



**Escola Tècnica Superior d'Enginyeries  
Industrial i Aeronàutica de Terrassa**  
UNIVERSITAT POLITÈCNICA DE CATALUNYA

Titulación:  
**Ingeniería Aeronáutica**

Alumno:  
**Javier Escalera Siles**

Título PFC:  
**Estudio aerodinámico de un aerogenerador  
mediante teoría BEM (Blade Element Momentum)**

Director del PFC:  
**Pau Nualart Nieto**

Convocatoria de entrega del PFC:  
**Septiembre 2014**

Contenido de este volumen: **ANEXOS**



# Índice general

<b>Código BEM .....</b>	<b>1</b>
1.1.    Programa principal .....	1
1.2.    Funciones .....	9
1.2.1.    Input_geometry_function .....	9
1.2.1.    Input_airfoil_function.m.....	9
1.2.2.    Viterna_function.m.....	9
1.2.3.    Snel_function.m .....	10
1.2.4.    Newton_rapshon_function.m .....	10
1.2.5.    Prandtl_Coleman_function.m.....	11
1.2.6.    Coordinate_System_function.m.....	11
1.2.7.    Aerodynamic_loads_function.m.....	12
1.2.8.    Beddoes_Leishman_function.m.....	13
1.2.9.    Draw_Axial_function.m .....	14
1.2.10.    Draw_Yaw_function.m .....	16
<b>Validación Beddoes-Leishman .....</b>	<b>24</b>
<b>Validación Snel_Viterna.....</b>	<b>29</b>
<b>Archivos.txt.....</b>	<b>32</b>
1.3.    Rotor_geometry.txt.....	32
1.1.    NACA 0012 _360000.txt .....	32
1.2.    S809_1000000.txt.....	32
1.3.    S809_Xfoil_500000.txt.....	33
1.4.    S809_Experimental_500000.txt .....	36
1.5.    S809_NREL_63.txt .....	37
1.6.    S809_NREL_30.txt .....	37
<b>Bibliografía.....</b>	<b>39</b>



## **Índice de tablas**

Tabla 1 Turbina de TU Delft .....	32
Tabla 2 NACA 0012; Re = 360.000 .....	32
Tabla 3 NREL S809 Airfoil Data from OSU; Re = 1,000,000 .....	32
Tabla 4 S809 Xfoil; Re = 500.000 .....	33
Tabla 5 NREL S809 Airfoil Data from CSU; Re = 500,000 (Sin rotación) .....	36
Tabla 6 NREL S809 Airfoil Data from NASA-Ames Tests: 30% Span .....	37
Tabla 7 NREL S809 Airfoil Data from NASA-Ames Tests: 30% Span .....	37



## Código BEM

### 1.1. Programa principal

```
close all; clc; clear all;
%% Variables de entrada y calculos previos

%Load data
%1-Airfoil (Aerodynamics)Ct
[archivo, ruta] = uigetfile('.txt', 'Seleccione perfil');
[Alpha,C1,Cd,Cm]=Input_airfoil_function(archivo, ruta);

%2-Twist
[archivo, ruta] = uigetfile('.txt', 'Seleccione geometria');
[x,c,twist]=Input_geometry_function(archivo, ruta);

%Input rotor geometry details
N=2; %Numero de palas
Rt=0.6; %Radio de punta de pala [m]
Rh=0.18; %Radio raíz de pala (cabeza de rotor) [m]
da=0.75; %Distancia hub-centre -- pivot centre[m]
beta=0; %Coning angle (Angulos pequeños) [grados]
ji=0; %Tilt angle (Angulos pequeños) [grados]

%Input rotor operating details
omega=699*2*pi/60; %rpm -- velocidad angular rotor[rad/s]
U=[0; 5.5; 0]; %Velocidad viento ejes X-Y-Z [m/s]
theta_pitch=0; %Angulo de cabeceo [grados]
yaw=30; %Angulo de guiñada [grados]
yaw_dt=0;
rho=1.225; %Densidad[kg/m3]
%Estructural
ni=0; %Local blade deflection [m]
ni_dt=0;
ni_slope=0; %Local blade slope due to flexure

%Input programming details
el_b=7; %Numero de elementos por pala
el_tau=12; %Numero de elementos en azimuth por revolucion
tolerancia=0.001; %Tolerancia (ua vs ua_c & ut vs ut_c)
Nrev=1; %Numero de revoluciones
contador=0;
flag_TL=1; %Tip loss factor && Wake (1 o 0)
flag_BL=1; %Beddoes Leishman (1 o 0)
flag_V=0; %Viterna (1 o 0)
flag_S=0; %Snel(1 o 0)
flag_Draw=0; %Draw
%Newton rapshon
tol = 0.00001; %Tolerancia (Newton rapshon)
x0=10; %Valor inicial

%Calculos previos & discretizacion
U_inf=sqrt(U(1)^2+U(2)^2+U(3)^2); %Módulo velocidad viento[m/s]
```

```

lambda=(omega*Rt)/U_inf; %Tip speed ratio
A=pi*Rt^2; %Area rotor (m^2)
delta_el=(Rt-Rh)/el_b; %Incremento radio [m]
delta_tau=360/el_tau; %Incremento angular [grados]
dt=delta_tau/(omega*180/pi); %Tiempo [s]

r_b=Rh:delta_el:(Rt-(delta_el/2)); %Radio elemento (1 pala)
r=r_b;
for i=1:N-1
    r=horzcat(r,r_b); %Radio elemento (N palas)
end

theta_twist=interp1(x,twist,r,'cubic')*(pi/180); %rad
theta=theta_twist+theta_pitch*pi/180;
c=interp1(x,c,r,'cubic');

%Interpolación coeficientes
c_mean=sum(c)/(el_b*N);
AR=Rt/c_mean;
Alpha_s=Alpha(Cl==max(Cl))*pi/180; %Alpha=Alpha_stall [rad]
Alpha_final=Alpha(length(Alpha))*pi/180; %Alpha=Alpha_final_XFOIL [rad]
Cl_s=max(Cl); %CL=CL_stall
Cd_s=Cd((Cl==max(Cl))); %CD=CD_stall

alfa_0=Alpha(abs(Cl-0)==min(abs(Cl-0)));
Cd0=Cd(abs(Alpha-0)==min(abs(Alpha-0)));
pos_Alpha0=abs(Alpha-0)==min(abs(Alpha-0));
pos_Alpha5=abs(Alpha-5)==min(abs(Alpha-5));
Cl_alfa=((Cl(pos_Alpha5)-Cl(pos_Alpha0))/(Alpha(pos_Alpha5)-... Alpha(pos_Alpha0)));

if flag_V==1
    [Alpha_2D, Cl_2D, Cd_2D]=Viterna_function(AR,Cl_s,... Alpha_s,Cd_s,Alpha_final,Alpha,Cl,Cd);
else
    Alpha_2D=Alpha;
    Cl_2D=Cl;
    Cd_2D=Cd;
end

for i=1:length(Alpha_2D)
    Cn(i)=Cl_2D(i)*cos(Alpha_2D(i)*pi/180)+Cd_2D(i)*... sin(Alpha_2D(i)*pi/180); %Eje eta
    Cc(i)=Cl_2D(i)*sin(Alpha_2D(i)*pi/180)-Cd_2D(i)*... cos(Alpha_2D(i)*pi/180); %Eje dseta
end
Cn_alfa=Cl_alfa*pi; %[1/rad] (Aproximadamente iguales)

for i=1:length(Alpha_2D)...
    f_s(i)=(2*(sqrt(Cn(i)/(Cn_alfa*((Alpha_2D(i)- alfa_0)*pi/180)))-1)^2;
    if f_s(i)>1 || isnan(f_s(i))
        f_s(i)=1;
    elseif f_s(i)<0
        f_s(i)=0;
    end
end

```

## Código BEM

```
%%Beddoes Leishman (Condiciones iniciales)
%BL
(X_old,Y_old,D_old,Dp_old,Df_old,Cn_p,f1,Cn_v,cv,delta_alfa,Cl_u,Cd_u)
BL=zeros(1,N*el_b,12);
BL(1,:,:)=1;      %f'

%Posición azimutal de cada elemento de pala a lo largo del tiempo
for j=1:1:Nrev*el_tau
    for i=1:1:N*el_b
        pos(i,j)=(j-1)*delta_tau;
        if i>el_b
            pos(i,j)=120+(j-1)*delta_tau;
        end
        if i>2*el_b
            pos(i,j)=240+(j-1)*delta_tau;
        end
        while pos(i,j)>360
            pos(i,j)=pos(i,j)-360;
        end
    end
end

%% Proceso iterativo
for j=1:1:Nrev*el_tau %Cada time step

    %Inicializacion velocidad inducida
    for i=1:1:N*el_b
        if j==1 % Primer time step
            ua(i,j)=0;
            ut(i,j)=0;
        else
            ua(i,j)=ua(i,j-1);
            ut(i,j)=ut(i,j-1);
            Inflow(i,j)=Inflow(i,j-1);
        end
    end
    stop=0; %Parametro de control

    while (stop~=N*el_b)
        k=1;
        t=1;
        stop=0;
        for i=1:1:N*el_b %Cada elemento de pala
            %Velocidad absoluta (Ejes eta-dseta-xi)
            Va_1(i,j)=r(i)*omega*cos(theta(i))-yaw_dt*((da+beta*r(i))*...
                cos(theta(i))+ni)*cos(pos(i,j)*pi/180)-yaw_dt*r(i)*...
                sin(theta(i))*sin(pos(i,j)*pi/180);
            Va_2(i,j)=r(i)*omega*sin(theta(i))+ni_dt-yaw_dt*(da+beta*...
                r(i))*sin(theta(i))*cos(pos(i,j)*pi/180)+yaw_dt*r(i)*...
                cos(theta(i))*sin(pos(i,j)*pi/180);
            Va_3(i,j)=(ni_slope*r(i)-ni)*omega*sin(theta(i))-yaw_dt*...
                (da+(ni-ni_slope*r(i))*cos(theta(i))*sin(pos(i,j)*pi/180));

            Va(:,i,j)=[Va_1(i,j), Va_2(i,j), Va_3(i,j)];

            %Prandtl tip/root loss factor:
        end
    end
end
```

```

if flag_TL==0
    f(i,j)=1;
    Fsa(i,j)=1;
else
    if ua(i,j)~=0
        [f(i,j),Fsa(i,j)]=Prandtl_Coleman_function...
            (Inflow(i,j)*pi/180,U_inf,ua(i,j),ut(i,j),...
            pos(i,j)*pi/180,yaw*pi/180,Rt,Rh,r(i),N,omega);
    else
        f(i,j)=1;
        Fsa(i,j)=1;
    end
end

%Velocidad inducida (Ejes xr-yr-zr)
u_1(i,j)=ut(i,j);
u_2(i,j)=ua(i,j);
u_3(i,j)=0;

%Velocidad relativa (Va+U+u en Ejes x-y-z)
[V(:,i,j), K, A4, K_2]=Coordinate_System_function(i,j, ...
    U,Va,yaw*pi/180,pos*pi/180,theta,ji,beta*pi/180,ni_slope);
%Velocidad inducida (Ejes x-y-z)
u(:,i,j)=A4*[u_1(i,j), u_2(i,j), u_3(i,j)]';
Vr(i,j)=sqrt((V(1,i,j)+u(1,i,j))^2+(V(2,i,j)+u(2,i,j))^2);

%Angulo de ataque
Inflow(i,j)=(180/pi)*atan(-((V(2,i,j)+u(2,i,j))/...
    (V(1,i,j)+u(1,i,j))));;
alfa(i,j)=Inflow(i,j)-theta(i)*180/pi;

%Calculo estacionario
%Fuerzas aerodinamicas (Cl,Cd,Cm)
Cl_t(i,j)=interp1(Alpha_2D(1,:),Cl_2D(1,:),...
    alfa(i,j),'linear','extrap');
Cd_t(i,j)=interp1(Alpha_2D(1,:),Cd_2D(1,:),...
    alfa(i,j),'linear','extrap');
Cm_t(i,j)=interp1(Alpha(1,:),Cm(1,:),alfa(i,j)...
    , 'linear','extrap');

if flag_S==1
    if (c(i)/r(i))<0.4
        [alfa(i,j),Cl_t(i,j)]=Snel_function(alfa_0*...
            pi/180,Cl_alfa*pi/180,c(i),r(i),Inflow(i,j),...
            alfa(i,j),Cl_t(i,j));
    end
end

%Fuerzas aerodinamicas (Ejes Cn,Cc)
Cn_t(i,j)=interp1(Alpha_2D(1,:),Cn(1,:),alfa(i,j),...
    'linear','extrap');
Cc_t(i,j)=interp1(Alpha_2D(1,:),Cc(1,:),alfa(i,j),...
    'linear','extrap');

%Calculo no estacionario (Beddoes-Leishman)
if flag_BL==1
    if j>1
        [BL(j,i,:)]=Beddoes_leishman_function(Alpha_2D, ...

```

## Código BEM

```
f_s,Cd0,dt,c(i),Vr(i,j),BL(j-1,i,:),Cn_alfa,...  
alfa_0*pi/180,alfa(i,j)*pi/180,alfa(i,j-1)*pi/180);  
  
else  
    BL(j,i,11)=Cl_t(i,j); %Cl_u  
    BL(j,i,12)=Cd_t(i,j); %Cd_u  
end  
Cl_t(i,j)=BL(j,i,11);  
Cd_t(i,j)=BL(j,i,12);  
end  
  
%Reordenación de vectores (Filas-Elemento 1, Elemento N,...;  
%Columnas-Pala 1,Pala N)  
%Util a la hora de resolver BEM equation  
if rem(i,el_b)==0  
    V_1(t,k)=abs(V(1,i,j));  
    V_2(t,k)=abs(V(2,i,j));  
    Cl_t2(t,k)=Cl_t(i,j);  
    Cd_t2(t,k)=Cd_t(i,j);  
    Inflow_2(t,k)=Inflow(i,j)*pi/180;  
    c_2(t,k)=c(i);  
    f_2(t,k)=f(i,j);  
    Fsa_2(t,k)=Fsa(i,j);  
    k=k+1;  
    t=1;  
else  
    V_1(t,k)=abs(V(1,i,j));  
    V_2(t,k)=abs(V(2,i,j));  
    Cl_t2(t,k)=Cl_t(i,j);  
    Cd_t2(t,k)=Cd_t(i,j);  
    Inflow_2(t,k)=Inflow(i,j)*pi/180;  
    c_2(t,k)=c(i);  
    f_2(t,k)=f(i,j);  
    Fsa_2(t,k)=Fsa(i,j);  
    t=t+1;  
end  
end  
  
%Valores medios (Momentum equation)  
V_1=sum(V_1,2)/N;  
V_2=sum(V_2,2)/N;  
f_medio=sum(f_2,2)/N;  
for i=1:el_b  
    if f_medio(i)==0; %r/R=Rh/Rt;  
        f_medio(i)=f_medio(i+1);  
    end  
end  
Fsa_medio=sum(Fsa_2,2)/N;  
  
%BEM Equation (Newton-Raphson)  
%1-ua<0 (Ejes xr-yr-zr) -- Valor radial medio  
sigma=(c_2(:,1)./(2*pi.*r_b'));  
RS_ua=(Cl_t2.*(cos(Inflow_2))+Cd_t2.*(sin(Inflow_2)))./. . .  
(-4*sin(Inflow_2).^2);  
RS_ua=sigma.*sum(RS_ua,2);  
syms x  
cf_ua=(x.*sqrt((U_inf*cos(yaw*pi/180)+x).^2+U_inf^2* . . .
```

```

sin(yaw*pi/180)^2))-RS_ua.*((V_2+x./f_medio.*Fsa_medio).^2;
[ua_sol]=Newton_rapshon_function(cf_ua,tol,x0);

%Caso particular: ua>0.38*U_inf

for i=1:1:el_b
    if abs(ua_sol(i))>0.38*U_inf
        ua_sol(i)=(RS_ua(i)-0.5776*U_inf^2)/(-0.96*U_inf);
    end
end

%ua para cada elemento de pala
ua_c=ua_sol;
for i=1:N-1
    ua_c=horzcat(ua_c,ua_sol);
end
ua_c=ua_c';

%2-ut<0 (Ejes xr-yr-zr) -- Valor radial medio
RS_ut=(Cl_t2.*((sin(Inflow_2))-Cd_t2.*((cos(Inflow_2)))).....
/(-4*cos(Inflow_2).^2);
RS_ut=sigma.*sum(RS_ut,2);
syms x
cf_ut=(x.*sqrt((U_inf*cos(yaw*pi/180)+ua_sol').^2+U_inf^2*...
    sin(yaw*pi/180)^2))-RS_ut.*((V_1-x./f_medio).^2;
[ut_sol]=Newton_rapshon_function(cf_ut,tol,x0);

%ut para cada elemento de pala
ut_c=ut_sol;
for i=1:N-1
    ut_c=horzcat(ut_c,ut_sol);
end
ut_c= ut_c';

%Convergencia
for i=1:1:el_b*N
    if abs(ua(i,j)-ua_c(i))>tolerancia || abs(ut(i,j)-ut_c(i))...
        >tolerancia
        ua(i,j)=ua_c(i);
        ut(i,j)=ut_c(i);
        contador=contador+1; %Variable control
    else
        stop=stop+1
        ua(i,j)=ua_c(i);
        ut(i,j)=ut_c(i);
    end
end
end
end

%% Cargas aerodinamicas

x_ac=c./4; %Distancia borde ataque - centro aerodinamico
x_ea=c./4; %Distancia borde ataque - centro elastico
e_ea=(x_ac-x_ea)./c; %Normalmente 0

%BEM results (0.45 & 0.85)

```

## Código BEM

```
for j=1:1:Nrev*el_tau
    delta_T(:,j)=0.5*rho*c'*(Vr(:,j).^2).*(Cl_t(:,j).*...
    (cos(Inflo...)
```

```
CQ=abs(Smy/(0.5*rho*A*U_inf^2*Rt));%Torque Coefficient (C_My0=CQ)
%C_Mz0=abs(Smz/(0.5*rho*A*U_inf^2*Rt));%Yawing Moment Coefficient
CP=abs(P/(0.5*rho*A*U_inf^3)); %Power output coefficient

%% Analisis resultados (Gráficos)

ua_tip=ua./f;
a_tip=ua_tip./U_inf;
a=ua./U_inf;
at=ut./(omega*Rt);

if flag_Draw==1
    if yaw==0 %Pala 1 = Pala 2

Draw_Axial_function(r_R,a,a_tip,Vr,alfa,Cl_t,el_b,el_tau, ...
    Nrev,f,delta_T,delta_Q,delta_T_extr,delta_Q_extr)
    else

Draw_Yaw_function(pos,alfa,Vr,Cl_t,a,yaw,el_b,r_R,delta_T, ...
    delta_Q)
    end
end
```

## 1.2. Funciones

### 1.2.1. Input\_geometry\_function.m

```
function [a,b,c] = Input_geometry_function(archivo, ruta)
if archivo==0
    return;
else fid =fopen([ruta archivo], 'r');
A=textscan(fid, '%f %f %f', 'headerlines',1);
A=cell2mat(A);
fclose(fid);
[filas, columnas]=size(A);
for i=1:filas
    a(i)=A(i,1);
    b(i)=A(i,2);
    c(i)=A(i,3);
end
end
```

### 1.2.1. Input\_airfoil\_function.m

```
function [a,b,c,d] = Input_airfoil_function(archivo, ruta)
if archivo==0
    return;
else fid =fopen([ruta archivo], 'r');
A=textscan(fid, '%f %f %f %f', 'headerlines',1);
A=cell2mat(A);
fclose(fid);
[filas, columnas]=size(A);
for i=1:filas
    a(i)=A(i,1);
    b(i)=A(i,2);
    c(i)=A(i,3);
    d(i)=A(i,4);
end
end
```

### 1.2.2. Viterna\_function.m

```
function [alfa, Cl, Cd]=...
Viterna_function(AR,Cl_s,Alpha_s,Cd_s,Alpha_final,Alpha,Cl,Cd)

%Viterna_Corrigan extrapolation (CL & CD)
if AR<=50
    Cd_max=1.11+0.018*AR;
else
    Cd_max=2.01;
end

Kl=(Cl_s-Cd_max*sin(Alpha_s)*cos(Alpha_s))*(sin(Alpha_s)/cos(Alpha_s)^2);
```

```

Kd=(Cd_s-Cd_max*sin(Alpha_s)^2)/cos(Alpha_s);

%Alfa_s<Alfa<90
alfa_1=((Alpha_final+pi/180):pi/180:pi/2);
for i=1:1:length(alfa_1)
    Cl_1(i)=(Cd_max*sin(alfa_1(i))*cos(alfa_1(i)))+Kl*...
        (cos(alfa_1(i))^2/sin(alfa_1(i)));
    Cd_1(i)=Cd_max*sin(alfa_1(i))^2+Kd*cos(alfa_1(i));
end

Cl=[Cl Cl_1];
Cd=[Cd Cd_1];
alfa=[Alpha*pi/180 alfa_1]*180/pi;

```

### 1.2.3. Snel\_function.m

```

function
[Alpha_2,Cl_3d]=Snel_function(Alpha_0,Cl_alpha,c,r,inflow,alfa,Cl)

%Snel et al & Lindenburg
c_r=c/r;
a=3.1;
b=2;

k=cos(inflow*pi/180)^2;    %=(lambda^2)/(1+lambda^2);
pos_alfa30=find(abs((round(alfa)-30))==min(abs(round(alfa)-30)));
delta_Cl(pos_alfa30)=(Cl_alpha*sin(alfa(pos_alfa30)*...
    pi/180-Alpha_0))-Cl(pos_alfa30);
Cl_lim=a*(c_r)^b*k*delta_Cl(pos_alfa30);

for i=1:1:length(alfa)
    if alfa(i)<=30
        delta_Cl(i)=(Cl_alpha*sin(alfa(i)*pi/180-Alpha_0))-Cl(i);
        Cl_3d(i)=Cl(i)+a*k*(c_r)^b*delta_Cl(i);
        Alpha_2(i)=alfa(i);
    elseif alfa(i)<=60
        Cl_3d(i)=Cl(i)+(Cl_lim-Cl_lim/30*(alfa(i)-30));
        Alpha_2(i)=alfa(i);
    else
        Cl_3d(i)=Cl(i);
        Alpha_2(i)=alfa(i);
    end
end

```

### 1.2.4. Newton\_rapshon\_function.m

```

function [sol]=Newton_rapshon_function(f,tol,x0)
syms x;

for i=1:1:length(f)
    x = x0;
    dx = tol + 1;
    count=0;

```

## Código BEM

```
while (abs(dx) > tol)
    dx = eval(f(i)/(diff(f(i)))); 
    x = x - dx;
    count = count + 1;
    if (count > 300)
        fprintf('Error...! No converge !!! \n');
        break;
    end
end
sol(i)=x;
end
```

### 1.2.5. Prandtl\_Coleman\_function.m

```
function [f,Fsa]=Prandtl_Coleman_function(inflow,U,ua,ut, ...
pos,yaw,Rt,Rh,r,B,omega)
%Input=U_inf,pos(i,j),ji,yaw,Rt,Rh,r,B,Inflow
%Output = f, Fsa

mu_r=r/Rt;
mu=Rh/Rt;

ji_s=atan(U*sin(yaw)/(U*cos(yaw)+ua)); %ua=70% 80%

K=tan(ji_s/2);
Fsa=(1+K*mu_r*sin(pos));

f_t=2/pi*acos(exp(-((B/2*(1-mu_r))/(mu_r*sin(inflow)))); 
f_r=2/pi*acos(exp(-((B/2*(mu_r-mu))/(mu*sin(inflow)))); 
f=f_t*f_r;
```

### 1.2.6. Coordinate\_System\_function.m

```
%Input(U=[vx vy vz];[fil col] Va=[3 nel*N] u=[3 nel*N]--- 1 time step
% Angulos psi, phi, theta, ji, beta, ni_slope

function [V, K, A4, K_2]=Coordinate_System_function...
(i,j,U,Va,psi,phi,theta,ji,beta,ni_slope)

A1=[cos(psi) sin(psi) 0; -sin(psi) cos(psi) 0; 0 0 1];
A2=[1 0 0; 0 1 -ji; 0 ji 1];
A3=[cos(phi(i,j)) 0 -sin(phi(i,j)); 0 1 0; sin(phi(i,j)) 0
cos(phi(i,j))];
A4=[1 0 0; 0 1 -beta; 0 beta 1];
A5=[cos(theta(i)) -sin(theta(i)) 0; sin(theta(i)) cos(theta(i)) 0;
0 0 1];
A6=[1 0 0; 0 1 -ni_slope; 0 ni_slope 1];

S=A4*A3*A2*A1;
S_2=A6*A5*A4*A3*A2*A1;
S2=inv(A6*A5);
K=inv(S);
```

```
K_2=inv(S_2);
V=S*U-S2*Va(:,i);
```

### 1.2.7. Aerodynamic\_loads\_function.m

```
%input (pos,r,da,i,j)
function [Sy,Smy]=Aerodynamic_loads_function...
(delta_el,Nrev,el_tau,el_b,N,r,pos,da,beta,delta_T, ...
delta_Q,theta,ni_slope,U,Va)

for j=1:1:Nrev*el_tau
    Sy(j)=0;
    Smy(j)=0;

    for i=1:1:el_b*N

        delta=[delta_Q(i,j)/r(i)*delta_el,delta_T(i,j)*delta_el,0];
        %pos[grados],r,da
        v0=[r(i)*sin(pos(i,j)*pi/180),da+beta*r(i),r(i)*cos(pos(i,j)*pi/180)];
        [V(:,i,j), K, A4, K_2]=Coordinate_System_function...
            (i,j,U,Va,0,pos*pi/180,theta,0,beta*pi/180,ni_slope);

        Vh=K*delta';
        Mo=cross(v0,Vh)';

        %Vh=delta_T(i,j)*delta_el;
        %Mo=delta_Q(i,j)*delta_el;

        if i==1 || q==2
            Sy(j)=Sy(j)+0.5*Vh(2); %Axial Thrust
            Smy(j)=Smy(j)+0.5*Mo(2);
            q=1;
        elseif rem(i,el_b)==0
            Sy(j)=Sy(j)+3/2*Vh(2); %Axial Thrust
            Smy(j)=Smy(j)+3/2*Mo(2);
            q=2;
        else
            Sy(j)=Sy(j)+Vh(2); %Axial Thrust
            Smy(j)=Smy(j)+Mo(2);
            q=1;
        end
    end
end
```

### 1.2.8. Beddoes\_Leishman\_function.m

```
function [BL]=Beddoes_leishman_function...
(Alpha,f,Cd0,delta_t,c,V_rel,BL,Cn_alfa,alfa_0,alfa,alfa_old)

a=340;
delta_s=(V_rel*delta_t)/(c/2);
M=V_rel/a;
beta=sqrt(1-M^2);

%Condiciones iniciales
X_old=BL(1);
Y_old=BL(2);
D_old=BL(3);
Dp_old=BL(4);
Df_old=BL(5);
Cn_p_old=BL(6);
f1_old=BL(7);
Cn_v_old=BL(8);
cv_old=BL(9);
delta_alfa_old=BL(10);

%Parametros experimentales
%Flujo adherido
A1=0.3;
A2=0.7;
b1=0.14;
b2=0.53;
K_alfa=0.75;
%Flujo desprendido
Tp=1.5;
Tf=5;
Tv=6;
Ti=c/a;
St=0.19;
etha=0.95;
Tvl=5;

delta_alfa=alfa-alfa_old;

%Modulo 1:

%Componente circulatoria
X=X_old*exp(-b1*beta^2*delta_s)+A1*delta_alfa*exp(-b1*beta^2*delta_s/2);
Y=Y_old*exp(-b2*beta^2*delta_s)+A2*delta_alfa*exp(-b2*beta^2*delta_s/2);
alfa_E=alfa-X-Y;
Cn_c=Cn_alfa*(alfa_E-alfa_0);
%Componente impulsiva
D=D_old*exp(-delta_t/(K_alfa*Ti))+((delta_alfa-delta_alfa_old)/delta_t)*...
exp(-delta_t/(2*K_alfa*Ti));
Cn_i=((4*K_alfa*c)/V_rel)*(delta_alfa/delta_t-D);
%Total
Cn_p=Cn_i+Cn_c;

%Modulo 2:

%Presion
```

```

Dp=Dp_old*exp(-delta_s/Tp)+(Cn_p-Cn_p_old)*exp(-delta_s/(2*Tp));
dCn=Cn_c-Dp;
alfa_f=dCn/Cn_alfa;
f1=interp1(Alpha(1,:),f(1,:),alfa_f*180/pi,'linear','extrap');
%Viscosidad
Df=Df_old*exp(-delta_s/Tf)+(f1-f1_old)*exp(-delta_s/(2*Tf));
f2=f1-Df;
Cn_f=Cn_alfa*((1+sqrt(f2))/2)^2*(alfa_E-alfa_0)+Cn_i;

%Modulo 3:

cv=Cn_c-Cn_f;
tau_v=Tvl+(2*sqrt(1-f2))/St;
if (tau_v<Tvl)
    TV=TV;
else
    TV=TV/2;
end

Cn_v=Cn_v_old*exp(-delta_s/TV)+(cv-cv_old)*exp(-delta_s/(2*TV));

%Total (Ejes cuerpo)
Cn_u=Cn_f+Cn_v;
Cc_u=etha*Cn_alfa*(alfa_E-alfa_0)^2*sqrt(f2);

%Total (Ejes viento)
Cl_u=Cn_u*cos(alfa)+Cc_u*sin(alfa);
Cd_u=Cn_u*sin(alfa)-Cc_u*cos(alfa)+Cd0;

BL(1)=X;
BL(2)=Y;
BL(3)=D;
BL(4)=Dp;
BL(5)=Df;
BL(6)=Cn_p;
BL(7)=f1;
BL(8)=Cn_v;
BL(9)=cv;
BL(10)=delta_alfa;
BL(11)=Cl_u;
BL(12)=Cd_u;

```

### 1.2.9. Draw\_Axial\_function.m

```

function Draw_Axial_function(r_R,a,a_tip,Vr,alfa,C1_t,el_b,el_tau, ...
    Nrev,f,delta_T,delta_Q,delta_T_extr,delta_Q_extr)

r1=r_R((1:el_b))'; %Pala 1
al=a(1:el_b,1:el_tau*Nrev); %Pala 1
al_tip=a_tip(1:el_b,1:el_tau*Nrev); %Pala 1
Vr1=Vr(1:el_b,1:el_tau*Nrev); %Pala 1
alfal=alfa(1:el_b,1:el_tau*Nrev); %Pala 1
C11=C1_t(1:el_b,1:el_tau*Nrev); %Pala 1
f1=f(1:el_b,1:el_tau*Nrev); %Pala 1

```

## Código BEM

```
delta_T1=delta_T(1:el_b,1:el_tau*Nrev); %Pala 1
delta_Q1=delta_Q(1:el_b,1:el_tau*Nrev); %Pala 1
delta_T1_extr=delta_T_extr(1:el_b,1:el_tau*Nrev); %Pala 1
delta_Q1_extr=delta_Q_extr(1:el_b,1:el_tau*Nrev); %Pala 1

%Datos experimentales
r_R_Exp=[0.4 0.5 0.6 0.7 0.8 0.9];
r_R_Exp_a=[0.4 0.5 0.6 0.7 0.8 0.9 1];
r_R_Exp_deltaT=[0.3 0.4 0.43 0.46 0.5 0.6 0.7 0.8 0.83 ...
    0.86 0.9 0.93 0.96 1];
r_R_Exp_deltaQ=[0.3 0.4 0.43 0.46 0.5 0.53 0.6 0.63 0.66 ...
    0.7 0.76 0.8 0.83 0.86 0.9 0.93 1];
Exp_alfa=[8 5.8 4.6 4 3.6 3.8];
Exp_Vr=[18.2 22.5 26.8 31.1 35.5 39.8];
Exp_a=[-0.24 -0.27 -0.28 -0.29 -0.30 -0.29 -0.11];
Exp_C1=[0.68 0.62 0.5 0.42 0.4 0.31];
Exp_deltaT=[0 11 13 14 15 16.5 20 24.5 25.6 25.6 24 20 12 0];
Exp_deltaQ=[0 0.52 0.62 0.7 0.74 0.78 0.78 0.79 0.8 0.82 0.92 ...
    0.97 0.97 0.91 0.81 0.62 0];

figure (1)
plot(r1,f1(:,1))
hold on
xlim([0.2 1])
ylim([0 1.1])
xlabel('r/R')
ylabel('f')
grid on

figure (2)
plot(r1,alfal(:,1))
hold on
plot(r_R_Exp,Exp_alfa,'-ro')
xlim([0.35 0.95])
xlabel('r/R')
ylabel('Alfa(°)')
grid on
legend('BEM','Experimental','Location','NorthEast')

figure (3)
plot(r1,Vr1(:,1))
hold on
plot(r_R_Exp,Exp_Vr,'-ro')
xlim([0.35 0.95])
xlabel('r/R')
ylabel('Vr(m/s)')
grid on
legend('BEM','Experimental','Location','SouthEast')

figure (4)
plot(r1,a1_tip(:,1), 'g')
hold on
plot(r1,a1(:,1))
hold on
plot(r_R_Exp_a,Exp_a,'-ro')
xlim([0.4 1])
ylim([-0.5 0])
xlabel('r/R')
```

```

ylabel('a')
grid on
legend('BEM','BEM + tip
loss','Experimental','Location','NorthEast')

figure (5)
plot(r1,C1(:,1))
hold on
plot(r_R_Exp,Exp_C1,'-ro')
xlim([0.35 0.95])
ylim([0 1])
xlabel('r/R')
ylabel('C1')
grid on
legend('BEM','Experimental','Location','NorthEast')

figure (6)
plot(r1,delta_T1(:,1))
hold on
plot(r1,delta_T1_extr(:,1),'g')
hold on
plot(r_R_Exp_deltaT,Exp_deltaT,'-ro')
xlim([0.3 1])
xlabel('r/R')
ylabel('dT (N/m)')
grid on
legend('BEM','BEM +
extrapolacion','Experimental','Location','SouthEast')

figure (7)
plot(r1,delta_Q1(:,1))
hold on
plot(r1,delta_Q1_extr(:,1),'g')
hold on
plot(r_R_Exp_deltaQ,Exp_deltaQ,'-ro')
xlim([0.3 1])
xlabel('r/R')
ylabel('dQ (Nm/m)')
grid on
legend('BEM','BEM +
extrapolacion','Experimental','Location','SouthEast')

```

### 1.2.10. Draw\_Yaw\_function.m

```

function Draw_Yaw_function(pos,alfa,Vr,C1_t,a,yaw,el_b,r_R,delta_T,delta_Q)

%Experimental data
Posicion=[0 30 60 90 120 150 180 210 240 270 300 330];

if yaw==30
    alfa_e_4=[9.1 8.7 7.9 6.6 5.8 4.6 ...
        4.1 4.8 5.7 6.2 7.2 7.9]; %r/R=0.4
    alfa_e_6=[4.5 4.1 3.8 3.2 3 3 3.1 ...
        3.4 3.8 4.2 4.7 4.9]; %r/R=0.6
    alfa_e_8=[3.4 3.1 2.7 2.6 2.5 2.4 ...

```

## Código BEM

```
    2.6 2.9 3.2 3.6 3.7 3.7]; %r/R=0.8
Cl_e_4=[0.69 0.61 0.58 0.5 0.42 0.36 ...
0.3 0.31 0.35 0.4 0.47 0.44]; %r/R=0.4
Cl_e_6=[0.48 0.44 0.41 0.36 0.33 0.3 ...
0.27 0.26 0.29 0.32 0.37 0.41]; %r/R=0.6
Cl_e_8=[0.36 0.34 0.32 0.28 0.27 0.25 ...
0.24 0.25 0.28 0.32 0.33 0.35]; %r/R=0.8
a_e_4=[-0.23 -0.23 -0.24 -0.234 -0.23 ...
-0.23 -0.22 -0.22 -0.22 -0.23 -0.23 -0.23]; %r/R=0.4
a_e_6=[-0.254 -0.254 -0.254 -0.254 -0.254 ...
-0.254 -0.254 -0.254 -0.254 -0.254 ...
-0.254 -0.254]; %r/R=0.6
a_e_8=[-0.27 -0.265 -0.265 -0.265 -0.27 ...
-0.274 -0.27 -0.27 -0.27 -0.27 -0.274 -0.274]; %r/R=0.8
deltaT_e_4=[7.8 7.5 7.8 7.9 7.9 7 6.1 ...
6.1 6.1 6.15 6.1 6.1]; %r/R=0.4
deltaT_e_6=[13.1 13 13 13 13.1 12.1 11.8 ...
11 11.3 11.8 11.8 12]; %r/R=0.6
deltaT_e_8=[19 18 17 17.5 18 17 16.1 16.8 ...
18 19.2 18.7 18.7]; %r/R=0.8
deltaQ_e_4=[0.4 0.36 0.36 0.35 0.31 0.22 ...
0.18 0.18 0.19 0.24 0.26 0.29]; %r/R=0.4
deltaQ_e_6=[0.59 0.52 0.5 0.47 0.43 0.32 ...
0.24 0.21 0.3 0.39 0.415 0.475]; %r/R=0.6
deltaQ_e_8=[0.71 0.64 0.51 0.425 0.38 0.25 ...
0.21 0.26 0.46 0.62 0.62 0.69]; %r/R=0.8
elseif yaw==45
alfa_e_4=[7.5 7.6 7.3 5.6 4.1 3 2 1 1.2 ...
2.8 4 5.2]; %r/R=0.4
alfa_e_6=[3.7 3.5 3.1 2.6 2.2 1.7 1.4 1 ...
1.3 1.7 2.3 3.7]; %r/R=0.6
alfa_e_8=[2.8 2.6 2.2 2 1.8 1.7 1.6 1.5 ...
2 2.4 2.5 2.5]; %r/R=0.8
Cl_e_4=[0.6 0.5 0.47 0.41 0.32 0.25 0.19 ...
0.15 0.17 0.21 0.3 0.39]; %r/R=0.4
Cl_e_6=[0.39 0.36 0.34 0.29 0.25 0.2 0.16 ...
0.14 0.15 0.19 0.24 0.29]; %r/R=0.6
Cl_e_8=[0.31 0.27 0.25 0.22 0.2 0.19 0.19 ...
0.22 0.25 0.29 0.32 0.33]; %r/R=0.8
a_e_4=[-0.17 -0.168 -0.168 -0.18 -0.19 ...
-0.18 -0.178 -0.178 -0.178 -0.183 -0.19 -0.19]; %r/R=0.4
a_e_6=[-0.186 -0.186 -0.186 -0.186 -0.186 ...
-0.186 -0.186 -0.186 -0.186 -0.186 ...
-0.186 -0.186]; %r/R=0.6
a_e_8=[-0.18 -0.18 -0.18 -0.19 -0.21 -0.21 ...
-0.19 -0.18 -0.18 -0.19 -0.21 -0.21]; %r/R=0.8
deltaT_e_4=[5.9 5.1 6 6.6 6.2 5.5 4.1 3.1 ...
3.1 3.6 3.8 3.9]; %r/R=0.4
deltaT_e_6=[10 9.8 10 10 9.1 7.9 6 5.9 ...
6.1 6.7 7.3 7.9]; %r/R=0.6
deltaT_e_8=[15.1 14 13.8 13.9 14 14.1 ...
13.9 13 14.3 15.9 14.9 13.9]; %r/R=0.8
deltaQ_e_4=[0.265 0.24 0.265 0.26 0.2 ...
0.14 0.05 -0.01 0.03 0.07 0.12 0.14]; %r/R=0.4
deltaQ_e_6=[0.38 0.35 0.32 0.27 0.18 0.16 ...
0.08 -0.1 0.02 0.1 0.12 0.21]; %r/R=0.6
deltaQ_e_8=[0.45 0.3 0.19 0.15 0.15 0.15 ...
0.14 0.11 0.13 0.27 0.3 0.28]; %r/R=0.8
end
```

```
%Alfa
figure (1)
titulo=sprintf('Yaw = %g grados ; r/R= %g', yaw, r_R(2));
plot(pos(2,:),alfa(2,:))
hold on
plot(Posicion,alfa_e_4,'-ro')
ylim([0 12])
xlim([0 330])
legend('BEM','Experimental','Location','SouthEast')
xlabel('Posición (grados)')
ylabel('Alfa (grados)')
title(titulo,'FontSize',16)
grid on

figure (2)
titulo=sprintf('Yaw = %g grados ; r/R= %g', yaw, r_R(4));
plot(pos(4,:),alfa(4,:))
hold on
plot(Posicion,alfa_e_6,'-ro')
ylim([0 6])
xlim([0 330])
legend('BEM','Experimental','Location','SouthEast')
xlabel('Posición (grados)')
ylabel('Alfa (grados)')
title(titulo,'FontSize',16)
grid on

figure (3)
titulo=sprintf('Yaw = %g grados ; r/R= %g', yaw, r_R(6));
plot(pos(6,:),alfa(6,:))
hold on
plot(Posicion,alfa_e_8,'-ro')
ylim([0 5])
xlim([0 330])
legend('BEM','Experimental','Location','SouthEast')
xlabel('Posición (grados)')
ylabel('Alfa (grados)')
title(titulo,'FontSize',16)
grid on

%Cl
figure (4)
titulo=sprintf('Yaw = %g grados ; r/R= %g', yaw, r_R(2));
plot(pos(2,:),Cl_t(2,:))
hold on
plot(Posicion,Cl_e_4,'-ro')
xlim([0 330])
ylim([0 1])
legend('BEM','Experimental','Location','SouthEast')
xlabel('Posición (grados)')
ylabel('Cl')
title(titulo,'FontSize',16)
grid on

figure (5)
titulo=sprintf('Yaw = %g grados ; r/R= %g', yaw, r_R(4));
plot(pos(4,:),Cl_t(4,:))
hold on
```

## Código BEM

```
plot(Posicion,Cl_e_6,'-ro')
xlim([0 330])
ylim([0 0.6])
legend('BEM','Experimental','Location','SouthEast')
xlabel('Posición (grados)')
ylabel('Cl')
title(titulo,'FontSize',16)
grid on

figure (6)
titulo=sprintf('Yaw = %g grados ; r/R= %g', yaw, r_R(6));
plot(pos(6,:),Cl_t(6,:))
hold on
plot(Posicion,Cl_e_8,'-ro')
xlim([0 330])
ylim([0 0.5])
legend('BEM','Experimental','Location','SouthEast')
xlabel('Posición (grados)')
ylabel('Cl')
title(titulo,'FontSize',16)
grid on

%a
figure (7)
titulo=sprintf('Yaw = %g grados ; r/R= %g', yaw, r_R(2));
plot(pos(2,:),a(2,:))
hold on
plot(Posicion,a_e_4,'-ro')
xlim([0 330])
ylim([-0.3 0])
legend('BEM','Experimental','Location','NorthEast')
xlabel('Posición (grados)')
ylabel('a')
title(titulo,'FontSize',16)
grid on

figure (8)
titulo=sprintf('Yaw = %g grados ; r/R= %g', yaw, r_R(4));
plot(pos(4,:),a(4,:))
hold on
plot(Posicion,a_e_6,'-ro')
xlim([0 330])
ylim([-0.35 0])
legend('BEM','Experimental','Location','NorthEast')
xlabel('Posición (grados)')
ylabel('a')
title(titulo,'FontSize',16)
grid on

figure (9)
titulo=sprintf('Yaw = %g grados ; r/R= %g', yaw, r_R(6));
plot(pos(6,:),a(6,:))
hold on
plot(Posicion,a_e_8,'-ro')
xlim([0 330])
ylim([-0.35 0])
legend('BEM','Experimental','Location','NorthEast')
xlabel('Posición (grados)')
```

```
ylabel('a')
title(titulo,'FontSize',16)
grid on

%dT
figure (10)
titulo=sprintf('Yaw = %g grados ; r/R= %g', yaw, r_R(2));
plot(pos(2,:),delta_T(2,:))
hold on
plot(Posicion,deltaT_e_4,'-ro')
xlim([0 330])
ylim([0 12])
legend('BEM','Experimental','Location','SouthEast')
xlabel('Posición (grados)')
ylabel('dT (N/m)')
title(titulo,'FontSize',16)
grid on

figure (11)
titulo=sprintf('Yaw = %g grados ; r/R= %g', yaw, r_R(4));
plot(pos(4,:),delta_T(4,:))
hold on
plot(Posicion,deltaT_e_6,'-ro')
xlim([0 330])
ylim([0 18])
legend('BEM','Experimental','Location','SouthEast')
xlabel('Posición (grados)')
ylabel('dT (N/m)')
title(titulo,'FontSize',16)
grid on

figure (12)
titulo=sprintf('Yaw = %g grados ; r/R= %g', yaw, r_R(6));
plot(pos(6,:),delta_T(6,:))
hold on
plot(Posicion,deltaT_e_8,'-ro')
xlim([0 330])
ylim([0 25])
legend('BEM','Experimental','Location','SouthEast')
xlabel('Posición (grados)')
ylabel('dT (N/m)')
title(titulo,'FontSize',16)
grid on

%dQ
figure (13)
titulo=sprintf('Yaw = %g grados ; r/R= %g', yaw, r_R(2));
plot(pos(2,:),delta_Q(2,:))
hold on
plot(Posicion,deltaQ_e_4,'-ro')
xlim([0 330])
ylim([0 0.6])
legend('BEM','Experimental','Location','NorthEast')
xlabel('Posición (grados)')
ylabel('dQ (Nm/m)')
title(titulo,'FontSize',16)
grid on
```

## Código BEM

```
figure (14)
titulo=sprintf('Yaw = %g grados ; r/R= %g', yaw, r_R(4));
plot(pos(4,:),delta_Q(4,:))
hold on
plot(Posicion,deltaQ_e_6,'-ro')
xlim([0 330])
ylim([0 0.9])
legend('BEM','Experimental','Location','NorthEast')
xlabel('Posición (grados)')
ylabel('dQ (Nm/m)')
title(titulo,'FontSize',16)
grid on

figure (15)
titulo=sprintf('Yaw = %g grados ; r/R= %g', yaw, r_R(6));
plot(pos(6,:),delta_Q(6,:))
hold on
plot(Posicion,deltaQ_e_8,'-ro')
xlim([0 330])
ylim([0 1])
legend('BEM','Experimental','Location','NorthEast')
xlabel('Posición (grados)')
ylabel('dQ (Nm/m)')
title(titulo,'FontSize',16)
grid on

%Graficos generales
titulo=sprintf('Yaw = %g grados', yaw);

figure (16)
hold all
x=1;
for i=3:el_b
    plot(pos(i,:),alfa(i,:),'color',rand(1,3))
    str=sprintf('r/R= %g',r_R(i));
    Leg{x}=str;
    x=x+1;
end

legend(Leg,'Location','NorthEast')
xlabel('Posición (grados)')
ylabel('Alfa (grados)')
title(titulo,'FontSize',16)
grid on

figure (17)
hold all
x=1;
for i=3:el_b
    plot(pos(i,:),Vr(i,:),'color',rand(1,3))
    str=sprintf('r/R= %g',r_R(i));
    Leg{x}=str;
    x=x+1;
end

legend(Leg,'Location','Best')
xlabel('Posición (grados)')
ylabel('Vr(m/s)')
```

```
title(titulo,'FontSize',16)
grid on

figure (18)
hold all
x=1;
for i=3:el_b
    plot(pos(i,:),Cl_t(i,:),'color',rand(1,3))
    str=sprintf('r/R= %g',r_R(i));
    Leg{x}=str;
    x=x+1;
end

legend(Leg,'Location','NorthEast')
xlabel('Posición (grados)')
ylabel('Cl')
title(titulo,'FontSize',16)
grid on

figure (19)
hold all
x=1;
for i=3:el_b
    plot(pos(i,:),a(i,:),'color',rand(1,3))
    str=sprintf('r/R= %g',r_R(i));
    Leg{x}=str;
    x=x+1;
end

legend(Leg,'Location','SouthWest')
xlabel('Posición (grados)')
ylabel('a ')
title(titulo,'FontSize',16)
grid on

figure (20)
hold all
x=1;
for i=3:el_b
    plot(pos(i,:),delta_T(i,:),'color',rand(1,3))
    str=sprintf('r/R= %g',r_R(i));
    Leg{x}=str;
    x=x+1;
end

legend(Leg,'Location','Best')
xlabel('Posición (grados)')
ylabel('dT (N/m) ')
title(titulo,'FontSize',16)
grid on

figure (21)
hold all
x=1;
for i=3:el_b
    plot(pos(i,:),delta_Q(i,:),'color',rand(1,3))
    str=sprintf('r/R= %g',r_R(i));
```

## Código BEM

---

```
Leg{x}=str;
x=x+1;
end

legend(Leg,'Location','NorthEast')
xlabel('Posición (grados)')
ylabel('dQ (Nm/m) ')
title(titulo,'FontSize',16)
grid on
```

## Validación Beddoes-Leishman

```

close all; clear all; clc;

[archivo, ruta] = uigetfile('.txt', 'Seleccione un archivo de texto');
[Alpha,C1,Cd]=Input_function(archivo, ruta);

alfa_0=Alpha(abs(C1-0)==min(abs(C1-0)))*pi/180;
Cd0=Cd(abs(Alpha-0)==min(abs(Alpha-0)));

for i=1:1:length(Alpha)
    Cn(i)=C1(i)*cos(Alpha(i)*pi/180)+Cd(i)*sin(Alpha(i)*pi/180);
    Cc(i)=C1(i)*sin(Alpha(i)*pi/180)-Cd(i)*cos(Alpha(i)*pi/180);
end
Cn_alfa=((Cn(7)-Cn(1))/((Alpha(7)-Alpha(1)))*180/pi; %1/rad

for i=1:1:length(Alpha)
    f(i)=(2*(sqrt(Cn(i))/(Cn_alfa*((Alpha(i)-
    alfa_0*180/pi)*pi/180)))-1)^2;
    if f(i)>1 || isnan(f(i))
        f(i)=1;
    elseif f(i)<0
        f(i)=0;
    end
    Cn_1(i)=(((sqrt(f(i))+1)/2)^2)*(Cn_alfa*((Alpha(i)-alfa_0*...
    180/pi)*pi/180));
end

Re=1*10^6;
delta_t=0.01;
k=0.077;
c=0.457;
a=340;
M=0.1;

V_rel=M*a;
omega=(2*V_rel*k)/c;
delta_s=(V_rel*delta_t)/(c/2);
T=(2*pi)/omega;

t=0:delta_t:2*(T+delta_t);

A0=8;
A=10;
for i=1:1:length(t)
    alfa(i)=(A0+A*sin(omega*t(i)))*pi/180;
    Cn_t(i)=interp1(Alpha(1,:),Cn(1,:),alfa(i)*180/pi,'linear','extrap');
    Cn_t(i)=interp1(Alpha(1,:),Cc(1,:),alfa(i)*180/pi,'linear','extrap');
    Cn_1t(i)=interp1(Alpha(1,:),Cn_1(1,:),alfa(i)*180/pi,',...
    'linear','extrap');
end

beta=sqrt(1-M^2);

```

```
%Condiciones iniciales
X(1)=0;
Y(1)=0;
D(1)=0;
Dp(1)=0;
Df(1)=0;
Cn_c(1)=0;
f1(1)=1;
Cn_v(1)=0;
cv(1)=0;
delta_alfa(1)=0;

%Parametros experimentales
%Flujo adherido
A1=0.3;
A2=0.7;
b1=0.14;
b2=0.53;
K_alfa=0.75;
%Flujo desprendido
Tp=1.5;
Tf=5;
Tv=6;
Ti=c/a;
St=0.19;
etha=0.95;

for j=2:1:length(alfa)
    Tvl=5;
    delta_alfa(j)=alfa(j)-alfa(j-1);

    %Modulo 1:

    %Componente circulatoria
    X(j)=X(j-1)*exp(-b1*beta^2*delta_s)+A1*delta_alfa(j)*...
        exp(-b1*beta^2*delta_s/2);
    Y(j)=Y(j-1)*exp(-b2*beta^2*delta_s)+A2*delta_alfa(j)*...
        exp(-b2*beta^2*delta_s/2);
    alfa_E(j)=alfa(j)-X(j)-Y(j);
    Cn_c(j)=Cn_alfa*(alfa_E(j)-alfa_0);
    %Componente impulsiva
    D(j)=D(j-1)*exp(-delta_t/(K_alfa*Ti))+((delta_alfa(j)-...
        delta_alfa(j-1))/delta_t)*exp(-delta_t/(2*K_alfa*Ti));
    Cn_i(j)=((4*K_alfa*c)/V_rel)*(delta_alfa(j)/delta_t-D(j));
    %Total
    Cn_p(j)=Cn_i(j)+Cn_c(j);

    %Modulo 2:

    %Presion
    Dp(j)=Dp(j-1)*exp(-delta_s/Tp)+(Cn_c(j)-Cn_c(j-1))*...
        exp(-delta_s/(2*Tp));
    dCn(j)=Cn_c(j)-Dp(j);
    alfa_f(j)=dCn(j)/Cn_alfa;
    f1(j)=interp1(Alpha,f,alfa_f(j)*180/pi,'linear','extrap');
    %Viscosidad
    Df(j)=Df(j-1)*exp(-delta_s/Tf)+(f1(j)-f1(j-1))*...
        exp(-delta_s/(2*Tf));
```

```

f2(j)=f1(j)-Df(j);
Cn_f(j)=Cn_alfa*((1+sqrt(f2(j)))/2)^2*...
(alfa_E(j)-alfa_0)+Cn_i(j);

%Modulo 3:

cv(j)=Cn_c(j)-Cn_f(j);
tau_v=Tvl+(2*sqrt(1-f2(j)))/St;

if (tau_v<Tvl)
    TV=Tv;
else
    TV=Tv/2;
end

Cn_v(j)=Cn_v(j-1)*exp(-delta_s/TV)+...
(cv(j)-cv(j-1))*exp(-delta_s/(2*TV));

%Total (Ejes cuerpo)
Cn_u(j)=Cn_f(j)+Cn_v(j);
Cc_u(j)=etha*Cn_alfa*(alfa_E(j)-alfa_0)^2*sqrt(f2(j));

%Total (Ejes viento)
Cl_u(j)=Cn_u(j)*cos(alfa(j))+Cc_u(j)*sin(alfa(j));
Cd_u(j)=Cn_u(j)*sin(alfa(j))-Cc_u(j)*cos(alfa(j))+Cd0;
end

alfa(1:round(length(t)/2))=[ ];
Cn_v(1:round(length(t)/2))=[ ];
Cn_p(1:round(length(t)/2))=[ ];
Cl_u(1:round(length(t)/2))=[ ];
Cn_f(1:round(length(t)/2))=[ ];
Cd_u(1:round(length(t)/2))=[ ];
str=sprintf...
('Alfa = %g + %gsin (%g t) \n c= %g ; k = %g ;... M= %g ; Re= %g',...
,A0,A,omega,c, k, M, Re);

figure(1)
plot(Alpha,Cl,'m')
hold on
plot(alfa*180/pi,Cn_v,'k')
hold on
plot(alfa*180/pi,Cn_p,'g')
hold on
plot(alfa*180/pi,Cn_f,'r')
grid on

title(str)
xlabel('Alfa');
ylabel('Cn');
legend('Estacionario','Vortex','Flujo adherido',...
'Flujo no adherido','Location','SouthEast')

```

```

figure (2)
plot(alfa*180/pi,Cl_u,'r')
hold on
plot(Alpha,Cl,'m')
title(str)
xlabel('Alfa');
ylabel('Cl')
grid on

%Experimental
%Exp=1; k=0.026 ; A0=8; A=5.5;
%Exp=2; k=0.026 ; A0=14; A=5.5;
%Exp=3; k=0.077 ; A0=8; A=10;
%Exp=4; k=0.077 ; A0=14; A=10;
Exp=3;

if Exp==1
    Alpha_e=[3 4 5 6 7 8 9 10 11 12 13 13.5 13 ...
              12 11 10 9 8 7 6 5 4 3];
    Cl_e=[0.51 0.6 0.7 0.76 0.9 1 1.05 1.1 1.13...
              1.1 1.03 1 1.03 0.9 0.87 0.84 0.8 0.74 0.72 0.76 0.65 0.52
0.48];
    hold on
    plot(Alpha_e,Cl_e,'--')
    hold on
    plot (Alpha_e,Cl_e,'o')
elseif Exp==2
    Alpha_e=[9.5 10 11 12 13 14 15 16 17 18 19 ...
              19.5 19 18 17 16 15 14 13 12 11 10 9.5];
    Cl_e=[0.9 0.99 1.1 1.11 1.1 1.12 1.13 1.12 ...
              1.1 1.05 0.95 0.76 0.8 0.85 0.87 0.87 0.88 ...
              0.89 0.9 0.91 0.9 0.89 0.9];
    hold on
    plot(Alpha_e,Cl_e,'--')
    hold on
    plot (Alpha_e,Cl_e,'o')
    axis([8 20 0.5 1.5])
elseif Exp==3
    Alpha_e=[-2.5 -2 -1 0 2 4 6 8 10 12 14 15 16 ...
              18 18 16 14 12 10 8 6 4 2 0 -1 -2 -2.5];
    Cl_e=[-0.2 -0.1 0.1 0.15 0.35 0.45 0.6 1 1.3 ...
              1.45 1.47 1.55 1.45 1.2 0.9 0.7 0.63 0.6 ...
              0.59 0.51 0.49 0.45 0.15 0 -0.1 -0.19 -0.2];
    hold on
    plot(Alpha_e,Cl_e,'--')
    hold on
    plot (Alpha_e,Cl_e,'o')
elseif Exp==4
    Alpha_e=[2 4 6 8 10 12 14 16 18 20 22 23 23 ...
              23 22 20 18 16 15 14 12 10 8 6 4 2];
    Cl_e=[0.4 0.53 0.7 0.9 1.13 1.33 1.5 1.67 ...
              1.6 1.57 1.2 1.22 1.82 0.9 0.7 0.6 0.53 ...
              0.48 0.4 0.51 0.7 0.72 0.62 0.6 0.52 0.4];
    hold on
    plot(Alpha_e,Cl_e,'--')
    hold on
    plot (Alpha_e,Cl_e,'o')
end

```

```

legend('Modelo','Estacionario','Experimental','Location','SouthEast')

figure (3)

plot(alfa*180/pi,Cd_u,'r')
grid on
hold on
plot(Alpha,Cd,'m')

if Exp==1
    Alpha_e=[3 4 5 6 7 8 9 10 11 12 13 13.5 ...
              13 12 11 10 9 8 7 6 5 4 3];
    Cd_e=[0 0.002 0.004 0.006 0.008 0.009 ...
              0.01 0.016 0.02 0.031 0.04 0.046 ...
              0.05 0.04 0.025 0.022 0.017 0.015 0.013 0.012 0.01 0.009
0];
    hold on
    plot(Alpha_e,Cd_e,'--')
    hold on
    plot (Alpha_e,Cd_e,'o')
    axis([2 14 -0.1 0.3])
elseif Exp==2
    Alpha_e=[9.5 10 11 12 13 14 15 16 17 ...
              18 19 19.5 19 18 17 16 15 14 13 12 11 10 9.5];
    Cd_e=[0.002 0.01 0.02 0.017 0.037 0.05...
              0.06 0.07 0.1 0.12 0.14 0.144 0.142 ...
              0.13 0.11 0.09 0.07 0.057 0.047 0.037 0.032 0.027 0];
    hold on
    plot(Alpha_e,Cd_e,'--')
    hold on
    plot (Alpha_e,Cd_e,'o')
    axis([8 20 -0.1 0.3])
elseif Exp==3
    Alpha_e=[-2.5 -2 -1 0 2 4 6 8 10 12 14 ...
              15 16 18 19.5 19.5 19.5 18 16 14 12 10 8 6 4 2 0 -1 -2 -
2.5 ];
    Cd_e=[-0.003 -0.003 0 0.001 0.01 0.02...
              0.021 0.04 0.05 0.06 0.073 0.09 ...
              0.11 0.14 0.142 0.23 0.14 0.12 0.077...
              0.042 0.028 0.02 0.007 -0.003 0.001 ...
              0.013 0.0014 0.001 -0.003 -0.003 ];
    hold on
    plot(Alpha_e,Cd_e,'--')
    hold on
    plot (Alpha_e,Cd_e,'o')
    axis([-5 20 -0.1 0.3])
end

legend('Modelo','Estacionario','Experimental','Location','NorthWest')
title(str)
xlabel('Alfa');
ylabel('Cd')
grid o

```

## Validación Snel\_Viterna

```
clear all; clc;
[archivo, ruta] = uigetfile('.txt', 'S809_Xfoil_500000.txt');
[Alpha,C1,Cd,Cm]=Input_airfoil_function(archivo, ruta);

[archivo, ruta] = uigetfile('.txt',
'S809_Experimental_500000.txt');
[Alpha_e,C1_e,Cd_e]=Input_airfoil_experimental_function(archivo, ruta);

[archivo, ruta] = uigetfile('.txt', 'S809_NREL_63.txt');
[Alpha_63,C1_63,Cd_63]=Input_airfoil_experimental_function(archivo, ruta);

[archivo, ruta] = uigetfile('.txt', 'S809_NREL_30.txt');
[Alpha_30,C1_30,Cd_30]=Input_airfoil_experimental_function(archivo, ruta);

pos_CLs=find(C1==max(C1)); %Posicion
CL=CL_max=CL_stall
pos_Alpha0=find(Alpha==min(abs(Alpha-0))); %Posición Alpha = 0
pos_CL0= C1==min(abs(C1)); %Posición CL = 0

Alpha_s=Alpha(pos_CLs)*pi/180; %Alpha=Alpha_stall
Alpha_0=Alpha(pos_CL0)*pi/180; %Alpha=Alpha_CL0
Alpha_inicial=Alpha(1)*pi/180;
Alpha_final=Alpha(length(Alpha))*pi/180;
C1_s=max(C1); %CL=CL_stall
Cd_s=Cd(pos_CLs); %CD=CD_stall
Cm_s=Cm(pos_CLs); %CM=CM_stall
Cm0=Cm(pos_CL0); %CM = 0
C1_alpha=((C1(pos_Alpha0+1)-C1(pos_Alpha0))/...
(Alpha(pos_Alpha0+1)-Alpha(pos_Alpha0)))*180/pi;

%% Viterna_Corrigan extrapolation (CL & CD)
AR=11;

if AR<=50
    Cd_max=1.11+0.018*AR;
else
    Cd_max=2.01;
end

Kl=(C1_s-
Cd_max*sin(Alpha_s)*cos(Alpha_s))*(sin(Alpha_s)/cos(Alpha_s)^2);
Kd=(Cd_s-Cd_max*sin(Alpha_s)^2)/cos(Alpha_s);

%Alfa_s<Alfa<90
alfa=(Alpha_final:pi/180:pi/2);
for i=1:1:length(alfa)
    C1_1(i)=(Cd_max*sin(alfa(i))*...
    cos(alfa(i)))+Kl*(cos(alfa(i))^2/sin(alfa(i))));;
    Cd_1(i)=Cd_max*sin(alfa(i))^2+Kd*cos(alfa(i));
end
```

```

Cl=[Cl Cl_1];
Cd=[Cd Cd_1];
alfa=[Alpha*pi/180 alfa]*180/pi;

%% Snel et al & Lindenburg

%R=5.029 m
%63%R -- c=0.542 ; r=3.185 ; theta=5.89°
%30%R -- c=0.711 ; r=1.51 ; theta=19.08°
R=5.029;
R80=80/100*R;
c=0.711;
r=1.51;
theta=19.08;
c_r=c/r;
a=3.1;
b=2;

k=cos((theta+30)*pi/180)^2;
pos_alfa30=find(abs((round(alfa)-30))==...
    min(abs(round(alfa)-30)));
delta_Cl(pos_alfa30)=(Cl_alpha*...
    sin(alfa(pos_alfa30)*pi/180-Alpha_0))-Cl(pos_alfa30);
Cl_lim=a*(c_r)^b*k*delta_Cl(pos_alfa30);

for i=1:1:length(alfa)
    if alfa(i)<=30
        k=cos((theta+alfa(i))*pi/180)^2;
        delta_Cl(i)=(Cl_alpha*...
            sin(alfa(i)*pi/180-Alpha_0))-Cl(i);
        Cl_3d(i)=Cl(i)+a*k*(c_r)^b*delta_Cl(i);
        Alpha_2(i)=alfa(i);
    elseif alfa(i)<=60
        Cl_3d(i)=Cl(i)+(Cl_lim-Cl_lim/30*(alfa(i)-30));
        Alpha_2(i)=alfa(i);
    else
        Cl_3d(i)=Cl(i);
        Alpha_2(i)=alfa(i);
    end
end

figure(1)
plot(alfa,Cl)
hold on
plot(Alpha_e,Cl_e, 'r')
hold on
plot(Alpha_2,Cl_3d, 'm')
hold on

if r==3.185
    plot(Alpha_63,Cl_63, 'g')
else
    plot(Alpha_30,Cl_30, 'g')
end

grid on
legend('Xfoil + Viterna (2D)', 'CSU (2D)', ...

```

## Validación Snel Viterna

---

```
'Snel et al(3D)', 'NREL (3D)', 'Location', 'SouthEast')
str=sprintf('c/r = %g ; Theta =%g grados',c_r,theta);
title(str)
xlabel('Alfa')
ylabel('Cl')

figure(2)
plot(alfa,Cd)
hold on
plot(Alpha_e,Cd_e, 'r')
hold on

if r==3.185
    plot(Alpha_63,Cd_63, 'g')
else
    plot(Alpha_30,Cd_30, 'g')
end

grid on
legend('Xfoil + Viterna (2D) ','CSU (2D)',...
    'NREL (3D)', 'Location', 'SouthEast')
title ('Re = 500.000')
xlabel('Alfa')
ylabel('Cdp')
```

## Archivos.txt

### 1.3. Rotor\_geometry.txt

**Tabla 1** Turbina de TU Delft [1]

x(m)	chord	twist(°)
0.18	0.08	5.99
0.24	0.08	5.33
0.30	0.08	4.66
0.36	0.08	3.99
0.42	0.08	3.31
0.48	0.08	2.66
0.54	0.08	1.99
0.60	0.08	2.00

### 1.1. NACA 0012 \_360000.txt

**Tabla 2** NACA 0012; Re = 360.000 [2]

Alpha	CL	CD	CM
0	0	0.008589	1.35E-05
5	0.519895	0.015782	-0.00471
10	1.000683	0.02755	-0.01002
15	0.657947	0.132609	-0.05835
20	0.465796	0.15698	0.061568
30	0.927874	0.576852	0.177157
50	1.157604	1.190874	0.411402
90	0.027871	1.927949	0.518723
130	-0.97926	1.345253	0.651483
160	-0.53523	0.172799	0.18316
170	-0.84022	0.127649	0.454261
180	-3.3E-06	0.000844	7.19E-07

### 1.2. S809\_1000000.txt

**Tabla 3** NREL S809 Airfoil Data from OSU; Re = 1,000,000 [3]

Alpha	C <sub>l</sub>	C <sub>d</sub>
-4.1	-0.4	0.0004
-2.1	-0.16	0.0009

-0.6	0	0.0018
0.0	0.07	0.0022
2.1	0.30	0.0037
4.1	0.55	0.0050
6.1	0.79	0.0063
8.2	0.90	0.0096
10.1	0.94	0.0231
11.2	0.93	0.0236
12.2	0.97	0.0368
13.3	1.00	0.0551
14.2	1.02	0.0618
15.2	1.03	0.0705
16.2	1.01	0.0880
17.2	0.95	0.1043
18.1	0.90	0.1325
19.2	0.78	0.3474
20.0	0.67	0.3211
22.1	0.70	0.3699
24.0	0.77	0.4348
26.1	0.91	0.5356
28.1	0.99	0.6192
30.0	1.06	0.7025
32.1	1.11	0.7916
34.1	1.18	0.8963
36.0	1.21	0.9737
38.1	1.24	1.0627
39.9	1.26	1.1509

### 1.3. S809\_Xfoil\_500000.txt

**Tabla 4** S809 Xfoil; Re = 500.000 [4]

Alpha	CL	CD	CDp	CM	Top_Xtr	Bot_Xtr
-18.000	-0.3992	0.16006	0.15751	-0.0156	0.7720	0.0155
-17.750	-0.4079	0.15215	0.14959	-0.0203	0.7665	0.0161
-11.500	-0.6189	0.04377	0.03655	-0.0617	0.6633	0.0097
-11.250	-0.5860	0.04451	0.03728	-0.0603	0.6604	0.0101
-11.000	-0.5621	0.04466	0.03741	-0.0591	0.6580	0.0102
-10.750	-0.5447	0.04405	0.03677	-0.0587	0.6559	0.0105
-10.500	-0.5268	0.04369	0.03639	-0.0577	0.6537	0.0106
-10.250	-0.5115	0.04291	0.03560	-0.0573	0.6518	0.0112
-10.000	-0.4961	0.04213	0.03478	-0.0569	0.6500	0.0117
-9.750	-0.4845	0.04130	0.03392	-0.0557	0.6483	0.0118
-9.500	-0.4724	0.04032	0.03288	-0.0551	0.6467	0.0122
-9.250	-0.4626	0.03919	0.03171	-0.0543	0.6452	0.0121

-9.000	-0.4523	0.03806	0.03053	-0.0537	0.6436	0.0124
-8.750	-0.4464	0.03661	0.02905	-0.0528	0.6425	0.0124
-8.500	-0.4437	0.03492	0.02734	-0.0517	0.6413	0.0129
-8.250	-0.4412	0.03328	0.02568	-0.0506	0.6400	0.0134
-8.000	-0.4363	0.03183	0.02422	-0.0495	0.6387	0.0149
-7.750	-0.4298	0.03055	0.02290	-0.0485	0.6376	0.0150
-7.500	-0.4241	0.02926	0.02156	-0.0474	0.6365	0.0160
-7.250	-0.4247	0.02760	0.01988	-0.0458	0.6354	0.0184
-7.000	-0.4257	0.02592	0.01819	-0.0441	0.6344	0.0240
-6.750	-0.4352	0.02378	0.01612	-0.0417	0.6334	0.0433
-6.500	-0.4597	0.02075	0.01327	-0.0383	0.6325	0.0911
-6.250	-0.4872	0.01788	0.01051	-0.0341	0.6316	0.1316
-6.000	-0.5147	0.01444	0.00733	-0.0314	0.6307	0.2250
-5.750	-0.5255	0.01121	0.00508	-0.0294	0.6298	0.4604
-5.500	-0.4954	0.01172	0.00554	-0.0300	0.6288	0.4870
-5.000	-0.4319	0.01373	0.00765	-0.0307	0.6271	0.5106
-4.750	-0.4013	0.01452	0.00838	-0.0311	0.6265	0.5184
-4.500	-0.3701	0.01496	0.00890	-0.0315	0.6258	0.5212
-4.250	-0.3414	0.01485	0.00874	-0.0321	0.6251	0.5221
-4.000	-0.3126	0.01475	0.00859	-0.0327	0.6244	0.5230
-3.750	-0.2838	0.01465	0.00844	-0.0333	0.6237	0.5239
-3.500	-0.2550	0.01455	0.00830	-0.0340	0.6231	0.5250
-3.250	-0.2262	0.01445	0.00815	-0.0346	0.6222	0.5261
-3.000	-0.1974	0.01435	0.00800	-0.0353	0.6213	0.5272
-2.750	-0.1685	0.01426	0.00787	-0.0359	0.6205	0.5285
-2.500	-0.1395	0.01419	0.00775	-0.0366	0.6198	0.5299
-2.250	-0.1105	0.01414	0.00764	-0.0373	0.6192	0.5311
-2.000	-0.0814	0.01409	0.00754	-0.0380	0.6185	0.5321
-1.750	-0.0524	0.01399	0.00739	-0.0387	0.6179	0.5331
-1.500	-0.0235	0.01383	0.00724	-0.0394	0.6173	0.5341
-1.250	0.0056	0.01377	0.00719	-0.0400	0.6168	0.5350
-1.000	0.0348	0.01376	0.00718	-0.0407	0.6162	0.5359
-0.750	0.0641	0.01376	0.00719	-0.0413	0.6157	0.5368
-0.500	0.0934	0.01378	0.00721	-0.0420	0.6151	0.5377
-0.250	0.1226	0.01381	0.00724	-0.0426	0.6145	0.5387
0.000	0.1519	0.01386	0.00729	-0.0433	0.6137	0.5398
0.250	0.1812	0.01394	0.00738	-0.0439	0.6130	0.5410
0.500	0.2103	0.01404	0.00749	-0.0446	0.6125	0.5423
0.750	0.2394	0.01415	0.00761	-0.0453	0.6119	0.5435
1.000	0.2685	0.01415	0.00764	-0.0459	0.6113	0.5446
1.250	0.2975	0.01414	0.00765	-0.0466	0.6103	0.5457
1.500	0.3265	0.01415	0.00769	-0.0472	0.6092	0.5467
1.750	0.3555	0.01420	0.00776	-0.0478	0.6081	0.5476
2.000	0.3844	0.01419	0.00778	-0.0485	0.6071	0.5485
2.250	0.4131	0.01415	0.00781	-0.0490	0.6059	0.5497
2.500	0.4418	0.01418	0.00791	-0.0496	0.6048	0.5508
2.750	0.4706	0.01423	0.00803	-0.0501	0.6035	0.5519
3.000	0.4996	0.01426	0.00812	-0.0506	0.6019	0.5530

3.250	0.5289	0.01426	0.00815	-0.0511	0.5999	0.5541
3.500	0.5584	0.01429	0.00819	-0.0517	0.5978	0.5552
3.750	0.5871	0.01439	0.00833	-0.0522	0.5951	0.5563
4.000	0.6152	0.01425	0.00827	-0.0525	0.5914	0.5575
4.250	0.6442	0.01407	0.00813	-0.0529	0.5870	0.5586
4.500	0.6745	0.01386	0.00790	-0.0534	0.5827	0.5598
4.750	0.7033	0.01378	0.00784	-0.0538	0.5781	0.5611
5.000	0.7310	0.01353	0.00766	-0.0539	0.5719	0.5623
5.250	0.7610	0.01332	0.00744	-0.0544	0.5671	0.5633
5.500	0.7893	0.01315	0.00732	-0.0547	0.5626	0.5645
5.750	0.8160	0.01285	0.00716	-0.0546	0.5560	0.5657
6.000	0.8446	0.01254	0.00685	-0.0548	0.5486	0.5668
6.250	0.8703	0.01229	0.00676	-0.0545	0.5398	0.5679
6.500	0.8970	0.01207	0.00662	-0.0544	0.5305	0.5690
6.750	0.9232	0.01187	0.00654	-0.0541	0.5162	0.5702
7.000	0.9468	0.01166	0.00632	-0.0534	0.4731	0.5715
7.250	0.9485	0.01260	0.00670	-0.0492	0.3779	0.5727
7.500	0.9493	0.01382	0.00754	-0.0451	0.3042	0.5739
7.750	0.9494	0.01493	0.00835	-0.0411	0.2451	0.5751
8.000	0.9473	0.01583	0.00904	-0.0366	0.2033	0.5763
8.250	0.9442	0.01675	0.00980	-0.0320	0.1690	0.5772
8.500	0.9435	0.01782	0.01072	-0.0283	0.1400	0.5781
8.750	0.9435	0.01897	0.01176	-0.0251	0.1163	0.5788
9.000	0.9437	0.02018	0.01290	-0.0223	0.0961	0.5801
9.250	0.9448	0.02155	0.01422	-0.0201	0.0805	0.5812
9.500	0.9477	0.02302	0.01566	-0.0184	0.0669	0.5823
9.750	0.9512	0.02458	0.01721	-0.0169	0.0563	0.5835
10.000	0.9567	0.02611	0.01873	-0.0158	0.0481	0.5846
10.250	0.9638	0.02757	0.02022	-0.0148	0.0423	0.5859
10.500	0.9668	0.02935	0.02200	-0.0136	0.0374	0.5871
10.750	0.9770	0.03062	0.02334	-0.0129	0.0343	0.5884
11.000	0.9827	0.03225	0.02496	-0.0120	0.0310	0.5899
11.250	0.9887	0.03388	0.02664	-0.0112	0.0286	0.5913
11.500	0.9974	0.03533	0.02815	-0.0105	0.0269	0.5926
11.750	1.0044	0.03694	0.02977	-0.0099	0.0250	0.5939
12.000	1.0031	0.03921	0.03208	-0.0088	0.0234	0.5951
12.250	1.0132	0.04059	0.03358	-0.0084	0.0226	0.5966
12.500	1.0220	0.04212	0.03520	-0.0080	0.0215	0.5980
12.750	1.0304	0.04374	0.03688	-0.0076	0.0205	0.5995
13.000	1.0371	0.04554	0.03873	-0.0072	0.0196	0.6009
13.250	1.0364	0.04803	0.04126	-0.0066	0.0185	0.6024
13.500	1.0452	0.04976	0.04308	-0.0064	0.0180	0.6039
13.750	1.0549	0.05145	0.04485	-0.0063	0.0173	0.6057
14.000	1.0626	0.05334	0.04683	-0.0062	0.0167	0.6073
14.250	1.0713	0.05522	0.04876	-0.0062	0.0160	0.6089
14.500	1.0799	0.05712	0.05070	-0.0065	0.0151	0.6107
14.750	1.0838	0.05949	0.05315	-0.0065	0.0147	0.6124
15.000	1.0866	0.06199	0.05576	-0.0063	0.0141	0.6142

15.250	1.0952	0.06406	0.05795	-0.0067	0.0136	0.6163
15.500	1.1021	0.06635	0.06036	-0.0071	0.0131	0.6185
15.750	1.1082	0.06877	0.06286	-0.0076	0.0126	0.6209
16.000	1.1149	0.07120	0.06535	-0.0084	0.0120	0.6232
16.500	1.1171	0.07737	0.07165	-0.0098	0.0109	0.6275
16.750	1.1216	0.08027	0.07470	-0.0107	0.0106	0.6300
17.000	1.1255	0.08335	0.07794	-0.0119	0.0102	0.6326
17.250	1.1276	0.08669	0.08141	-0.0131	0.0098	0.6352
17.500	1.1296	0.09009	0.08491	-0.0145	0.0094	0.6381
17.750	1.1308	0.09372	0.08862	-0.0161	0.0091	0.6408
18.000	1.1307	0.09757	0.09259	-0.0178	0.0088	0.6437
18.250	1.1292	0.10166	0.09679	-0.0196	0.0086	0.6466
18.500	1.1243	0.10630	0.10154	-0.0216	0.0084	0.6497
18.750	1.1179	0.11131	0.10668	-0.0239	0.0081	0.65261

#### 1.4. S809\_Experimental\_500000.txt

**Tabla 5** NREL S809 Airfoil Data from CSU; Re = 500,000 (Sin rotación) [3]

Alpha	CL	CD
-2.23	-0.06	0.006
-0.16	0.156	0.004
1.840	0.369	0.006
3.880	0.571	0.008
5.890	0.755	0.009
7.890	0.860	0.017
8.950	0.887	0.024
9.910	0.869	0.035
10.90	0.868	0.039
12.00	0.894	0.048
12.90	0.938	0.061
14.00	0.929	0.074
14.90	0.908	0.080
16.00	0.912	0.106
17.00	0.655	0.271
18.00	0.588	0.265
19.00	0.587	0.281
20.00	0.597	0.299
22.00	0.603	0.326
24.00	0.647	0.375
26.00	0.683	0.419
28.10	0.745	0.482
30.00	0.824	0.560
35.00	1.050	0.817
40.00	1.140	1.030

45.00	1.200	1.260
50.00	1.120	1.380
55.00	1.170	1.700
60.00	1.080	1.870
65.00	0.940	1.980
70.00	0.857	2.190
74.90	0.666	2.170
79.90	0.472	2.210
84.80	0.356	2.320
89.90	0.142	2.090

### **1.5. S809\_NREL\_63.txt**

**Tabla 6** NREL S809 Airfoil Data from NASA-Ames Tests: 30% Span [3]

Alpha	CL	CD
2.2	0.5484	-0.0033
4.1	0.7119	-0.0064
5.9	0.8708	-0.0054
7.8	0.9669	-0.0075
9.9	1.0945	0.0130
11.8	1.1211	0.0513
13.7	0.9088	0.1807
15.5	0.9909	0.2583
16.9	1.0490	0.3059
18.6	1.1797	0.3896
20.3	1.2364	0.4533
22.2	1.2548	0.5157
23.7	1.1794	0.5490
25.0	1.2025	0.5796
26.6	1.1043	0.5758
27.8	1.0717	0.6053
29.5	1.0152	0.6144
30.8	0.9985	0.6410
32.4	1.0065	0.6749
33.8	1.0119	0.7155
35.2	1.0060	0.7459

### **1.6. S809\_NREL\_30.txt**

**Tabla 7** NREL S809 Airfoil Data from NASA-Ames Tests: 30% Span [3]

Alpha	CL	CD
1.4	0.3651	0.0057
4.7	0.5994	0.0069

8.4	0.8102	0.0223
12.1	1.1068	0.0515
15.9	1.2110	0.1216
19.2	1.2802	0.2587
22.0	1.4406	0.4205
24.8	1.6713	0.6183
27.2	1.8962	0.7837
29.6	2.1607	1.0857
31.9	2.1057	1.2497
33.9	2.0150	1.3318
36.0	1.9491	1.4103
37.1	1.9037	1.4420
39.1	1.8369	1.4969
40.5	1.8164	1.5530
42.0	1.7517	1.5851
43.2	1.7312	1.6288
44.6	1.6769	1.6528
45.8	1.6142	1.6639
46.8	1.5688	1.6709

## Bibliografía

1. Vermeer, L. J., '*A Review of Wind Turbine Wake Research at TU Delft*', AIAA-2001-0030, 20th ASME Wind Energy Symposium, 39th Aerospace Sciences Meeting and Exhibit, ASME, Reno, NV, 2001.
2. Yin Lu, Y., '*Low-Reynolds-Number Aerodynamic Characteristics of NACA 0012 Airfoil Section at Angle of Attack from 0° to 180°*', University of Michigan, 2013.
3. Jonkman, J.M., '*Modelling of the UAE Wind Turbine for Refinement of FAST\_AD*', NREL, 2003.
4. Aerofoil Tool, online: <http://airfoiltools.com/>, Last accessed July 20, 2014.