlation(i,j)=self_correlation(i,j)+u_prime(wrap(t))*u_prime(wrap(t));
    end
    auto_correlation(i,j)=auto_correlation(i,j)/T(i);
    self_correlation(i,j)=self_correlation(i,j)/T(i);
    auto_correlation_normalized(i,j)=auto_correlation(i,j)/self_correlation(i,j);
end
end

figure(5)
plot(tau,auto_correlation_normalized(5,:));
title('Plot of autocorrelation function where bar is over all data points')
xlabel('tau')
ylabel('r(tau)')

%g pdf of u over all time
a=-2;
b=11;
N=100;
dv=(b-a)/N;
v=zeros(N,1);
f=zeros(N,1);

for i=1:N
     for j=1:size(u)
         if(u(j)>=a+(i-1)*dv && u(j)<a+i*dv)
             f(i)=f(i)+1;
         end
     end
     v(i)=a+(i-0.5)*dv;  % mid point of wind speed bin
end

f1=f/(252000*dv);
figure(6)
plot(v,f1)
%integral
pdf_integral=trapz(v,f1);
pdf_u_integral=trapz(v,f1.*v);

title("PDF of u for the whole time")
xlabel('u')
```

Function file - wrap.m

```
function [index] = wrap(input_index)
if input_index>252000
    index = input_index-252000;
else
    index=input_index;
end
```