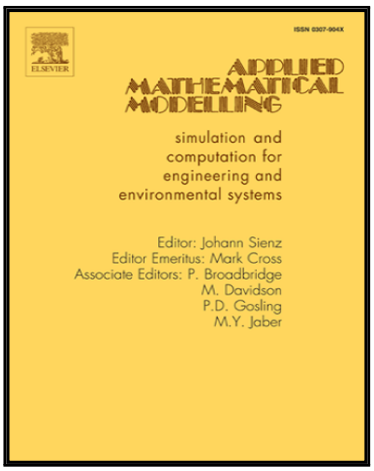


Computational Aerodynamic Optimisation of Vertical Axis Wind Turbine Blades

Matt Kear, Ben Evans, Rob Ellis, Sam Rolland

PII: S0307-904X(15)00398-4
DOI: 10.1016/j.apm.2015.07.001
Reference: APM 10633

To appear in: *Applied Mathematical Modelling*

Received date: 10 October 2014
Revised date: 24 April 2015
Accepted date: 1 July 2015

Please cite this article as: Matt Kear, Ben Evans, Rob Ellis, Sam Rolland, Computational Aerodynamic Optimisation of Vertical Axis Wind Turbine Blades, *Applied Mathematical Modelling* (2015), doi: 10.1016/j.apm.2015.07.001

Highlights

- CFD used for novel vertical axis wind turbine design.
- A new parameterisation scheme developed
- Design of Experiments and Nelder-Mead simplex optimisation used.
- New blade design developed – currently being tested.

Available online at www.sciencedirect.com

ScienceDirect

Applied Mathematical Modelling 00 (2015) 1–18

App. Math. Mod.

Computational Aerodynamic Optimisation of Vertical Axis Wind Turbine Blades

Matt Kear[1], Ben Evans[1] †, Rob Ellis[2], Sam Rolland[1]

[1]Zienkiewicz Centre for Computational Engineering, Swansea University, SA2 8PP
[2]C-FEC Ltd, Digital Technium, SA2 8PP
†*b.j.evans@swansea.ac.uk*

Abstract

The approach and results of a parametric aerodynamic optimisation study is presented to develop the blade design for a novel implementation of a vertical axis wind turbine. It was applied to optimise the two-dimensional cross-sectional geometry of the blades comprising the turbine. Unsteady viscous computational fluid dynamic simulations were used to evaluate blade performance. To compare geometries, the non-dimensional Coefficient of Power was used as a fitness function. Moving meshes were used to study the transient nature of the physical process. A new parameterisation approach using circular arcs has been developed for the blade cross sections. The optimisation process was conducted in two stages: firstly a Design of Experiments based response surface fitting was used to explore the parametric design space followed by the use of a Nelder-Mead simplex gradient-based optimisation procedure. The outcome of the optimisation study is a new blade design that is currently being tested in full-scale concept trials by a partnering wind energy company.

Keywords: aerodynamic optimisation, VAWT, parametric design, Design of Experiments, Nelder-Mead

Nomenclature

- BEMT - Blade Element Momentum Theory
- CFD - Computational Fluid Dynamics
- C-FEC - Cross Flow Energy Company
- DoE - Design of Experiments
- HAWT - horizontal axis wind turbine
- LHS - Latin Hypercube Sampling
- RMSE - root mean square error
- VAWT - vertical axis wind turbine
- A - turbine frontal area
- C_p - coefficient of power
- R_0 - blade outer arc radius (X_1)
- A_0 - blade half arc angle of outer arc (X_2)

- T_L - blade left thickness (X_3)
- T_C - blade centre thickness (X_4)
- T_R - blade right thicknes s (X_5)
- Y - blade yaw angle (X_6)
- Ψ - blade azimuthal angle
- Ω - turbine rotational speed
- U_∞ - freestream wind speed
- P - power
- T_i - torque on blade i

1. Introduction

1.1. Background and Motivation

In recent years, environmental and economic pressures to reduce society's dependence on fossil fuels in power generation has led to significant research investment into sustainable power generation. A leading contender in this field is the extraction of energy from the wind using turbines, often co–located as large wind farms, which can be based on either land or at sea.

A vast array of academic and industrial optimisation studies have been performed on wind turbines. Tu *et al.* conducted an optimisation study on a Horizontal Axis Wind Turbine (HAWT) using a combination of CFD simulations and neural networks to evaluate turbine performance [1]. This was incorporated into a genetic algorithm for optimisation. However, not every study utilises CFD simulations for performance evaluations. This is because they require a significant computational resource. Instead, some studies such as those carried out by Cencelli *et al.* and Dossing *et al.* [2, 3] have performed optimisations on HAWTs using Blade Element Momentum Theory (BEMT) which places a much lower demand on computational resource. However, BEMT cannot readily be applied to Vertical Axis Wind Turbine (VAWT) studies. This is because BEMT fundamentally relies on the concept of an actuator disk and known lift and drag blade properties. While the first shortcoming can be circumvented through the use of the double multiple stream tubes method [4], a prerequisite knowledge of the blade's lift and drag properties is incompatible with a numerical optimisation of its aerodynamic behaviour.

This work builds upon this knowledge and uses CFD tools to develop an automated optimisation program that does not require human interaction to search for a potentially optimal geometry. The focus of the optimisation study is on the two-dimensional blade cross section of a VAWT. The rest of this section outlines the problem and clearly defines the objectives of the research. Following this an insight into geometry parametrisation is given with the developed scheme shown in detail. Then, different aspects of the CFD simulations such as geometry meshing and the flow field solver are introduced. A sample is taken of the almost infinite number of possible blade shapes (design space) and an analysis is conducted to identify any possible trends in an attempt to gain a basic understanding of what is to be expected. Finally, the results obtained are presented and analysed.

1.2. Design Overview of C-FEC Vertical Axis Wind Turbine

The research study presented in this paper was undertaken to assist a wind turbine design company, Cross-Flow Energy Company Ltd (C-FEC Ltd), in the development of a blade optimisation method for their wind turbine design. The concept turbine developed by the company is a Vertical Axis Wind Turbine (VAWT) using both lift (Darrieus effect) and drag (Savonius effect) to extract energy from the wind and produce the torque necessary to generate an exploitable form of energy. A limited number of optimisation studies are provided in the literature on the geometric optimisation of VAWTs that are designed specifically to take advantage of both lift and drag on the turbine blades for torque generation. Gupta *et al* [5] have considered the efficiency improvements in the combination of Darrieus and Savonius effects in the design of a VAWT compared with a pure Savonius design . Manohar *et al* [6] have considered using the centrifugal effect to rotate a turbine's blades to move from operation in Savonius mode to operation in Darrieus mode to avoid the problem with self-starting in Darrieus turbines whilst taking advantage of the usually higher efficiency of Darrieus turbines operating at high speed. The C-FEC

turbine design incldues an additional and novel, as far as the authors are aware, asymmetric shield. This feature, which moves dependent on the oncoming wind direction, is not commonly seen in commercial wind turbines and increases turbine efficiency by shrouding the returning blades and thus minimising the negative torque that they would otherwise be generating. Figure 1 shows a top-down view of the turbine designed by C-FEC Ltd with key components labelled. The azimuthal angle (ψ) is used to give reference to blade positions. In this paper it is defined as the angle between the direction of the free stream flow and the centre of the blade being considered. The green arrows indicate positive torque that is used in generating power and the red arrows indicate negative torque that hinders the production of power.

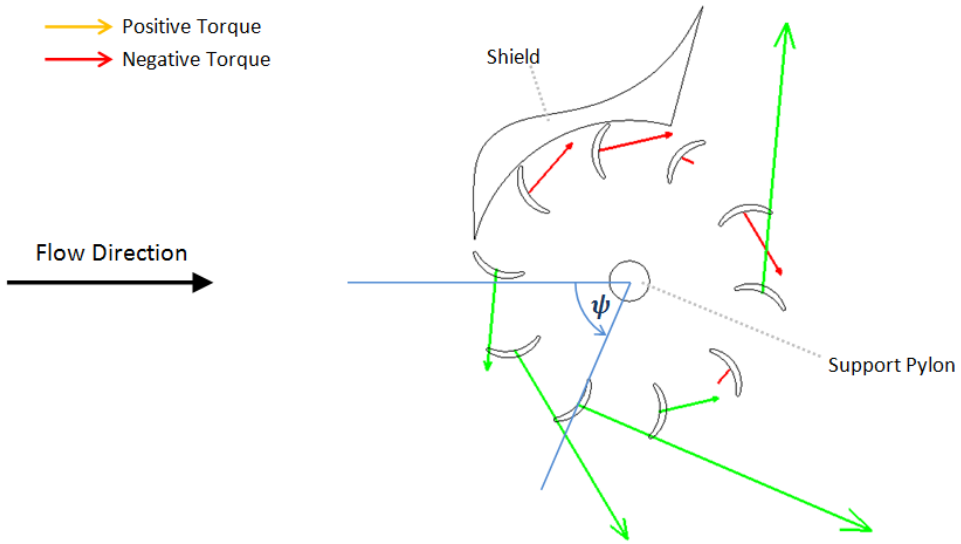


Figure 1: Top view of C-FEC's turbine concept.

As the blades move perpendicularly to the wind direction (at ψ approaching 90°) drag becomes the dominant force. As the blades then move to become parallel to the flow (at ψ approaching 180°) the drag reduces and lift increases, making it the dominant force in this position. It is this combined use of lift and drag to create torque that potentially makes this machine very efficient. To further improve the design, the shield that is in place aids in reducing the drag on the retreating blades ($180° < \psi < 360°$) by redirecting the freestream flow. It is the hybrid nature of this design that provided the motivation for exploring a new method for blade shape parameterisation.

1.3. Objectives

The prime objective of this research was to develop a new blade shape, building on existing work carried out by C-FEC Ltd [7, 8] that took advantage of hybrid aerodynamic torque i.e. effectively utilising both lift and drag in torque generation.

To evaluate the performance of each blade geometry, unsteady viscous CFD simulations were used to calculate the Coefficient of Power (Cp). The way in which Cp is defined in this study is detailed in Section 4.2.

2. Geometry Parameterisation

2.1. Background to parameterisation approaches

There are a wide range of techniques available for geometry parameterisation in the context of aerodynamic optimisation. Many of these approaches are reviewed in [9]. The correct selection of a parameterisation scheme is essential to the success of an optimisation process. The suitability of a technique is always a balance between how flexible the representation of the model can be and the number of design variables required. The more flexibility there is in the model, the greater the probability that the optimum shape exists within the design space. However, to gain more flexibility requires more design variables. If the number of design variables becomes too large then

the design space becomes unmanageable and it becomes improbable that the optimum will be found [10]. It is the sensitive nature of the parameterisation that provides the motivation for careful parameter selection.

Typically, for devices such as Darrieus turbines and HAWTs, which primarily rely on the creation of lift to produce torque, a standard aerofoil cross section is utilised. Extensive work has been undertaken to efficiently parametrise aerofoils and a popular scheme is PARSEC. The PARSEC parameterisation uses 12 design variables to control the shape [11, 9]. It would be impractical to extend PARSEC to be used in this study. The parameterisation is tailored to produce sharp points at the trailing edge, whereas in this application flexibility is required to be able to create smooth and rounded edges at trailing edge as well as the leading edge.

Other VAWT optimisation studies that are primarily utilising drag to produce torque have only focused on the effects of drag production. As a result they have not been interested in the effects of blade thickness, limiting the blades to be thin curves. This approach can be seen in Ghatage and Joshi [12] and Ypma [13], and is not appropriate for this study which considers both lift and drag.

A new parameterisation methodology has been developed by the authors that is tailored for hybrid blades operating within the context of the overall C-FEC turbine design. The parameterisation has been designed with automation in mind to allow for an easy integration into an optimisation routine.

2.2. *Geometry Parameterisation Approach*

Blade parameterisation is based on constructing the 2D shape using an assembly of circular arcs. This 2D shape is then extruded to form a 3D turbine blade. Prior experimental studies were conducted showing that the spanwise variation of pressures are not a primary concern [8]. While this does not imply that they are insignificant, this, combined with the computational resource demands that stem from running transient simulations of each geometry, was deemed a sufficient basis to adopt the simplified 2D geometry. The blade is initially drawn at an azimuthal angle $\psi = 0$ (i.e. assuming freestream flow from left to right in Figure 2). After the blade shape is completed it can then be copied, translated, rotated and replicated to build the entire turbine (which consists of 10 blades). Initially the constraints that are imposed are that the centre point of the outer arc, $P0$, is anchored to the origin $(0, 0)$ and that the arc is symmetrical about the y-axis. Therefore, the radius (RO) and the angle (AO) are sufficient to draw the outer arc surface of the blade and obtain the points $P2$ and $P3$. The three thicknesses (TL,TC and TR) are defined to be normal to the surface of the outer arc. TR is the thickness at $P2$, TC is the thickness at the line of symmetry on the outer arc, and TL is the thickness at $P3$.

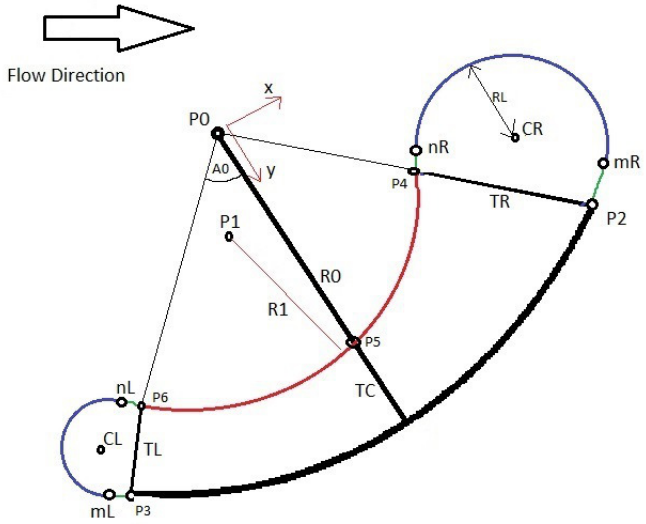


Figure 2: Blade shape parameterisation using circular arcs.

With the three thicknesses defined, the coordinates of the points $P4$, $P5$ and $P6$ can be calculated. These points

will all lie on the inner arc surface. Therefore, to be able to draw the inner arc, the remaining unknown to be calculated is the arc's center point $P1$. To do this, the system of equations (1) is solved for the co-ordinates of $P1$ using a Newton–Raphson approach [14, 15].

$$\begin{aligned} (P4(x) - P1(x))^2 + (P4(y) - P1(y))^2 - R1^2 &= 0 \\ (P5(x) - P1(x))^2 + (P5(y) - P1(y))^2 - R1^2 &= 0 \\ (P6(x) - P1(x))^2 + (P6(y) - P1(y))^2 - R1^2 &= 0 \end{aligned} \tag{1}$$

The only two points that remain to be found are the centres of the capping arcs CL and CR. The capping arcs are constructed such that the transition from the inner and outer blade arcs are tangential at the transition to the capping arcs. In order to achieve this, the coordinates of the transition points (mL, nL, mR and nR) must be computed.

The set of equations (2) is based on the requirement that the inner and outer arcs must both meet with and be tangential to the capping arcs at either end of the blade. A similar set is constructed to find the transition points at the left-hand side of the blade.

$$\begin{aligned} (CR(x) - P0(x))^2 + (CR(y) - P0(y))^2 - (RL - R0)^2 &= 0 \\ (CR(x) - P1(x))^2 + (CR(y) - P1(y))^2 - (RL - R1)^2 &= 0 \\ (mR(x) - CR(x))^2 + (mR(y) - CR(y))^2 - RL^2 &= 0 \\ (nR(x) - CR(x))^2 + (nR(y) - CR(y))^2 - RL^2 &= 0 \end{aligned} \tag{2}$$

To solve this equation system the Levenberg-Marquardt [15, 16] method was applied in MATLAB [15]. This approach, also know as 'damped least squares', is an effective method for solving non-linear least squares problems. The Levenberg-Marquardt algorithm in MATLAB interpolates between the Gauss-Newton algorithm and the method of gradient descent to curve fit under the least squares condition.

Once solved, the blade geometry is fully defined. The geometry of the blade is checked to ensure that it is within the predefined tolerances and that it is physically possible. Section 2.3 details the checking process. It is then rotated such that it is positioned at zero yaw angle before the process of replicating and rotating it to construct the entire turbine.

This approach allows the blade geometry to be constructed based on the specification of 6 parameters. These are summarised in Table 1. These six parameters then define the positions of 8 points on the blade surface (P2, P3, P4, P6, mL, nL, mR, nR) which are necessary for the geometry to be realised in the mesh generator and pre-processor. A parametric optimisation problem is now fully defined in a 6-dimensional design space.

X_1	Outer arc radius (m)	R_O
X_2	Half arc angle of outer arc (deg)	A_O
X_3	Left thickness (m)	T_L
X_4	Centre thickness (m)	T_C
X_5	Right thickness (m)	T_R
X_6	Yaw angle (deg)	Y

Table 1: Key of design variables.

2.3. Design Space Limits

Before the blade geometry can be passed to the pre-processor it must be checked to ensure that it exists within the defined parametric design space and that the geometry is physically possible.

There are a number of limits placed on the design space. Some of the limits that are imposed upon the geometry cannot be checked until all of the eight points on the blade have been calculated, which is why the geometry checking is completed at this stage. Table 2 lists the limits imposed along with the reasoning for these limits.

Limit	Reason
Blade Chord Length < 0.3× Turbine Radius	This limitation was imposed in an attempt to limit the cost of blade fabrication. Previous case studies had shown that the increase in efficiency seen through larger blades was not economically viable.
0.2 m < R_O < 40m	The significant difference was to allow for a large flat blade to be made that had a blade chord length that was approximately 30% of the turbine radius when $A_O = 1°$. Although 40 m is insufficient to entirely fulfil this criteria, it was limited to 40 m in an attempt to prevent the design space from becoming too large.
$1° < A_O < 110°$	This is imposed to ensure that self-intersection cannot occur.
0.01 m < T_L and T_R < 0.5m	These limits were set based on the cost of the blades. They were set using the experience of C-FEC Ltd and the blade shapes they had previously produced.
0.07 m < T_C < 0.5m	This limit was set based on the cost of the blades. It was set using the experience of C-FEC Ltd and the blade shapes they had previously produced.
$-90° < Y < +90°$	As it was unclear whether lift or drag would be the dominant force in creating positive torque, the blade has been allowed to yaw into positions in which it can be completely perpendicular, or parallel to the flow direction. It is unnecessary for the blades to rotate up to 180° in either direction as this effect can be simulated by swapping the blade edge thickness values.

Table 2: Limitations imposed on the design variables and the blade geometry.

In addition to this, the checking routine ensures that the blade geometry is physically possible i.e. that there are no overlapping curves and that the blade will fit within the turbine housing.

2.4. Limitations of the Parameterisation Approach

It was deemed that the parameterisation technique outlined is sufficiently flexible for this study and offers the ability to optimise a hybrid blade which draws upon high lift and drag attributes to maximise torque. It does however have two significant shortcomings for general optimisation:

1. sharp-tipped blades cannot be generated as collapsing the tip radii to a value of zero would cause a singularity. If the optimisation drives the solution towards a sharp-tipped blade, the geometry generation algorithm would require adaptation.
2. It is not possible to get a completely flat blade with this method. In the eventuality that a flat blade yielded the optimum results, it is anticipated that the optimisation algorithm would drive the solution towards very large values of the inner and outer arcs' radii.

3. Meshing

The meshing of the geometries was undertaken using the pre-processing software package FEMGV [17]. The geometry alterations throughout the optimisation process was driven by the automated generation and execution of FEMGV scripts to rebuild the geometries and regenerate meshes. This research built upon existing modelling work undertaken by C-FEC using these meshing tools where model validation has take place [7, 8]. The meshing approach focuses on enabling the modelling of the transient behaviour of the turbine. To do so a conforming sliding mesh was chosen and the mesh was split into two subdomains: a static one and a rotating one with the splitting plane illustrated in Figure 3. The main advantage of the conforming sliding mesh technique is that the use of an interpolation scheme is avoided and the time step length can be defined such that no deformation of the elements adjacent to the sliding plane is avoided thus ensuring the conservative properties of the finite volume scheme. A typical simulation involved a mixed mesh with some 140,000 elements with 52% of them located within the rotor. Three full rotations of the rotor are made during a simulation in 1440 time steps, by which time a stable flow regime had been established. All the results are then based on the data from the final rotation. The code was run on a parallel cluster at Swansea University, using 8 cores (2 INTEL Nehalem quad core processors) at a 2.93 GHz clock rate. The interconnect is a Myrinet system with a latency of 110 ns and a full duplex bandwidth of 10 Gb/s. The software was run under Linux CentOS 5.5 and the communication is managed by the MPICH

library. A typical simulation took 43 CPU-hours. The very significant overhead in communications between cores due to the split in the computational domain meant that simulation took approximately 8 hours to run.

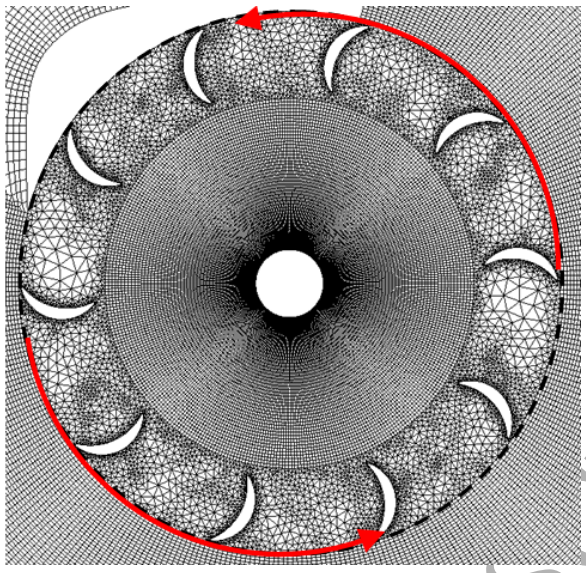


Figure 3: Location of the 'click gap' in the mesh.

4. Simulation

The simulations require the resolution of the governing conservation equations for mass and momentum,

$$\frac{\partial \rho}{\partial t} + \nabla.\left(\rho \underline{u}\right) = S_m \tag{3}$$

$$\frac{\partial}{\partial t}\left(\rho \underline{u}\right) + \nabla.\left(\rho \underline{u}\underline{u}\right) = \nabla.\left(\mu \nabla \underline{u}\right) - \nabla p + \underline{S} \tag{4}$$

in which ρ, u, μ and p are respectively the density, velocity, viscosity and pressure of the fluid. S is a term gathering the momentum source terms and S_m stands for the mass source terms. At low Mach numbers and in the absence of mass sources, equation (3) simplifies to:

$$\nabla.\left(\rho \underline{u}\right) = 0 \tag{5}$$

The CFD package PHYSICA, documented by Croft *et al.* [18], was used to solve the governing equations with a standard finite volume cell centred discretisation scheme. The solver uses the Rhie-Chow interpolation scheme [19] and the SIMPLEC method is used for pressure coupling [20, 21].

Turbulence handling for dynamic stall of blades is an area of ongoing research. Recent progress shows that the $k-\omega$ SST model by Menter [22] is a potential candidates. Maitre *et al.* and McNaughton *et al.* both showed application of the model to 2D simulations [23, 24]. Both demonstrated that the performance of the turbines were over-estimated by approximately 20% using the standard $k-\omega$ SST. Lanzafame *et al.* demonstrated good predictions using the SST transition model but admitted have to carry out an undisclosed "*long process of optimizing the local correlation variables*". In the context of an optimisation framework, the resources required to run such models within an optimisation loop are prohibitive. thus, the pragmatic choice was made to adopt an artificially high and dissipative approach to turbulence modelling, which is based on an empirical calibration of the viscosity to achieve good power coefficient predictions[7]. A value of viscosity of 200 times that of air at sea level standard was adopted throughout the domain in conjunction with a laminar handling of turbulence, thereby only modelling turbulence down to the space and time scales resolved directly by the mesh and time step respectively. This approach is known to yield mesh dependent results as the viscosity multiplier is tied to the mesh density, highlighting the need for experimental data. This calibration requirement, however, is not as complex as

the requirements of Lanzafame *et al.*, nor is it as computationally expensive, therefore making it suitable to an optimisation framework. Rolland *et al.* have shown that despite the mesh dependency owed to the turbulence resolution, the calibration is effective and is able to correctly predict performance across a range of operating conditions [7].

4.1. Boundary Conditions

Standard boundary conditions were imposed. Figure 4 shows top view of the computational geometry with labelled boundary conditions. A Dirichlet boundary condition was imposed at the inlet i.e. the free stream velocity of the wind (U_∞) was fixed.

Neumann boundary conditions were imposed on the top, bottom and sides of the domain. These conditions were symmetry conditions and impose no flux across the boundaries thus effecting an infinite domain given a sufficient distance between the device and the boundary. By using a no flux condition, it is then possible to check to see that the flow properties have returned to their freestream values at the domain edge. If it has not, then the domain is too small and some interaction may be present between the turbine and the boundary. On all the turbine's geometry (blades, housing and pylon) a no-slip condition was imposed. The use of no-slip wall boundary conditions on the blades, which rotate, required the implementation of specific code: the velocity on the blades' surfaces is calculated in cylindrical co-ordinates with respect to the rotor's axis such that the velocity imposed by the boundary condition is null relative to the blade motion. It requires recomputing in cartesian coordinates on each blade face at each time step.

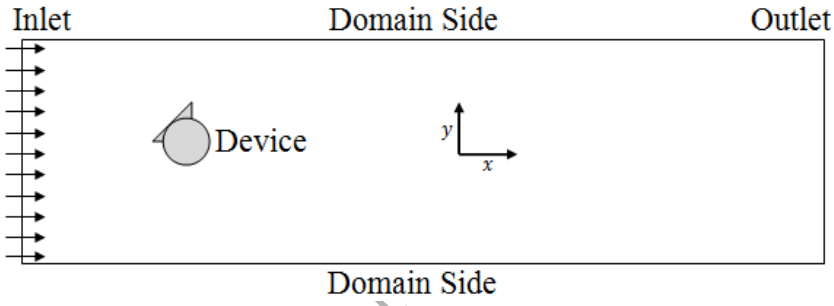


Figure 4: Simple layout of the simulation domain.

4.2. Post-Processing

To be able to perform a single fitness optimisation process, each geometry must be mapped to a single value 'fitness' or 'objective function'. This is a single value that is a measure of how good a geometry is at completing the task that it is being optimised for. In this optimisation study, the geometry fitness to be maximised is a measure of the turbine's effectiveness at generating power. The calculated fitness value is C_p.

One of the files that Physica outputs is a Comma Separated Value (.csv) file that contains the forces and torque values for each of surfaces on the turbine. The post-processor within this optimisation suite uses the torque values from this .csv file to calculate C_p. To do this the average power (P) generated in a full revolution must first be calculated. Equation (6) was used to calculate the power generated at each time step.

$$P = \sum_{i=1}^{N} T_i * \Omega \tag{6}$$

where N is the number of blades, T_i is the torque created by blade number i and Ω is the rotational velocity of the turbine.

The average power generated over one rotation ($\bar{P}$) is then calculated by taking the mean of the power generated over the last 368 time steps.

The turbine is simulated over three full rotations. However, as it can be seen from Figure 5, the torque varies significantly over the first rotation whilst the solutions begin to settle and the residuals at each time step begin to decrease. To be able to gain a better idea of the mean power output the final rotation is used to calculate $\bar{P}$. It

is expected that the torque will vary periodically in a VAWT with straight blades. This effect is primarily due to blades passing the edge of the shield and becoming exposed to the full effects of the wind.

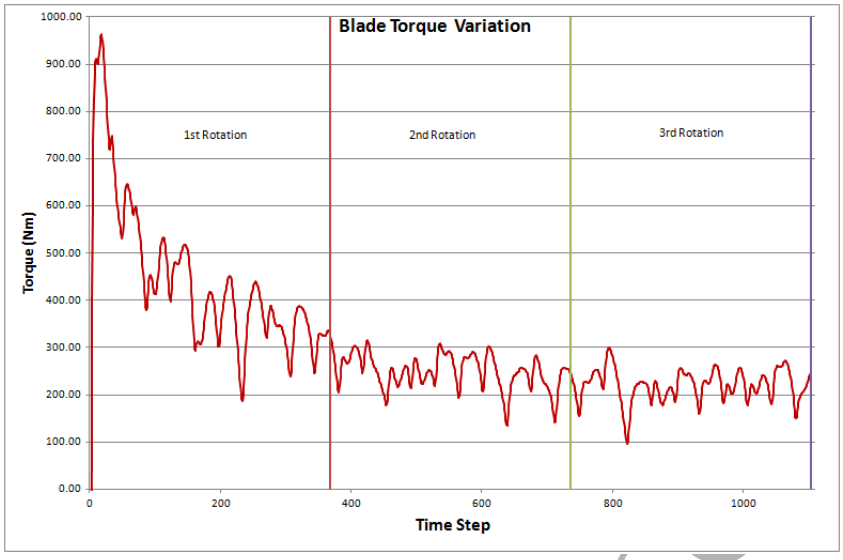


Figure 5: The variation of the total blade torque over time.

A non-dimensionalisation of the power output is undertaken by normalising the mean power, $\bar{P}$ by the maximum possible power that could be extracted from the air flow, P_{max}. The maximum power is calculated using Equation (7) in which ρ_∞ and U_∞ are the density and velocity of the air in the freestream respectively [25]. A is then the frontal area of the turbine. However, because the geometries simulated are all of height unity, the area will always reduce to the diameter of the turbine. The Coefficient of Power is then calculated using Equation (8). Note that throughout, ρ_∞ is assumed to be SLS at $1.225 kg/m^3$.

$$P_{max} = \frac{1}{2}\rho_\infty A U_\infty^3 \tag{7}$$

$$Cp = \frac{\bar{P}}{P_{max}} \tag{8}$$

It is important to note that the Betz limit [25] has not been used in the above calculations, and that P_{max} is the flux rate of kinetic energy of the wind that passes through the area that the wind turbine will occupy.

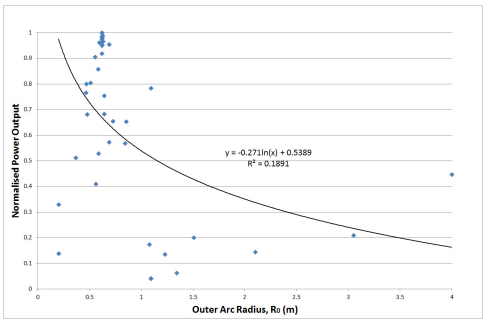
5. Optimisation Stage 1: Design Space Sampling and Response Surface Modelling

A Design of Experiments [26] sampling approach was first used to gain an understanding of the behaviour of the fitness function (Coefficient of Power) across the design space using a response surface model. This groundwork paved the way for the second phase of optimisation using the Nelder-Mead approach.

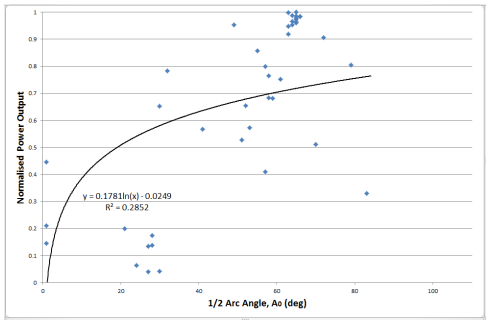
5.1. Design Space Sampling Approach

Initially the design space was sampled using Latin Hypercube Sampling (LHS). LHS is an extension of quota sampling [27] and can be generalised as being a randomised Latin square sampling technique applied over higher dimensions [28]. In LHS the design variables are unevenly divided up into n points based on the n samples being taken [28]. The completed vector of numbers that is representative of a geometry is then created by randomly assigning the points for each of the design variables into cases. This is done by observing one rule: when a point on a variable has been assigned to a case it cannot be assigned to another [28].

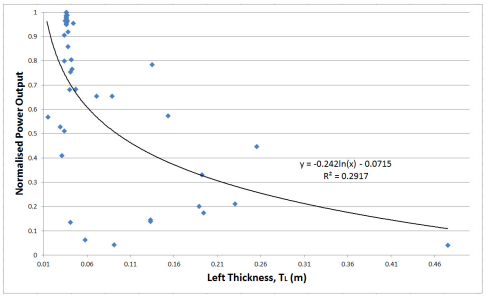
The design space was sampled 50 times using LHS. The number of samples taken was based on the estimated time to simulate each case (4 hours) and project time constraints. Geometry checking found that only 43 of the



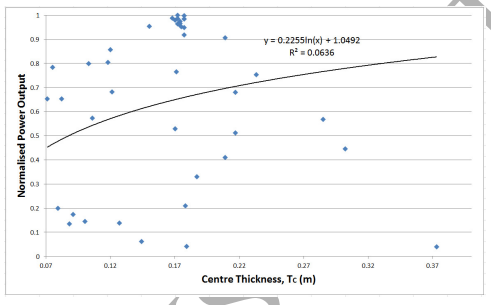
(a) Outer arc radius, R_0 (X_1) - Spearman's Rank = -0.316



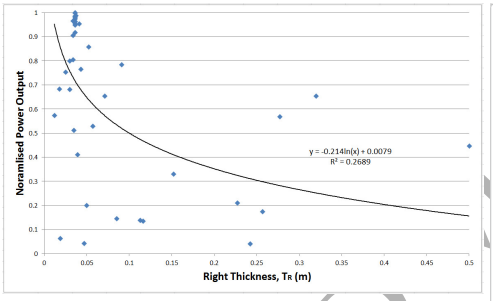
(b) Outer arc 1/2 angle, A_0 (X_2) - Spearman's Rank = -0.742



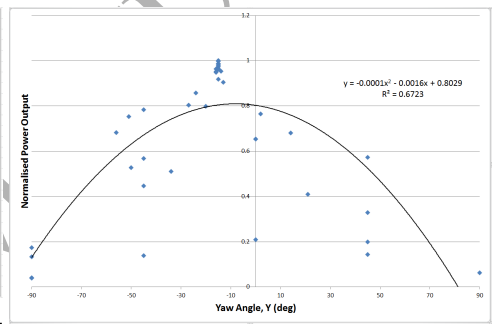
(c) Left blade thickness, T_L (X_3) - Spearman's Rank = -0.340



(d) Centre blade thickness, T_C (X_4) - Spearman's Rank = 0.165



(e) Right blade thickness, T_R (X_5) - Spearman's Rank = -0.289



(f) Blade yaw angle, Y (X_6) - Spearman's Rank = 0.036

Figure 6: Scatter plots indicating variation in normalised power coefficient as a function of design parameters. Best fit regression curves are also shown along with their associated R^2 correlation coefficient (note that the closer R^2 is to 1.0, the better the curve fit). A Spearman's Rank correlation value is also shown for each parameter.

combinations of design variables proved to be valid geometries. The most common reason for an invalid geometry was that the radius and the angle of the blade's outer arc combined to give a chord length that was larger than the originally defined limit of 30% of the rotor radius.

5.2. Design Space Sampling: Results, Statistical Analysis and Discussion

Figure 6 shows the raw data obtained from the design space sampling described. A normalised power output is plotted against the each of design parameters in each case. Note that the absolute C_p values cannot be shown due requirements to protect intellectual property. Instead a power output normalised by the maximum C_p value obtained by the sampling is shown. In each case a regression curve has been fitted to the data. Three curve types; linear, logarithmic and polynomial, were fitted for each parameter and the best (i.e. maximum R^2 correlation coefficient) in each case is displayed. In each case a Spearman's Rank correlation was also performed to give an indication of the strength of any simple rank correlations that might be present in the raw data. As can be seen in Figure 6, the strongest linear correlation exists as a negative correlation between Outer Arc Angle, A_0 and power coefficient whilst the weakest correlation between blade yaw angle and power coefficient.

By performing a statistical response surface analysis on the data more information was gained about the response to design parameters. There were 43 data points across the design space available. Therefore, it was possible to fit

the following response surface models:

- Linear model - requires 7 data points.
- Pure quadratic model - requires 13 data points.
- Quadratic interactions model - requires 22 data points.
- Full quadratic model - requires 28 data points.

The resulting output is a set of tuned coefficients (β_n) that could then be used to create a model in one of the forms shown in Equations (9) to (12), where X_n represents one of the six design variables.

$$Linear = \beta_1 + \beta_2 X_1 + \beta_3 X_2 + \beta_4 X_3 + \beta_5 X_4 + \beta_6 X_5 + \beta_7 X_6 \tag{9}$$

$$\begin{aligned} Interactions = Linear + \beta_8 X_1 X_2 + \beta_9 X_1 X_3 + \beta_{10} X_1 X_4 + \beta_{11} X_1 X_5 + \beta_{12} X_1 X_6 + ... \\ \beta_{13} X_2 X_3 + \beta_{14} X_2 X_4 + \beta_{15} X_2 X_5 + \beta_{16} X_2 X_6 + \beta_{17} X_3 X_4 + \beta_{18} X_3 X_5 + ... \\ \beta_{19} X_3 X_6 + \beta_{20} X_4 X_5 + \beta_{21} X_4 X_6 + \beta_{22} X_5 X_6 \end{aligned} \tag{10}$$

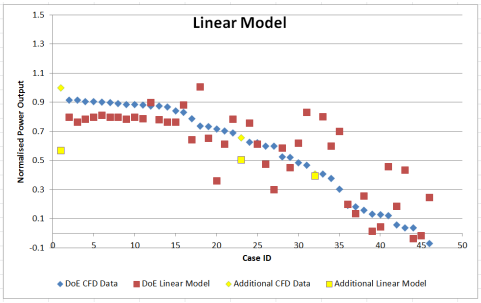
$$PureQuadratic = Linear + \beta_{23} X_1^2 + \beta_{24} X_2^2 + \beta_{25} X_3^2 + \beta_{26} X_4^2 + \beta_{27} X_5^2 + \beta_{28} X_6^2 \tag{11}$$

$$FullQuadratic = Interactions + \beta_{23} X_1^2 + \beta_{24} X_2^2 + \beta_{25} X_3^2 + \beta_{26} X_4^2 + \beta_{27} X_5^2 + \beta_{28} X_6^2 \tag{12}$$

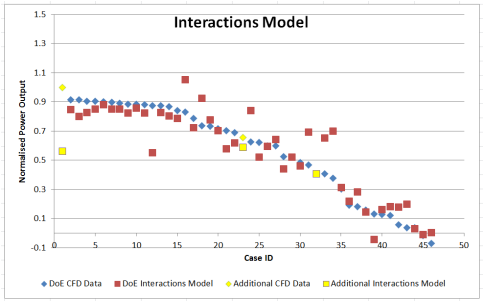
The cases were ranked based on the values obtained from the CFD simulations. The design variables for each case were then entered into all four of the prediction models. The resulting predictions were then plotted alongside the values obtained from the CFD simulations in scatter graphs. The resulting plots can be seen in Figures 7a to 7d.

By inspecting the scatter of the predicted values from the CFD data, it is clear to see that the full quadratic model is the best choice for a prediction model. To confirm this further, three additional cases were run. The cases were chosen such that there was a prediction for a poor performing, mediocre performing and a potentially new optimum geometry. This would challenge the mathematical model across the scale. These three data points are also shown in Figures 7a to 7d. The yellow diamonds are the values obtained from the CFD simulations and the yellow squares are the predicted values.

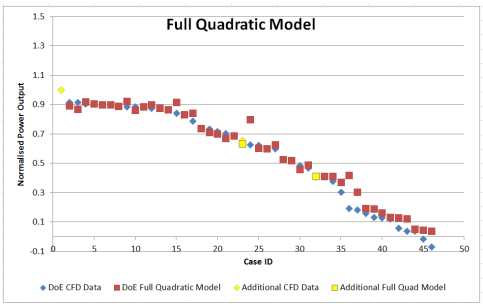
In addition to this, by finding this optimal case it can be seen that there are potential optima outside of the current design space.



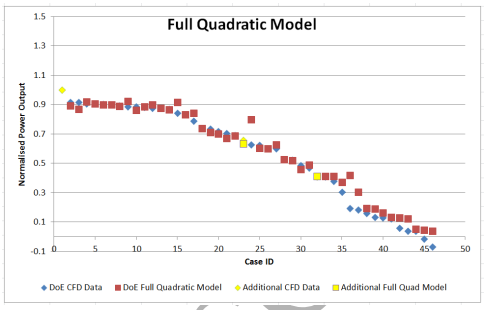
(a) Linear model comparison. RMSE = 0.190



(b) Interactions model comparison. RMSE = 0.106



(c) Pure quadratic model comparison. RMSE = 0.135



(d) Full quadratic model comparison. RMSE = 0.030

Figure 7: Comparison of response surface models showing the RMSE between the CFD results and response surface model predictions

Following this study it was clear to see that the full quadratic model was the best choice in terms of minimising the root mean square error (RMSE) between the actual CFD results and the response surface model prediction. Figure 8, created using *rstool* in MATLAB, shows how each of the design variables changes when the other five are held constant at their mean values (green solid line). The red dashed lines indicate the upped and lower 95% simultaneous confidence bands. Finally, the blue dashed lines indicate the current value of the design variables that have been entered into the prediction model.

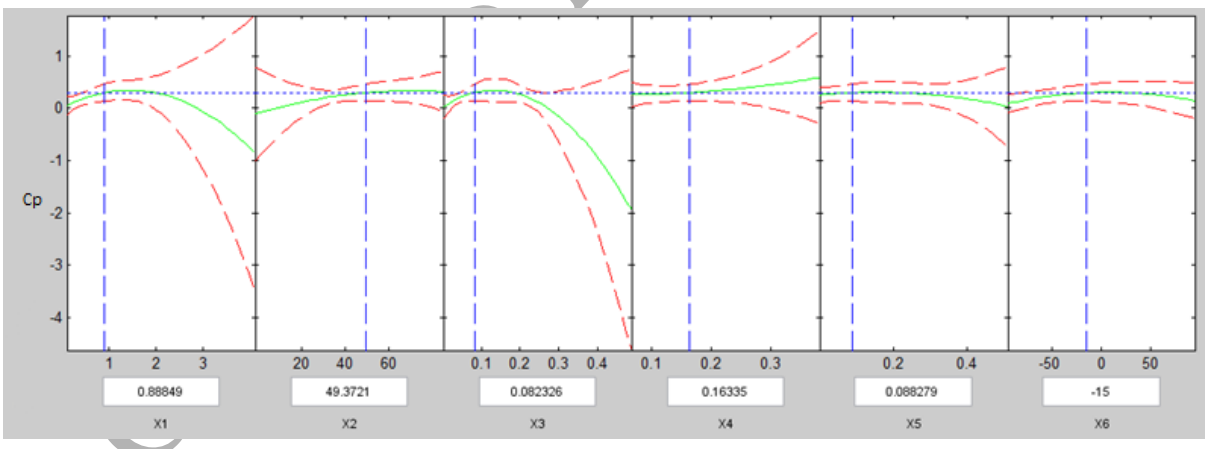


Figure 8: Full quadratic model variables.

Using the coefficients of the full quadratic model (β_n terms), it is now possible to see the relative importance of the design variables and how strongly they interact. Figure 9 shows a histogram of the top 15 most important combinations of design variables and how high the level of interaction is. Again, a key for the design variables is shown in Table 1. It should be remarked that only the magnitude of the coefficients is plotted to see the relative impacts on the model. Whether they are positive or negative is not shown here.

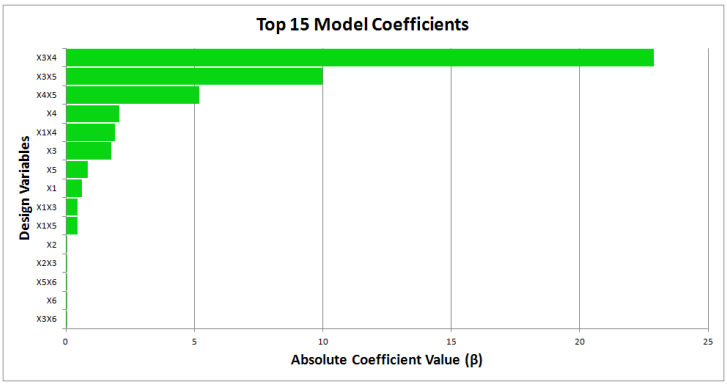


Figure 9: Levels of interaction and importance of design variables.

As it can be seen, the three highest interaction levels are the various combinations of the blade thickness parameters (T_L, T_C and T_R). There is clearly a level of importance placed on the relative differences of these parameters. Relating this result to what is known about normal aerofoil cross sections, it is clearly due to the high lift that the blades are sometimes capable of exploiting at certain azimuth angles. This indicates that the blade thicknesses will need to be carefully tuned to each other.

The next highest level of interaction between two variables is between the outer arc radius (R_O) and the centre thickness (T_C). This again can be an intuitive result if a traditional aerofoil cross section is considered. The outer arc radius has a significant impact on the blade chord length and in most of the blade shapes sampled, the center thickness has been the largest of the three. Therefore, this interaction is comparable to the ratio of an aerofoil's maximum thickness to its chord length. It is widely known that this ratio has significant impact upon the lift and drag characteristics.

The top three most important individual (linear) parameters are also the blade thickness parameters (T_L,T_C,T_R). This is again clearly related to the lift producing aspects of the blades. Following the thicknesses, the outer arc radius is the next most important individual parameter. By inspecting Figure 6a, it can be seen that the gradient of the linear line of best fit is significant. The gradient of the line in Figure 6a will not be exactly the same as the linear coefficient seen in the response surface model. This is to be expected because additional effects will come into effect through the quadratic coefficients. It is noteworthy that the gradient of the line is negative whilst Figure 9 suggests a positive coefficient. This is because Figure 9 only shows the absolute magnitudes of the coefficients.

What is interesting to observe is that the blade yaw parameter, X_6 in Figure 9) is perceived to have low importance. Another method of confirming that this result is correct, is by comparing it with Figure 6f. There is a significant amount of scatter and the gradient of the line of best fit is small when compared with the lines of best fit of the other design variables. This indicates that the impact of the yaw angle is not comparable to that of other parameters within the chosen design space.

This analysis has clearly been worthwhile. It has helped to strengthen some of the understanding behind the dominant effects that can be seen in this turbine as well as introducing new theories, some of which are initially counter-intuitive, as to the importance of the design variables. In addition, a prediction model has been produced that identified an optimum blade geometry and showed that the design space should be widened.

6. Optimisation Stage 2: Application of a Gradient-based Optimiser

The optimum blade shape identified using the design space sampling and response surface modelling detailed in section 5.2 was primarily to develop an understanding of the importance of the various parameters used to define the blade shape. This led to a 'quasi-optimum' solution that could be used as the starting point for a gradient-based optimiser to search the design space for an improved optimum that might not have been discovered by the response surface approach. The Nelder-Mead Simplex Method was used for this [29].

6.1. The Nelder-Mead Simplex Method

The algorithm that was chosen was based on the Nelder-Mead simplex method. A simplex is the generalisation of a tetrahedron to higher dimensions . A simplex in n dimensions will have $n + 1$ vertices. Each vertex represents

a solution or in this case a blade shape. When the algorithm is initialised it will first 'build' the simplex by completing $n + 1$ fitness evaluations. Once the simplex is created it then moves away from the weakest vertex. It does this by numerically evaluating the gradient of the design space and then replacing the vertex that has worst solution, with a new solution. The new solution will have been selected based on an attempt to move the vertex in the direction of the negative gradient.

An example of the Nelder-Mead algorithm in two dimensions is shown in Figure 10. The simplex in two dimensions is a triangle. The numbers represent the steps taken by the algorithm. Once the simplex is created, stage one is to evaluate the gradient, then the worst vertex is moved to a new position that is 'down the hill' and passed the other two vertices. A new simplex is created and the process repeats until the gradient is approximately zero. Note that the step size made at each stage reduces in size as the algorithm progresses.

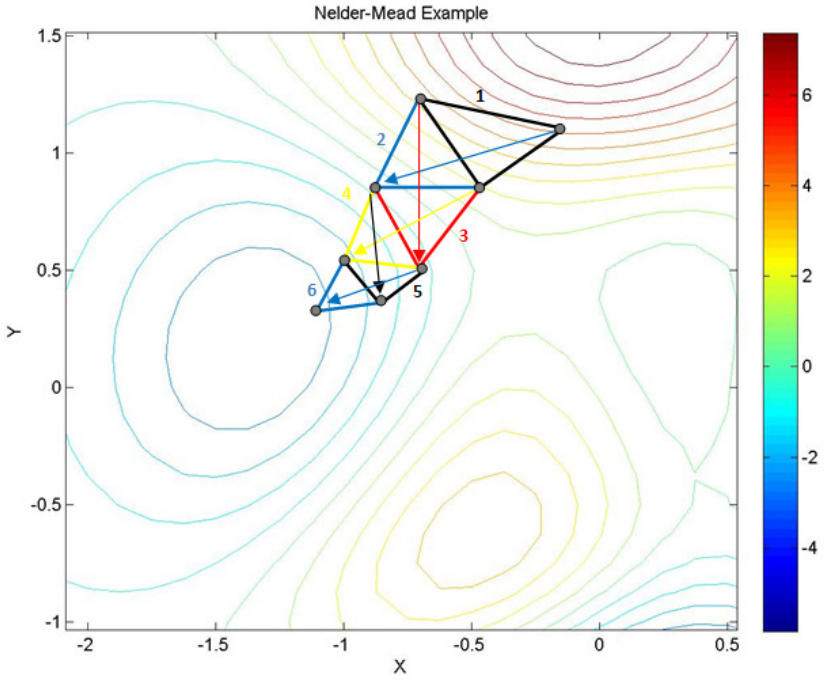


Figure 10: Example illustration of the Nelder-Mead algorithm in two-dimensions.

The Nelder-Mead algorithm is regarded as being robust when compared to other gradient-based algorithms. If the fitness evaluation stage fails at any point then that vertex can be moved to a different solution and the simplex can once again be established. This is one of the most significant advantages of using this type of algorithm in an optimisation study in which there is a relatively high chance that the fitness evaluation will fail. The greatest disadvantage with this algorithm is that it can only find a local optimum. However, as previously mentioned, because the design space was sampled and the prediction model created it has been possible to identify the areas in the design space that will most likely contain the optimum. This is not a faultless guarantee, however, it does increase the probability of finding the global optimum.

6.2. Nelder Mead Optimisation: Results and Discussion

It was decided to initiate the algorithm at a point in the design space close to the optimum found in the response surface study. This meant widening the design space to allow for larger blade chord lengths. The new upper limit imposed on the blade chord length was 40% of the turbine radius as opposed to the previously set 30%. This was done for three reasons:

1. The sampling study identified that potentially higher performing geometries lie in this region of the design space.
2. By initiating the algorithm close to, but not at the point, where the optimum is known, it would be possible to identify if the algorithm initially successfully discovers the known optimum before moving on to improved shapes. This would aid in giving confidence in the results found from the previous run.

3. It was also hoped that the automated optimiser would either converge on the newly found optimum geometry or find an even better geometry in this region. If both the response surface study and the optimiser agree on a blade shape, then it acts as a crude validation and can be more reliably regarded as an optimum.

Figure 11 shows the results from the algorithm when executed in the larger design space. Note that the power coefficient has improved by 23% compared with the geometry found following the response surface study.

It was decided to terminated the optimiser after 40 iterations (equating to 11 days of runtime) as the C_p curve appeared to be flattening at this point combined with inherent project time constraints and deadlines imposed by the partnering company, C-FEC. It is still unclear whether or not extending the runtime further would find an improved geometry but investigating this was deemed to be beyond the scope of this work.

Figure 11: Fitness values obtained over 40 iterations of the Nelder-Mead optimiser

A geometry similar to, but not exactly representing, the final optimum shape of the turbine blade is shown in Figure 12. Note that the exact optimum shape cannot be shown due to intellectual property restraints.

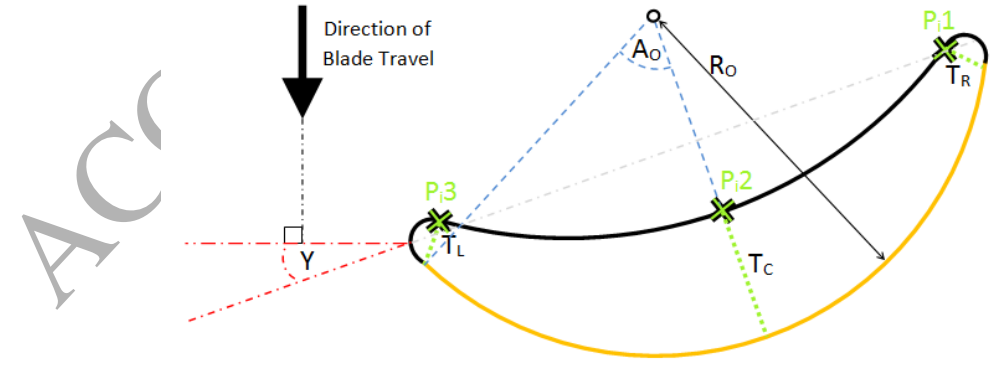


Figure 12: Representation of the optimum blade shape as obtained from 40 iterations of the Nelder-Mead optimiser

7. Conclusions

A novel approach to the blade design of a vertical axis wind turbine using hybrid aerodynamic torque generation has been presented. Parameterisation has been developed and used for computational aerodynamic optimisation using both Design of Experiments response surface and Nelder-Mead gradient-based optimisation schemes. The blade geometry resulting from this study is, at the time of writing, being used by the partner company, C-FEC Ltd for full-scale concept trials. If these trials are successful it will then enter production.

Acknowledgements

This research was partly funded by the European Social Fund (ESF) through the European Union's Convergence Programme administered by the Welsh Government. The authors would like to thank the support of the design engineers at C-FEC Ltd without whose support this research would not have been possible.

References

[1] B. F. Tu, J. Hu, Z. Q. Wang, Y. F. Wang, 3d numerical simulation and optimum design method of wind turbine, in: Proceedings of 2007 Non-Grid-Connection Wind Power Systems, 2007.
[2] N. A. Cencelli, T. W. von Backstrom, T. S. A. Denton, Aerodynamic optimisation of a small-scale wind turbine blade for low windspeed conditions, in: SACAM 2006: Fifth South African Conference on Computational and Applied Mechanics, Pts 1-3, 2006.
[3] M. Dossing, H. A. Madsen, C. Bak, Aerodynamic optimization of wind turbine rotors using a blade element momentum method with corrections for wake rotation and expansion, Wind Energy 15 (2012) 563–574.
[4] J. H. Strickland, The darrieus turbine: A performance prediction model using multiple streamtubes, Tech. Rep. SAND75-0431, Sandia Laboratories (1975).
[5] R. Gupta, A. Biswas, K. K. Sharm, Comparative study of a three-bladed savonius rotor with a combined three-bucket savonius-three-bladed darrieus rotor, Renewable Energy 33 (9) (2008) 1974–1981.
[6] K. Manohar, Hybrid lift-drag wind turbine, International Journal of Applied Engineering Research 5 (14) (2010) 2425.
[7] S. A. Rolland, W. Newton, A. J. Williams, T. N. Croft, D. T. Gethin, M. Cross, Simulations technique for the design of a vertical axis wind turbine device with experimental validation, Applied Energy 111 (2013) 1195–1203.
[8] S. A. Rolland, M. Thatcher, W. Newton, A. J. Williams, T. N. Croft, D. T. Gethin, M. Cross, Benchmark experiments for simulations of a vertical axis wind turbine, Applied Energy 111 (2013) 1183–1194.
[9] J. A. Samareh, A survey of shape parametrisation techniques, Proceedings of Langley International Forum on Aeroelasticity and Structural Dynamics (1999) 333–343.
[10] A. Nurdin, N. W. Bressloff, A. J. Keane, Shape optimisation using cad linked free-form deformation, Journal of Energy Resources Technology 116 (1183) (2012) 915–939.
[11] A. Shahroki, A. Jahangirian, Airfoil shape parametrisation for optimum navier-stokes design with genetic algorithm, Aerospace Science and Technology 11 (6) (2007) 443–450.
[12] H. U. Yong-Hai, The influence of windshield on aerodynamic performance of vawt, International Conference on Energy and Environment Technology (2009) 893–896.
[13] S. V. Ghatage, J. B. Joshi, Optimisation of vertical axis wind turbine: Cfd simulations and experimental measurements, The Canadian Journal of Chemical Engineering 90 (5) (2011) 1186–1201.
[14] T. Ypma, Historical development of the newton-raphson method, Society for Industrial and Applied Mathematics Review 37 (4) (1995) 531–551.
[15] The MathWorks Inc, Natick, Massachusetts, United States, "MATLAB and Statistics Toolbox", v7.5 r2011a Edition.
[16] D. J. Higham, Trust region algorithms and time step selection, Society for Industrial and Applied Mathematics Review 37 (1) (1999) 194–210.
[17] FEMSYS Ltd, FEMGV, release 7.0-04 Edition (August 2005).
[18] T. N. Croft, K. Pericleous, M. Cross, Physica: a multiphysics environment for complex flow processes, Numerical Methods for Laminar and Turbulent Flow 3 (1995) 12691280.
[19] C. M. Rhie, W. L. Chow, Numerical study of the turbulent flow past an airfoil with trailing edge separation, AIAA journal 21 (11) (1983) 1525–1532.
[20] S. V. Patankar, D. B. Spalding, A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows, International Journal of Heat and Mass Transfer 15 (10) (1972) 1787–1806.
[21] J. P. Van Doormal, G. D. Raithby, Enhancement of the simple method for predicting incompressible fluid flow, Numerical Heat transfer 7 (1984) 147163.
[22] F. Menter, Zonal two equation $k-\omega$ turbulence models for aerodynamics flows, in: 23rd Fluid Dynamics, Plasmadynamics, and Lasers Conference, American Institute of Aeronautics and Astronautics, 1993.
[23] T. Maître, E. Amet, C. Pellone, Modeling of the flow in a darrieus water turbine: Wall grid refinement analysis and comparison with experiments, Renewable Energy 51 (2013) 497–512.
[24] J. McNaughton, F. Billard, A. Revell, Turbulence modelling of low reynolds number flow effects around a vertical axis turbine at a range of tip-speed ratios, Journal of Fluids and Structures 47 (0) (2014) 124 – 138, special Issue on Unsteady Separation in Fluid-Structure Interaction-l. doi:http://dx.doi.org/10.1016/j.jfluidstructs.2013.12.014.
URL http://www.sciencedirect.com/science/article/pii/S0889974614000036
[25] Mechanical and Electrical Engineering Power Industry - RWE npower, Wind turbine power calculations, Tech. rep., The Royal Academy of Engineering (2010).

[26] G. E. P. Box, N. R. Draper, Aerodynamic Optimization of Wind Turbine Rotors Using a Blade Element Momentum Method with Corrections for Wake Rotation and ExpansionEmpirical model-building and response surfaces, Wiley, 1987.
[27] H. A. Steinberg, Generalized quota sampling, Nuclear Sci Engng 15 (1963) 142–145.
[28] J. C. Helton, F. J. Davies, Latin hypercube sampling and the propagation of uncertainty in analysis of complex systems, Reliability Engineering and Safety System 81 (2003) 23 – 69.
[29] J. A. Nelder, R. Mead, A simplex method for function minimization, The Computer Journal 7 (4) (1965) 308–313. doi:10.1093/comjnl/7.4.308.