University of Bath
Faculty of Engineering & Design

Word count: 2047

November 15, 2018

Advanced Helicopter Dynamics

Supervisor
D. Cleaver

Assessor

Author's Candidate Number
10838

Summary

A computational model was created for the first stage design of a technically optimal blade for a wind turbine using a NACA 0012 aerofoil. The turbine radius was set at 20m with a mean chord of 1m. The optimal design was found by adjusting the linear chord gradient, c_g, angular twist rate, θ_{TW}, and initial inclination, θ_0. The initial model was very limited and so tip losses and bending criteria were added, reducing the optimal annual energy production from 59.8% of the theoretical maximum to 49.7%. Tip deflection was a limiting factor for the design with optimal designs having the maximum normal tip deflection of 3m. The final optimal values were $c_g = .008$, $\theta_{TW} = -0.54°/m$ and $\theta_0 = 9.87°$.

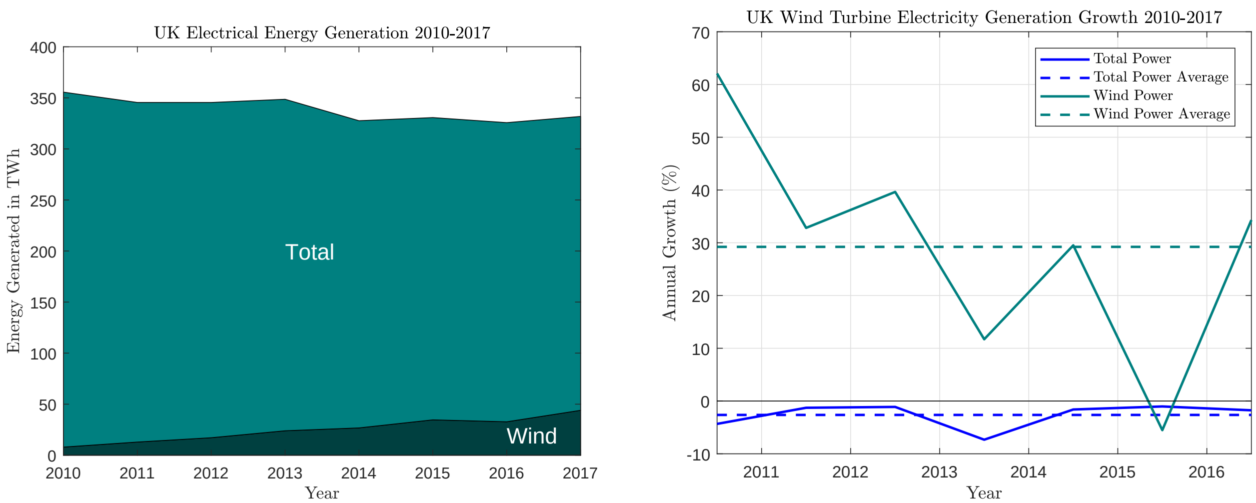
Contents

List of Figures

List of Tables

1 Introduction

Over the period 2010 to 2017 total electricity generation in the United Kingdom fell by 18% [1] as consumption also fell 10% across the period [2]. This is in spite of a rising population and number of households. This has been attributed to warmer winters lowering electric heating demands as well as improvements in refrigerator technology and increased use of LED lights [5]. Using data made available by the UK Department for Business, Energy & Industrial Strategy [1] it can be seen that wind power has defied the slowdown of electricity generation in the UK, rising on average 29% year on year over the same 2010 to 2017 period as shown by Figure 1b.



(a) Annual Electrical Energy Generation (b) Wind and Total Annual Growth

Figure 1: Statistics on UK Electrical Energy Generation

Traditionally the wind sectors growth was completely reliant on large government subsidies but in recent years governments such as the UK and Germany have adopted a new approach to subsidisation which has lead to a significant shift in the sector's economics. The policy change was to make energy companies bid for projects, and associated subsidies where available, meaning the actual subsidy received is the difference between the subsidy and the bid. This competitive approach (contract for difference) has driven innovation, with off-shore wind cost more than halving in the UK in just three years [6]. In some cases large wind projects have become viable with no subsidy or tax-break at all. The significance is the rate of development within the wind energy sector is now not reliant on government renewable energy policy but is shifting to large scale private sector investment. Due to the sheer size of the energy industry this could lead to significant wind turbine technology development. With the age of the electric car expected to dramatically reverse the trend of falling electricity consumption seen in Figure 1, research into wind turbine design has never been more relevant.

The optimal design of a wind turbine blade is one which produces the most power for the lowest unit cost [4]. This is would require a vast amount of cost data which is not readily available and so this paper shall instead focus on the design of a technically optimal design. Specifically, the design which has the highest annual energy production (AEP) for the parameters given in Table 1.

Table 1: Design Parameters for The Wind Turbine [4]

	Quantity	Symbol	Value	Units
DIMENSIONS	Hub Height	H_{hub}	35	m
	Hub-Tower Separation	d_{hub}	3	m
	Root Radius	R_{min}	1	m
	Maximum Radius	R_{max}	20	m
	Mean Chord	c_{mean}	1	m
FORCES	Maximum Root Bending Moment	$M_{root,max}$	0.5	MNm
	Maximum Vertical Hub Force	$F_{Y,max}$	70,000	N
QUANTITIES	Blade Density	ρ_{blade}	2,000	kg/m^3
	Blade Stiffness	$(EI)_{blade}$	-	GPa
	Cut-In Speed (assumed)	V_{min}	5	m/s
	Cut-Out Speed (assumed)	V_{max}	25	m/s
	Weibull Coefficient	A	7	-
	Weibull Coefficient	k	1.8	-
	Rotational Speed	ω	30	rpm

2 Methodology

To find the optimal design blade element momentum theory was used to calculate power output for a given velocity and parameter set. Using Weibull's wind speed distribution and power output for a given velocity, AEP was calculated. Part A of the flow diagram in Figure 2 shows the flow of information for the computer program used to calculate the AEP.

The design parameter's which could be changed for blade design were inclination at root: θ_0, rate of twist: θ_{tw} and chord gradient: c_g. The theoretical maximum power output for a given wind speed, V_0, and reference area, A, is the called the Betz limit and is given by equation 1. The maximum theoretical AEP can be found using the Weibull distribution and the Betz limit. Using a minimum finding function from Matlab's library the combination of the three parameters which gives the least difference to the Betz limit AEP can be found thus finding the optimal design for the given parameter set.

$$P_{Betz} = \frac{16}{27}\left(\frac{1}{2}\rho V_0^3 A\right) \tag{1}$$

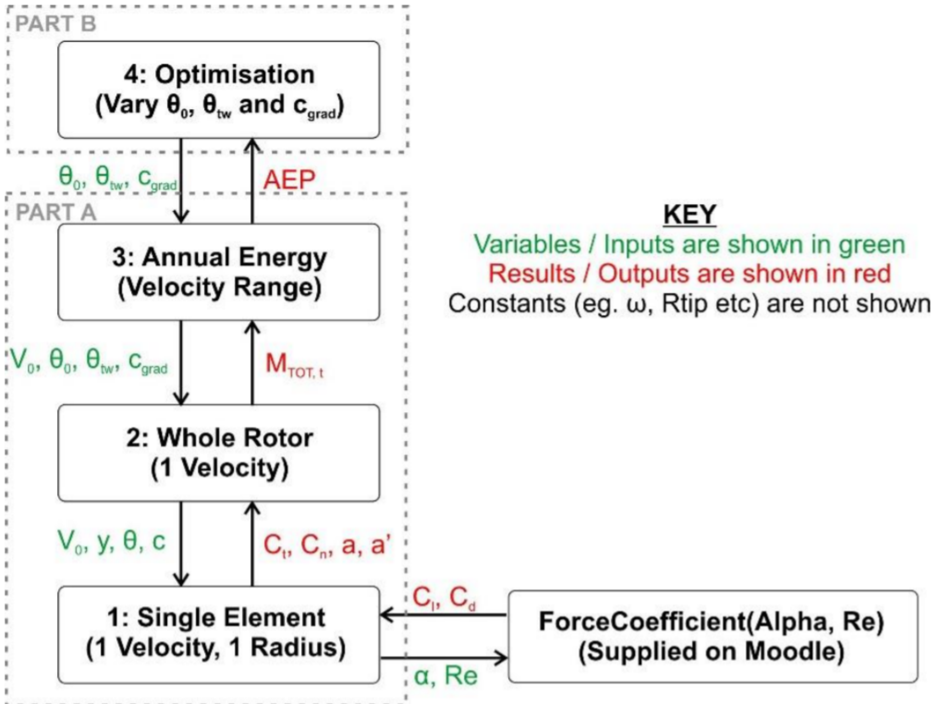


Figure 2: Flow Diagram of Method Used to Find Annual Energy Production [4]

3 Results and Discussion

3.1 Part A: Calculating AEP and Initial Optimisation

3.1.1 Validation of Model

The variable parameters used to interrogate the computer model are given *Set* 1 of Table 2 . These values were first used to validate the model against known results from a correct model. This showed almost exact agreement as shown in Figure 3.

$Variable$	Set 1	Set 2
θ_0	12°	14°
θ_{tw}	$-0.4°/m$	$-0.3°/m$
c_g	0	0

Table 2: Initial Values for Variable Parameters

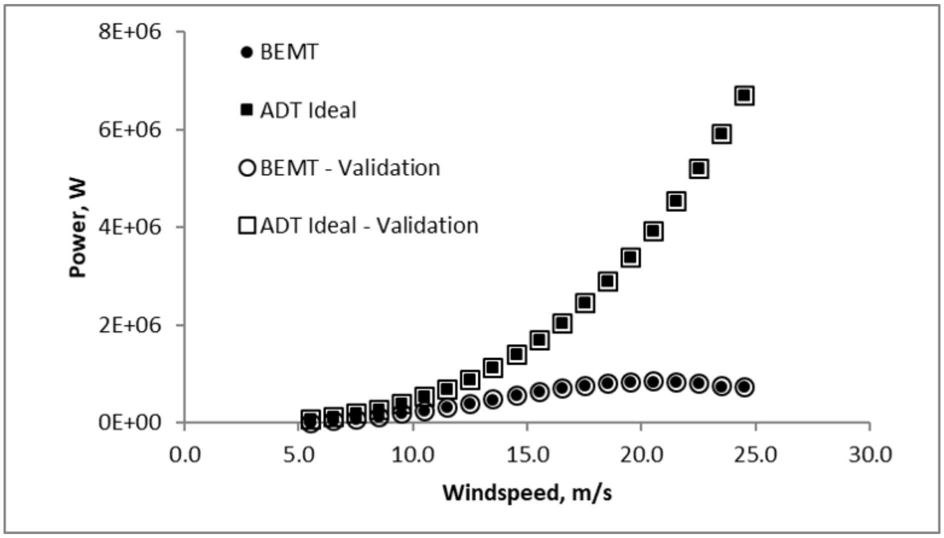


Figure 3: Validation of Model Against a Correct Model.

3.1.2 Power Variation with Wind Speed

The model described by Part A of Figure 2 was used to assess the effect of wind speed on power output. The result is shown by Figure 4 for the two sets of parameters given in Table 2. It can be seen that the turbine power output for *Set* 1 rises until approximately 22 m/s after which it begins to reduce. On the other hand the output of *Set* 2 increases with windspeed over the entire range and although slightly less than *Set* 1 until 19m/s the output for *Set* 2 is much greater in the 19-25 m/s region. As equation 1 shows the Betz limit is proportional to velocity cubed and this behavior is evident in Figure 4 with power output of both sets diverging from the Betz limit as wind speed increases.

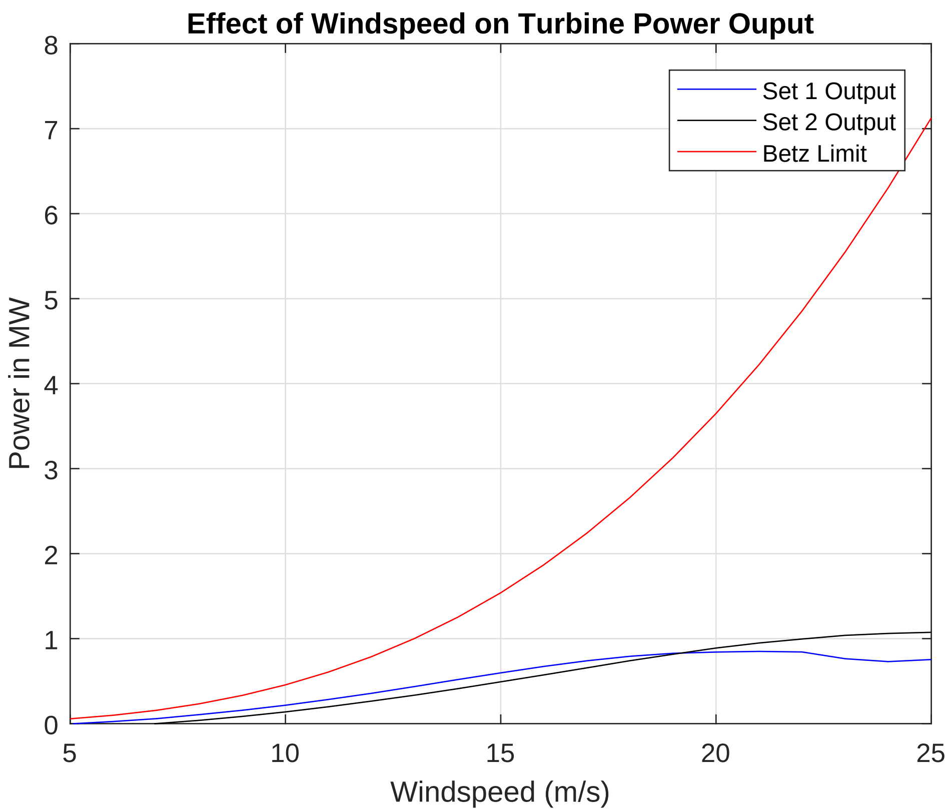


Figure 4: Validation of Model Against a Correct Model.

3.1.3 The Weibull Distribution

The Weibull distribution for wind speed is given by equation 2 and describes the frequency at which recorded wind speeds fall within a given speed range V_i to V_{i+1}. The parameters k and A are unique to the geographical location and are given in Table 1. Note that this is the only equation when A does not refer to turbine reference area. The Weibull distribution over the domain 5 m/s to 25 m/s has been plotted in Figure 5. The frequency at which windspeed exceeds 20 m/s is very low and by 25 m/s it is low enough to be insignificant which explains why cutting off generation at 25 m/s is sensible to turbine damage.

$$f(V_i < V_0 < V_{i+1}) = exp\left\{-\left(\frac{V_i}{A}\right)^k\right\} - exp\left\{-\left(\frac{V_{i+1}}{A}\right)^k\right\} \tag{2}$$

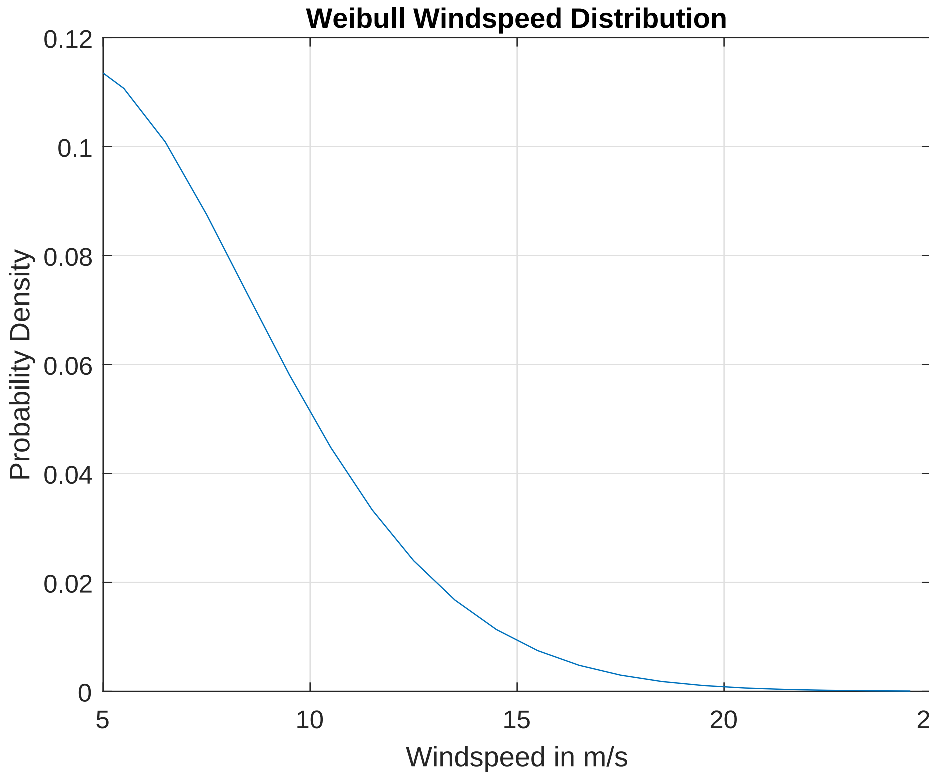


Figure 5: Weibull Windspeed Distribution with A =7 & k = 1.8.

3.1.4 AEP: Annual Energy Production

Figure 6 shows the power generated at each windspeed from Figure 4 multiplied by the probability density from Figure 5. Multiplying the area under a curve by the number of hours in the year gives the AEP. It can be seen that the low frequency at which windspeeds are in excess of 20 m/s means that the amount of energy generated beyond this speed is a fraction of that produced at lower speeds. As a result the large increase in power produced by *Set* 2 turbine compared to *Set* 1 at speeds at over 20 m/s is indistinguishable from Figure 6. Contrastingly the small increase in power generation of *Set* 1 over the 5 m/s to 20m/s speed range has been amplified by the high wind frequency in this region making it a much more effective design.

$Variable$	Betz	Set 1	Set 2	Units
AEP	2000	811	463	$(MWhr/year)$
% of Betz	-	41	23	%

Table 3: Initial Values for Variable Parameters

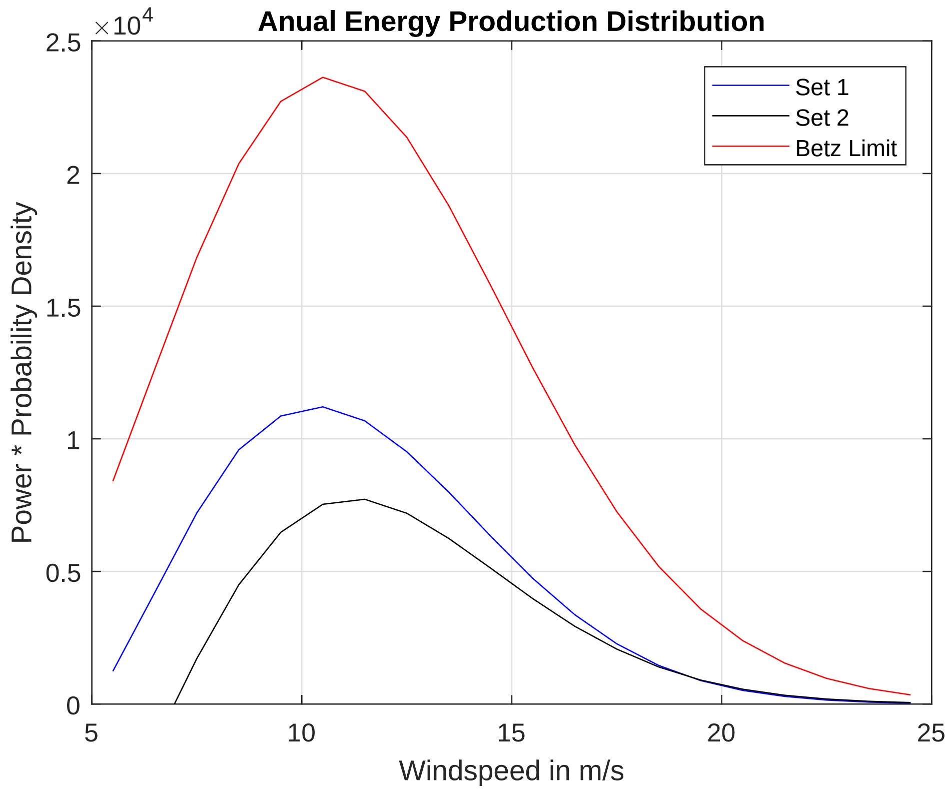
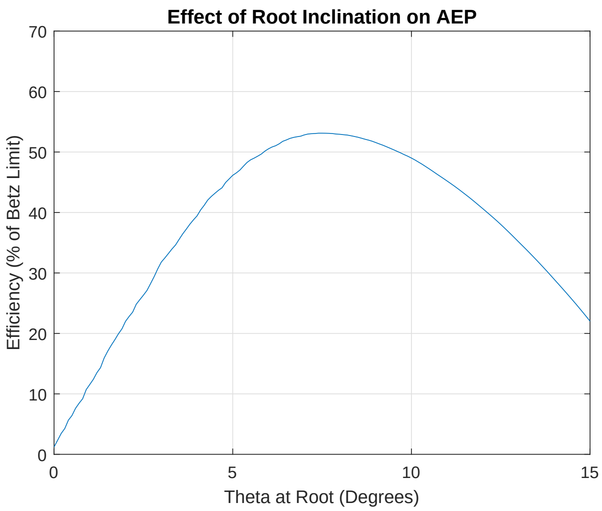


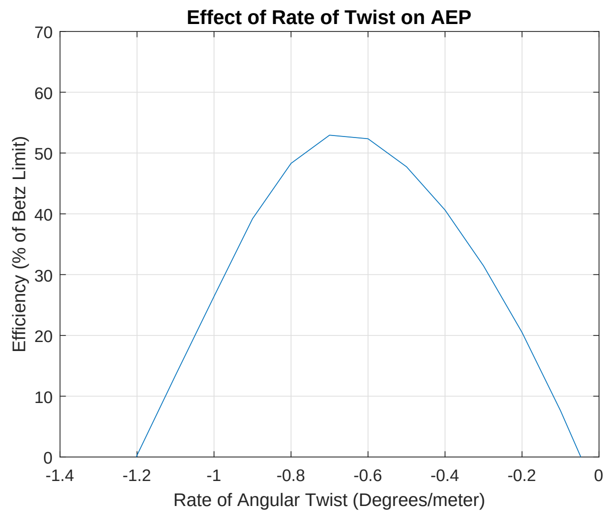
Figure 6: Power Multiplied by Probability Density for Each Windspeed

3.1.5 Optimal Design

The optimal design is the design for which the AEP is the highest percentage of the Betz limit AEP. The first stage in finding this set of parameters was to investigate the effect of each parameter individually. Using the parameters of Set 1 each parameter was adjusted with the rest remaining as the values from Set 1. Figures 7a and 7b show the effect of changing θ_0 and θ_{tw} on AEP respectively. Reading off the peaks approximately gives the optimal values of $\theta_0 = 7.5°$ and $\theta_{TW} = -0.7°/m$.



(a) The Effect of Root Inclination.



(b) The Effect of Rate of Twist.

Figure 7: Effect of Angular Parameters on Turbine AEP.

The effect of chord gradient was also investigated and the result is shown in Figure 8. The Figure shows AEP increases with increasing chord gradient. Larger chords give greater lift and so with the highest chord gradient there will be the greatest lift at the tip of the blade. This can be explained as rotational power is proportional to force, radius and rotational velocity. With rotational velocity set at 30rpm, by having lift increase with radius the maximum power output will be obtained. In reality turbine blades are not designed with large positive chord gradients and the model was later improved to reflect this. Temporarily disregarding this the chord gradient is only limited by the requirement to have a positive chord at 20m radius and at least 0.4m chord at the root for attaching to hub.

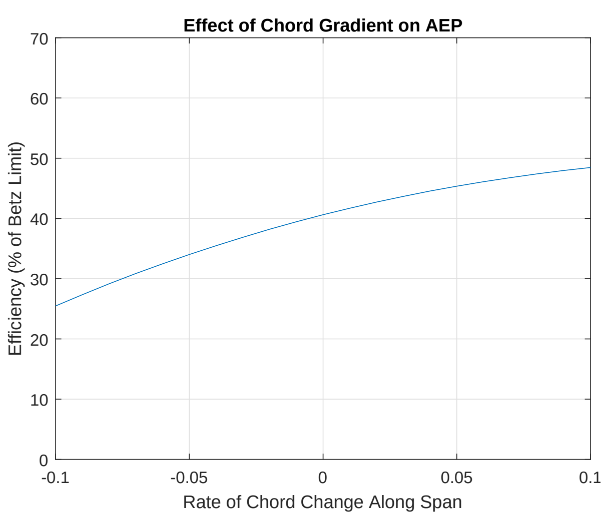


Figure 8: Effect of Chord Gradient on Turbine AEP.

A cost function was created which outputted the difference between AEP and the Betz limit AEP. A minimum finding algorithm was then used to find the values of θ_0, θ_{TW} and c_g which gave the optimum AEP. The result is shown in Table 4.

Parameter	**Value**	**Units**
θ_0	7.9	°
θ_{TW}	-0.37	$°/m$
c_g	0.084	m/m
AEP	1,190	MWhr/year
% of Betz	59.8	%
$M_{root,max}$	0.6	MNm
Tip Deflection	> 3	m

Table 4: Optimum for Initial Model

The function has improved efficiency compared to AEP$_{Betz}$ to 59.8% compared to 41% for the Set 1 starting case. The values are similar to those predicted from initial analysis which gives confidence that the minimum finding function has worked correctly. The criteria for tip deflection (maximum 3m) and maximum root bending moment (0.5MNm) were not met however and so this design is not viable. The chord gradient tended to the maximum which was constrained at 0.084 and the resulting high loads at the tip of the blade caused the excessive bending moment and deflection. Therefore an improved model was needed to keep these parameters below the requirements.

3.2 Part B: Limitations of Model and Enhancements

3.2.1 Limitations of Model

The model used in section A had several limitations which could be categorised into geometric, aerodynamic and structural.

- **Geometric:**
  1. *Linear Taper* - This is a fair simplification overall but near the tip, where velocities are large, a more complex shape or second taper rate may be optimal.
  2. *Linear Twist* - This simplifies the optimisation process but a non linear twist would likely increase AEP.
- **Aerodynamic:**
  1. *Tip Losses* - These have not been modelled which may have contributed to the unrealistic chord gradient (or taper rate) result.
  2. *Compressibilty* - No check for compressible flow at tip but as low speed operation this is reasonable assumption.

3. *Noise* - An technically optimal design should have noise limits. Large tip chords produce lots of noise so this aspect design should be considered.
4. *Small Values of a* - Assumed that a will not exceed 0.4 and so the Glauert Correction for a is not needed [7].

- **Structural:**

1. *Fatigue* - Fatigue considerations would give life of blade and optimisation could therefore be attempted for life of blade rather than single year. A blade which has 10% lower AEP is a much better design if it has 25 % longer life for example.
2. *Root Bending Moment* - This needs to be added to model to check it does not exceed the maximum specified in Table 1.
3. *Tip Deflection* - If tip deflection exceeds 3m the tip will collide with the tower and so the model needs to ensure this does not happen.

3.2.2 Adding Bending Analysis to Model

The most important limitations to correct were the root bending moment and tip deflection as then it could be assured the conditions in Table 1 are met. A new function was created to calculate these design parameters using a modified version of the method described by Hansen [7]. The beam is discretized as shown in Figure 9. Using the existing model the normal moments were found at each node as described by equation 3.

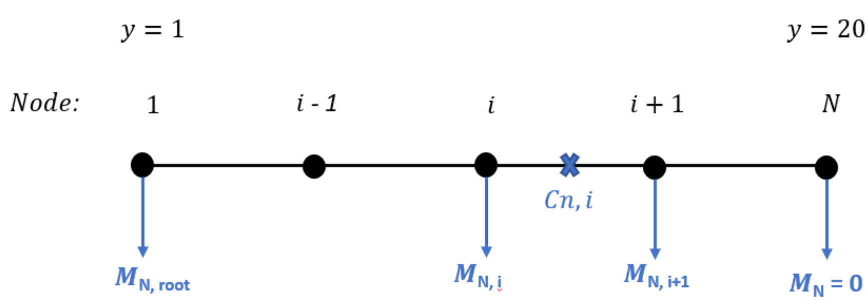


Figure 9: Discrete Beam for Deflection Analysis.

$$M_{N,i} = M_{N,i} + c_{N,i+1} * (0.5\rho V_{rel}^2) * (y_{i+1} - y_i) * y_i \tag{3}$$

The tangential bending moment was calculated in the same way after which the root bending moment was calculated using equation 4 with $V_0 = 25m/s$.

$$M_{root} = (M_{N,1} + M_{N,1})^{0.5} \tag{4}$$

The AEP was set to zero if the tip deflection or root bending moment exceeded the limits from Table 1. This led to a new optimum design which is displayed in Table 5 along with the initial model values for comparison. It should be noted that centrifugal force will resist motion and to improve accuracy the effect should have been accounted for.

Parameter	Initial Model	Bending Model	Units
θ_0	7.9	11	$^\circ$
θ_{TW}	-0.37	-0.6	$^\circ/m$
c_g	0.084	.006	m/m
AEP	1,190	1,080	MWhr/year
% of Betz	59.8	54.2	%
$M_{root,max}$	0.6	0.46	MNm
Tip Deflection	>> 3	3.0	m

Table 5: Comparison Between Initial and Bending Models

As expected the chord gradient was much reduced after implementing bending restriction now almost straight at $c_g = 0.006$. The restrictions also reduced the AEP of output to 54.2% from 59.8% of the Betz limit.

3.2.3 Adding Tip Losses to Model

As the optimal solution was had a slightly positive chord gradient the effect of tip losses was deemed potentially significant. Prandtl's tip loss correction factor was applied to the model in addition to the bending checks. The result has been added to Table 5 to produce Table 6.

Parameter	Initial Model	+ Bending	+ Tip Losses	Units
θ_0	7.9	11	9.87	$^\circ$
θ_{TW}	-0.37	-0.6	-0.54	$^\circ/m$
c_g	0.084	.006	.008	m/m
AEP	1,190	1,080	991	MWhr/year
% of Betz	59.8	54.2	49.7	%
$M_{root,max}$	0.59	0.46	0.44	MNm
Tip Deflection	>> 3	3.0	3.0	m

Table 6: Comparison Between Initial, Initial with Bending and Initial with Bending and Tip Losses Models

The tip loss addition was significant and led to a 8.2 % reduction in AEP. The comparison with the Betz limit can be seen graphically in Figure 10. For the final parameters the values of a very rarely exceeded 0.4 and so, although not ideal, the abscence of Glauert's correction factor is deemed acceptable.

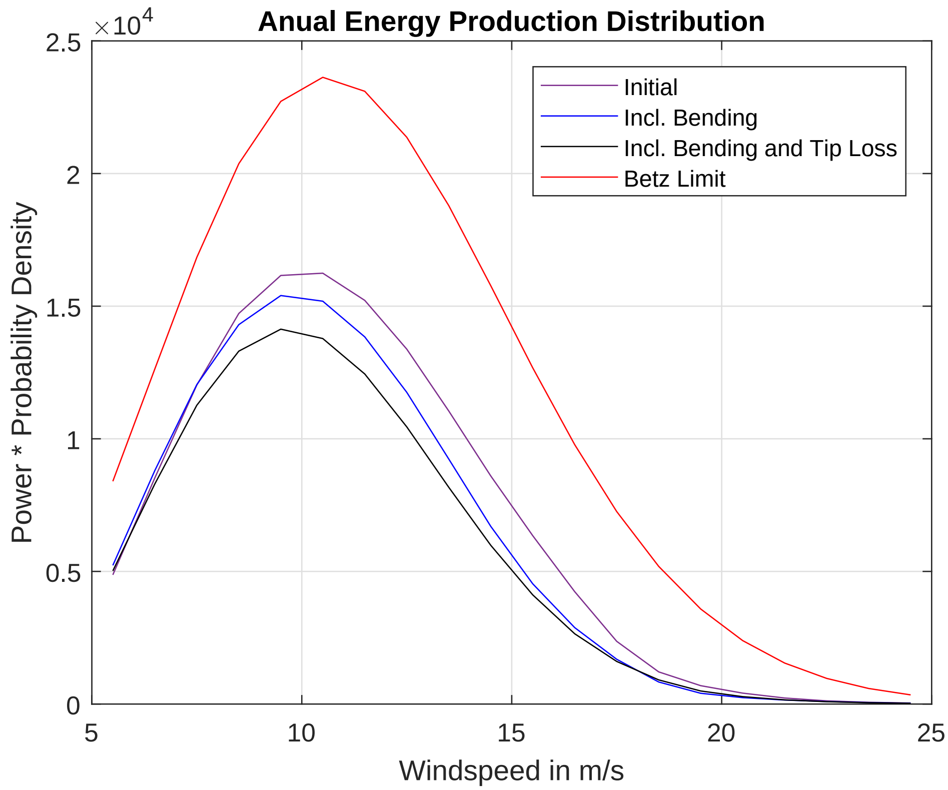


Figure 10: Comparison AEP as Features Were Added to Model.

3.2.4 Final Design

The final design chosen was the optimal design which included tip losses and bending analysis. Using the parameters from Table 6 and the set parameters from Table 1 a CAD model of the blade was made. The aerofoil used for the CAD model was a NACA 0012 as this is where the force coefficient data was derived from. The resulting engineering drawing is shown in Figure 11. It should be noted that the values of θ_0 from Table 6 are at $y = 0m$, so the at the root, where $y = 1m$, the actual inclination is: $\theta_{ROOT} = \theta_0 + \theta_{TW}$
The CAD model was also used to calculate the weight of the blade and it was found that the 'Maximum Vertical Hub Force' was exceeded by the weight of the three blades.

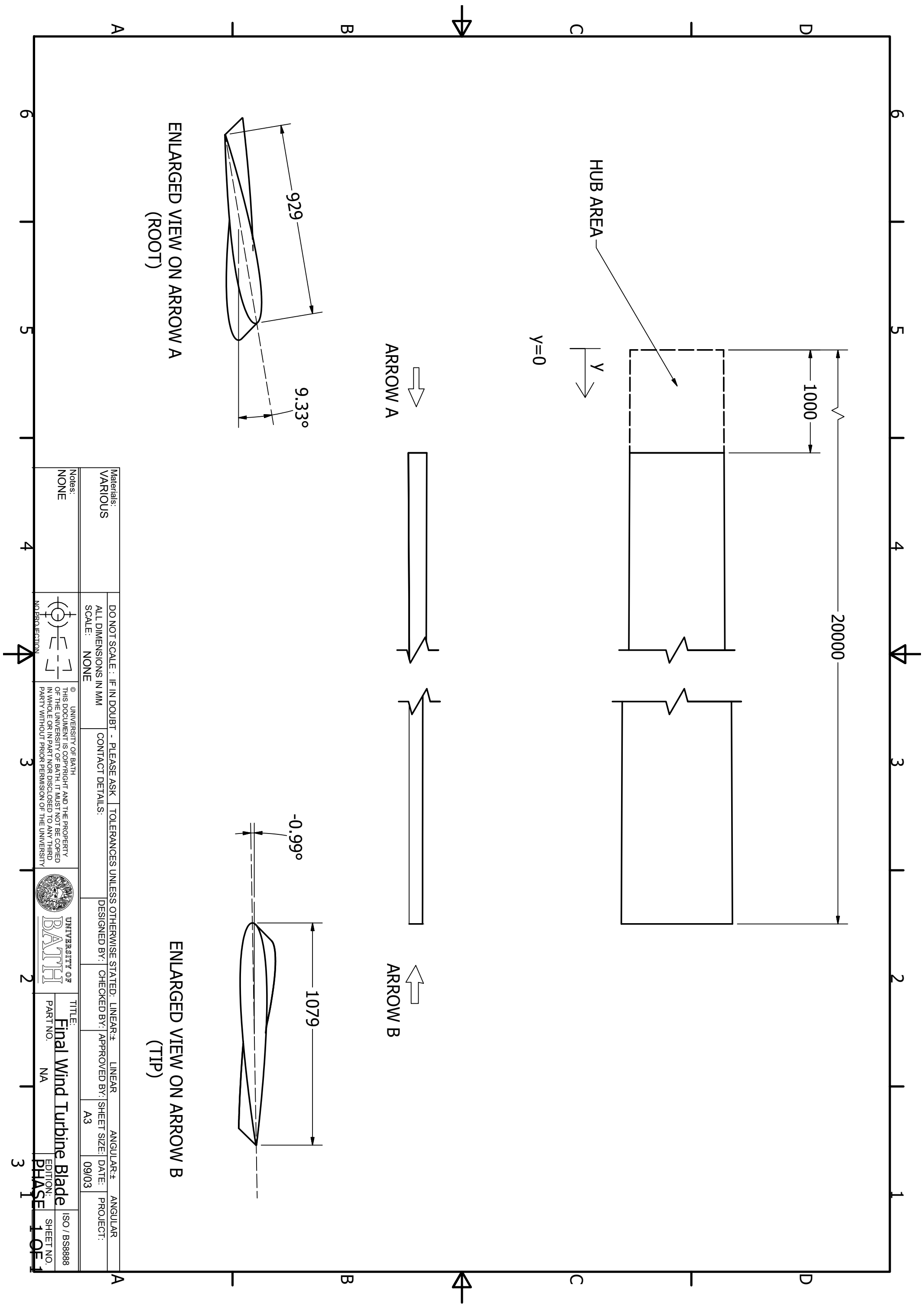


Figure 11: Drawing of Final Design.

Variable	Value	*Units*
Blade Density	2000	kg/m^3
Blade Volume	1.57	m^3
Single Blade Mass	3,150	kg
Number of Blades	3	
Maximum Vertical Hub Force	70,000	N
Combined Weight of Blades	92,600	N

Table 7: Weight Data for Final Design

4 Conclusions

A first stage design was found which had an AEP which was 49.7% of the Betz limit. All criteria were met except weight restrictions and so a lighter construction material would be required. Some major assumptions in the original model were accounted for to obtain a realistic design, with tip deflection becoming a limiting factor for chord gradient and tip losses reducing maximum AEP by 8.2 %. Despite many further limitations of the model due to the simple linear twist and taper design input parameters the model was indeed suitable. To improve the design it is suggested to implement additional linear twist rate or a non-linear twist and have a bespoke designed tip. With the more advanced design some more of the limitations discussed in the report should be accounted for.

References

[1] Department for Business, Energy & Industrial Stratergy, 2018. *Historical Electricity Data:1920 to 2017.*
Available: www.gov.uk/government/statistical-data-sets/historical-electricity-data

[2] Digest of United Kingdom Energy Statistics (DUKES), 2018. *Digest of UK Energy Statistics (DUKES): electricity.* Available:https://www.gov.uk/government/collections/electricity-statistics

[3] Cleaver, D., 2018. *Slides: WT Coursework.*
University of Bath.

[4] Cleaver, D., 2018. *ME40343 Helicopter Dynamics: Coursework.*
University of Bath.

[5] Vaughan, A., 2017. *Electric cars will fuel huge demand for power, says National Grid.*
The Guardian.

[6] Arie, S., 2018. *Renewables are primed to enter the global energy race.*
The Financial Times.

[7] Hansen, M.O.L., 2000. *Aerodynamics of Wind Turbines.*

Appendices

A Induced Calculations Function

```
function [a_out, adash_out, phi, Cn, Ct] = ...
    WTInducedCalcs(a_in, adash_in, V0, omega, y, theta, chord, B)
%1: SINGLE ELEMENT: use an iterative solution to find the ...
    values of a,
%adash, phi, Cn and Ct at a particular radius.

%% Set constants
rho = 1.225;          %density of air in kg/m^3
mu = 18.81e-6;        %kinematic viscosity in Pa.s
R = 20;
error_tol = .0001;  % setting allowable error
error = error_tol + 1; %set first error to initiate while loop

%% loop over until error goes below tolerance
i = 0;      %Initialize counter, will need to break loop if i ...
    exceeds 100
while error ≥ error_tol && i <101
    i = i + 1;

    %% Calculate  angles
    phi = atan(((1-a_in)*V0) / ((1+adash_in)*omega*y));
    alpha = phi - theta;

    %% Caluclate f and F for Prandtl tip loss factor (Hansen ...
        chapter 6)
    f = (B/2) * ((R-y) / (y * sin(phi)));
    F = (2/pi) * acos(exp(-f));
    %% Calculate Reynolds number
    V_rel = ((V0*(1-a_in))^2 + ((omega*y)*(1 + adash_in))^2)^.5;

    Re = (rho*V_rel*chord) / mu;

    %% Get non-dimensional lift and drag coefficients and convert
    [Cl, Cd] = ForceCoefficient(alpha, Re);
    Cn = Cl*cos(phi) + Cd*sin(phi);          %Normal Force ...
        Coefficient
    Ct = Cl*sin(phi) - Cd*cos(phi);          %Tangential Force ...
        Coefficient

```

```
    %% Calculate new values for a and a' for no tip loss:
    sigma = (B*chord) / (2*pi()*y);
    a_out = 1 / (((4*sin(phi)^2)/(sigma*Cn)) + 1);
    adash_out = 1 / (((4*sin(phi)*cos(phi))/(sigma*Ct)) - 1);

%     %% Calculate a and a' with tip loss
%     sigma = (B*chord) / (2*pi()*y);
%     a_out = 1 / (((4*F*sin(phi)^2)/(sigma*Cn)) + 1);
%     adash_out = 1 / (((4*F*sin(phi)*cos(phi))/(sigma*Ct)) - 1);
    %% Calculate error
    error = abs(a_out - a_in) + abs(adash_out - adash_in);

    %% If error greater than allowable error apply relaxation ...
       factor,
    %else the solution has been found and while loop can be ...
       broken
    if error ≥ error_tol
        a_in = 0.1*(a_out - a_in) + a_in;
        adash_in = 0.1*(adash_out - adash_in) + adash_in;
    else
        break
    end

end

%% If counter exceed 100 set a' to zero and solve for a

if i > 100

    a_in = 0;
    adash_out = 0;
    while error ≥ error_tol

        %% If counter reaches 500 the error tolerance
        % can be decreased to get a result.

        i = i+1;    %Increase counter

        if i ≥ 100             %Decrease error tol after 500 ...
           interations
            error_tol = 0.01;
        end
        %% Calculate  angles
        phi = atan(((1-a_in)*V0) / ((1)*omega*y));
        alpha = phi - theta;
```

```
        %% Caluclate f for Prandtl tip loss factor (Hansen ...
           chapter 6)
        f = (B/2) * ((R-y) / (y * sin(phi)));
        F = (2/pi) * acos(exp(-f));

        %% Calculate Reynolds number
        V_rel = ((V0*(1-a_in))^2 + ((omega*y)*(1))^2)^.5;
        Re = (rho*V_rel*chord) / mu;

        %% Get non-dimensional lift and drag coefficients and ...
           convert
        [Cl, Cd] = ForceCoefficient(alpha, Re);

        Cn = Cl*cos(phi) + Cd*sin(phi);         %Normal Force ...
           Coefficient
        Ct = Cl*sin(phi) - Cd*cos(phi);         %Tangential ...
           Force Coefficient

        %% Calculate new values for a and a' with no tip loss:
        sigma = (B*chord) / (2*pi*y);
        a_out = 1 / (((4*sin(phi)^2)/(sigma*Cn)) + 1);

%         %% Calculate a and a' with tip loss
%         sigma = (B*chord) / (2*pi()*y);
%         a_out = 1 / (((4*F*sin(phi)^2)/(sigma*Cn)) + 1);

        %% Calculate error
        error = abs(a_out - a_in);

        %% If error greater than allowable error apply ...
           relaxation factor,
        % else the solution has been found and while loop can ...
           be broken
        if error ≥ error_tol
            a_in = 0.1*(a_out - a_in) + a_in;  %Relaxation ...
               facto k=.1 added
        else
            break
        end

    end

end
end
```

B Wind Turbine Single Velocity Function

```
function [MT, MN, MT_local, MN_local] = WTSingleVelocity(V0, ...
   theta0, theta_twist, chord_mean, chord_grad, TipRadius, ...
   RootRadius, omega, B)
%2: WHOLE ROTOR - loop WTInducedCalcs to find the values for ...
   all radii,
%then integrate these to get the normal and tangential moment ...
   at the blade
%root.

%% Set constants
rho = 1.225;        %density of air in kg/m^3

%% Set up geometry
y = 1.5:1:19.5;   %Vector of local radii (elements) (in m)
y_△ = y(2) - y(1);  %△ is difference between adjacent elements

R = TipRadius;

%% Set up empty outputs arrays to improve efficiency
MN_local = zeros(length(y),1);
MT_local = zeros(length(y),1);
%% 2: Calculate individual element moment both in plane and ...
   normal

for i = 1:length(y)

    %Calculate local chord and angle of attack
    chord_local = chord_mean + ((y(i)) - (R/2))*chord_grad;
    assert(chord_local ≥ 0 && chord_local ≤ 2, strcat('Error ...
       in chord length! Chord length = ', num2str(chord_local)))
    theta_local = theta0 + (y(i))*theta_twist;

    %Calculate Cn and Ct for the local element
    [a_out, adash_out, ¬, Cn, Ct] = WTInducedCalcs(0, 0, V0, ...
       omega, y(i), theta_local, chord_local , B);
%   disp(strcat('Aout: ' , num2str(a_out)))
    % Calculate Relative Velocity
    V_rel = ((V0*(1-a_out))^2 + ((omega*y(i))*(1 + ...
       adash_out))^2)^.5;

    %Calculate the local moment in normal and tangential ...
       (torque) directions
```

```
    MN_local(i) = (.5*rho*(V_rel^2)*chord_local*Cn)*y_∆*y(i);
    MT_local(i) = (.5*rho*(V_rel^2)*chord_local*Ct)*y_∆*y(i);

end

%% Integrate the moments to find totals
MN = sum(MN_local);
MT = sum(MT_local);

end
```

C Wind Turbine Velocity Range Function

```
function [Diff] = WTVelocityRange(Parameters)
%3: ANNUAL ENERGY - loop WTSingleVelocity to find the moments ...
    across the
%entire velocity range. Combine this with the frequency ...
    information to get
%the AEP. Parameters = [theta0, theta_twist, chord_grad]

%% Setting as constants%%%%
A = 7;
k = 1.8;
omega = 3.14;
chord_mean = 1;
TipRadius = 20;
RootRadius = 1;
B = 3;
MinV0 = 5;
MaxV0 = 25;

%% Constants
rho = 1.225;         %density of air in kg/m^3
Area = (TipRadius^2)*(pi);           %Area in m^2

%% Calculate total tangential moment for each speed
∆_V0 = 1;
V0 = MinV0:∆_V0:MaxV0;       % Set up wing velocity range in m/s
MT = zeros(length(V0), 1);       % Initialise tangential moment ...
    vector for each wind speed
MN = zeros(length(V0), 1);       %Initialise normal moment ...
    vector for each wind speed
```

```
Power = zeros(length(V0), 1);    %Initialise power vector for ...
    each wind speed

P_speed = zeros(length(V0)-1, 1);            %Initialise vector ...
    of speed probablilities
AEP_speed = zeros(length(V0)-1, 1);          %Initialise vector ...
    of annual power at each speed
Power_midpoint = zeros(length(V0)-1, 1);     %Initialise vector ...
    of power inbetween speeds
AEP_speed_ideal = zeros(length(V0)-1, 1);
Power_ideal = zeros(length(V0)-1, 1);

%% Calculate power for each speed

[MT(1), MN(1)] = WTSingleVelocity(V0(1), Parameters(1), ...
    Parameters(2), chord_mean, Parameters(3), TipRadius, ...
    RootRadius, omega, B);

Power(1) = MT(1)*B*omega;

for i = 2:length(V0)

    [MT(i), MN(i), ¬, ¬] = WTSingleVelocity(V0(i), ...
        Parameters(1), Parameters(2), chord_mean, ...
        Parameters(3), TipRadius, RootRadius, omega, B);

    Power(i) = MT(i)*B*omega;

    %%%Calculating power and probablilities in the speed ...
        intervals
    P_speed(i-1) = exp(-(V0(i-1)/A)^k) - exp(-(V0(i)/A)^k); ...
        %Calculate probability of speed between Vi and Vi+1

    Power_midpoint(i-1) = 0.5*(Power(i-1) + Power(i)); ...
                 %Power at mid point is average of lower and ...
        upper bounds
    Power_ideal(i-1) = (16/27) * ...
        0.5*rho*(((V0(i-1)+V0(i))/2)^3)*Area;  %Theoretical ...
        maximum power defined by Betz limit

    AEP_speed(i-1) = Power_midpoint(i-1) * P_speed(i-1) * ...
        8760;         %Calculate predictied anual power ...
        generated at each windspeed range
    AEP_speed_ideal(i-1) = Power_ideal(i-1) * P_speed(i-1) * ...
        8760;   %Calculate the ideal anual power generated at ...
```

```
        each windspeed range

end

AEP = sum(AEP_speed);
AEP_ideal = sum(AEP_speed_ideal);

% [tip_deflection, M_root] = ...
    WTBendingDeflection(Parameters(1), Parameters(2), ...
    Parameters(3));
% disp(strcat('Tip deflection: ', num2str(tip_deflection), '  ...
    M_root: ', num2str(M_root)));
% if  tip_deflection > 3 || M_root > 0.5e6
%     AEP = 0;
% end
Diff = AEP_ideal - AEP;
assert(Diff > 0, 'Error!Predicted power greater than ideal.')

end
```

D Wind Turbine Velocity Range Function

```
function [x] = TurbineOptimisation()
%%%%Script to find the optimal solution for theta0, ...
    theta_twist and
%chrod gradient

%OPTIMISE FIT - 3 variables: Theta0, Theta_grad and Chord_grad
opts = optimset('fminsearch');
opts.Display = 'iter'; %What to display in command window
opts.TolX = 0.001; %Tolerance on the variation in the parameters
opts.TolFun = 1e7;  %Tolerance on the error
opts.MaxFunEvals = 150; %Max number of iterations

%fminsearch inputs: (function, initial guess, lower limits on ...
    variables,
% upper limits on variables, options)

%Limit for Theta0 is set to +/- 20 degrees
%Limit for rate of twist is set so 45 degrees twis over blade ...
    is the
%maximum
```

```
% Limit for chord grad set so there is at least 20cm chord at ...
    root for
% attatchment and so chord is positive at R = 20

[x] = fminsearchbnd(@WTVelocityRange, [5*pi/180 -.3*pi/180 ...
    -.03], [-20*pi/180 -2.36*pi/180 -0.105], [20*pi/180 ...
    2.36*pi/180 .07], opts);

disp(strcat('OPTIMAL VALUES: Theta0 = ', num2str(x(1)*180/pi), ...
    ' Twist Rate = ', num2str(x(2)*180/pi), ' Taper Rate = ', ...
    num2str(x(3))))

%% Check bending moment does not Exceed 0.5MNm
%WTSingleVelocity inputs => V0, theta0, theta_twist, ...
    chord_mean, chord_grad, TipRadius, RootRadius, omega, B

%[MT, ¬] = WTSingleVelocity(25, x(1), x(2), 1, x(3), 20, 1, ...
    30, 3);

%disp(MT)

%% Compute and display how
BetzLimit = 1.99684E+09;
diff = WTVelocityRange(x);
eff = (BetzLimit - diff) / BetzLimit * 100;

disp(strcat('Percentage of Betz Limit = ', num2str(eff), ' %'))



end
```

E Flow Chart of FEM Solver