Parametric Modelling Of Large Wind Turbine Blades

Paul A. Bonnet, and Geoff Dutton

Energy Research Unit – STFC Rutherford Appleton Laboratory, Didcot, Oxon, OX11 0QX

Electricity production from wind energy has grown at a fast pace over the last few years. The size of individual wind turbines has also increased significantly and it is unclear if this trend can be sustained in the future for structural reasons, especially regarding rotor blade components. A research programme is being undertaken to investigate these issues, and finite element modelling will be extensively used to examine the static and dynamic limitations of wind turbine blades. As an initial step in this research, a flexible full blade model was created and is presented in this paper. The model has been prepared in a Python script and features complete parameterisation of the geometry, mesh, materials, loads, analysis and post-processing operations. A parameter sweep functionality is also integrated in the script and some typical applications on blade structural analysis issues are presented to illustrate the capabilities of the model developed. Further details regarding the aerodynamic loading of wind turbine blades are also presented, followed by future work considerations, mainly articulated around the importance of dynamic analysis, as large wind turbine blades are becoming lighter and softer but are still subject to highly varying loads.

Symbols:

Symbols:	*M*	*turbine rotor and hub mass*
	K	*proportionality factor, depends on units used*
	D	*turbine rotor diameter*
	f	*rotor mass-diameter law exponential factor*

Keywords: Wind turbine blades, Composites, Design Optimization, Process Automation, Python Scripting.

1. Introduction

The energy problem we are facing today is articulated around two main drivers: supply and greenhouse gas emissions. Renewable energy sources are an inevitable part of the solution, and wind energy is, at the moment, the fastest growing installed production technology.

Commercial wind turbines have developed consistently in size over the last thirty years, largely for economic reasons in an attempt to reduce the electricity production cost, typically measured in p/kW.h. This is due to the fact that wind speed – and hence wind power capture – increases with altitude and that reducing the number of individual turbine units helps to reduce the overall cost of a wind farm, especially offshore. As shown in Figure 1, the largest current machine has a rated output of 5 MW and a rotor diameter of 124 m and so the question arises as to what the ultimate limits on size might be.

In all cases, wind turbine blades have pure strength requirements. A static case can, for instance, be calculated on the basis of a 50-year return period gust, while fatigue strength for a 25-year lifetime implies cycle numbers of the order of 10^7. Another crucial requirement relates to the blade stiffness, since at all times a minimum clearance must be ensured between the blade tip and the turbine tower. The increase in diameter also makes the rotor and blade mass related requirements more severe. Not only do the blade root and the rotor hub need to sustain the static loads (the 62m world-largest blade weighs ~18T), but the nacelle structure, tower and foundations also need to sustain the whole machine dynamics. For a complete discussion of design requirements, interested readers can refer to Burton *et al.* (Burton, 2001). The mass constraint is often expressed by an exponential relationship between the rotor and hub mass and the rotor diameter such as

$$M = KD^f$$

(e.g. Gardner, 2004), where the exponent f should ideally decrease for new larger wind turbine generations.

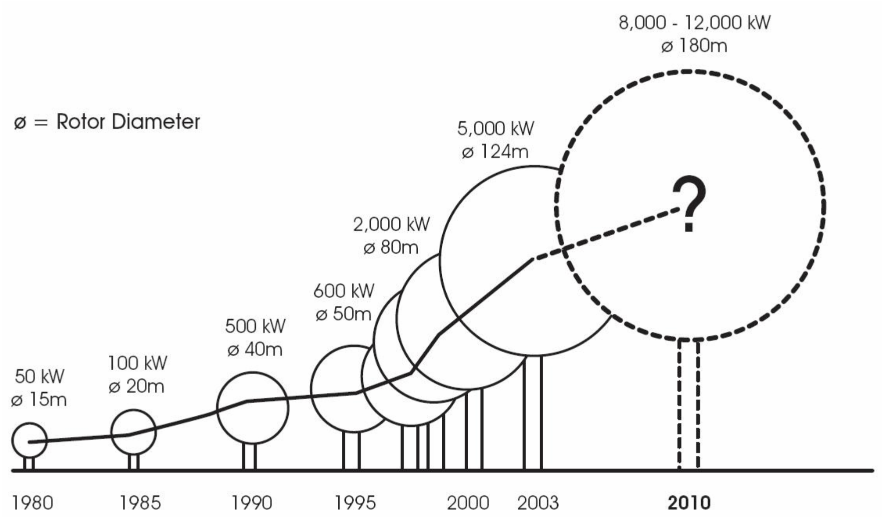

Figure 1. Wind turbine size development (source: European Commission, 2005).

To address this problem, the Energy Research Unit of STFC is funded as part of a consortium of University research groups and industrial partners under the EPSRC SuperGen Wind energy Technologies research programme. As a whole, the consortium's remit is to improve the cost effectiveness, reliability and availability of future large scale wind turbines in the UK. To conduct our research on blade structures and materials, a finite element analysis (FEA) modelling approach is used. This paper presents the methodology used to create a flexible parametric modelling platform to analyse the structure and materials of whole wind turbine blade assemblies.

Large wind turbine blades are typically manufactured with thin skins made from composite materials. Glass fibre/epoxy and wood laminates/epoxy are the most typical materials, but carbon fibre is also increasingly used. They are usually constructed from several parts glued together – compressive side, tensile side and shear webs. Their external geometry is fairly complex, made of 3D surfaces resulting from the aerofoil sections put together with varying twist angles, chord lengths and pitch axis locations. Regarding the internal structure, the manufacturing methods often result in thick adhesive joints in key structural locations and this is reflected in the models by 3D adhesive mesh elements. Finally, the materials used are highly anisotropic. The work conducted here notably takes advantage of the new capabilities of Abaqus v6.7.

The Abaqus model is defined in the Python scripting language and features a mathematical geometry definition for the blade. As described in the following section, this strategy allows the creation of a fully parameterised analysis. Automation is enabled not only for typical analyses such as mesh convergence but also for any parameter sweep optimisation, such as geometric design, material use, aerofoil section or stiffness distribution. For instance, stacking geometry analyses of the type conducted by Bechly and Clausen (Bechly, 1997) or bending-torsion coupling optimisation analyses such as in de Goeij *et al.* (de Goeij, 1999) can be easily automated. Such parametric studies are presented in section 3, as examples of possible work that can be conducted.

One key aspect in realistic structural modelling of wind turbine blades is the accuracy of the loading conditions. Section 4 describes the process followed to produce suitable aerodynamic loads, computed from a panel code simulation package specifically designed for aerofoil sections combined with an industry-standard blade element momentum theory package. Finally, section 5 presents some future developments of the model. The current implementation is static but varying loading conditions due to wind variations with height and time will be taken into account. Also, and perhaps more importantly for future generation machines, aero-elastic simulations will be enabled through the mathematical mesh interpolation.

2. Parametric blade model strategy development

2.1 Generic blade topology

Although the aerodynamic surfaces can be complex because based on non-planar surfaces, a complete blade geometry can be defined by way of a fairly simple topology adequate with our goal of obtaining a parametric model. It is based on the definition of 2D shells for the aerodynamic surfaces and the internal shear-web composites and 3D bricks for volumes of glue used to connect the sub-components.

As shown in Figure 2, the outer shape can merely be defined by consecutive sections (e.g. circle at the root and specific aerofoil profiles) and a list of distances from root to tip where each section is present. The aerodynamic shape can then be finally defined by an interpolation process creating a smooth surface through the successive curves.

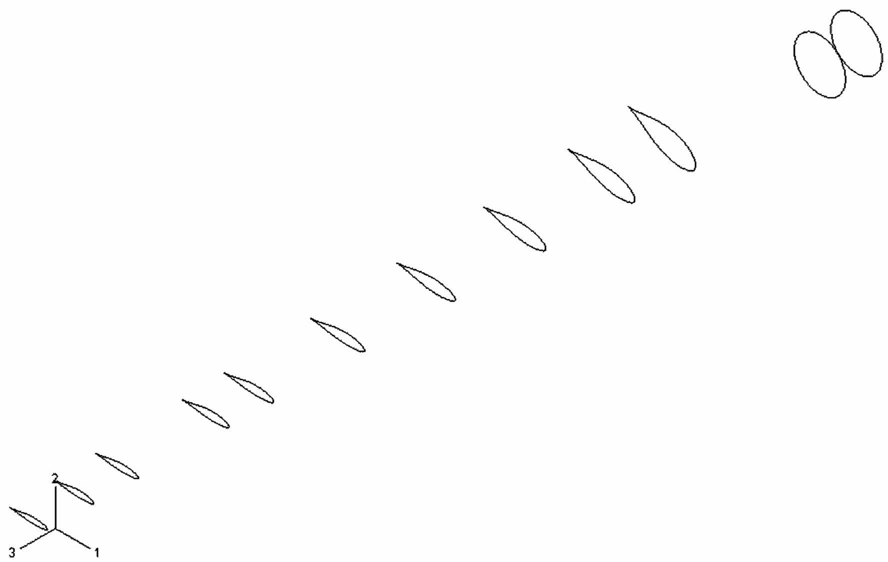

Figure 2. Outer shell topology.

Similarly, the internal structure, typically shear webs, can be parametrically defined by stipulating the distances from root where they start and end as well as the transverse ratios (defined as percentage of chord length) where they will connect to the outer skins. Finally, a glue layer can be inserted between shear-webs and skins, defined by its thickness. The shear-webs and glue volumes can be seen in Figure 4.

In this topology, it should be noted that the shear-webs may not necessarily start and end at places defined by aerofoil sections. Similarly, the various regions that receive specific composite layups may not be defined according to the longitudinal split of the blade defined by the aerofoil sections. Finally, even though Figure 2 shows a particular type of shear-web shape, only minor modifications should be needed to the model to reflect other types (e.g. I beam or closed square section). The model adopted should respect these independencies.

Because of the nature of our research project, a fully parameterised and extremely flexible modelling strategy is aimed at, so that the blade external geometry, internal structural shape, materials used, loading, analysis type and post-processing operations can be adjusted independently. In turn, the model development time will be optimised and a massive proportion of the Abaqus usage time will be spent producing the results. These specifications lead to the use of a Python script as the analysis definition core. All aspects of an analysis process can be automated in the script, and the full features of the Python programming language confer numerous options for parameterisation.

2.2 Main features of the script

The full parameterisation requirement would be difficult to meet if an external package was needed in the analysis process alongside Abaqus (e.g. CAD package for geometry definition or external mesher). On the other hand, it is well-known that the geometry definition capabilities of Abaqus are limited, especially for complex 3D surfaces such as those needed for blade aerodynamics. As a matter of fact, a lot of work was devoted to trying to create a parameterised blade geometry using the Abaqus geometry definition functions, using spline curves together with shell and solid lofts. Even though all the individual geometry items could be defined, a complete blade geometry could not be produced due to poor joining at the boundaries between the various faces (such as at the leading and trailing edges, but also at the shear-web/skin connections). The resulting mesh was also grossly inadequate, when meshable.

A radical strategy change was operated as it was decided upon that the mathematical capabilities of Python should be used to generate the geometry and mesh. A swept mesh being desirable, the list/matrix data structures that Python is articulated around are very adequate to such a task. The user enters inputs such as the chord lengths, twist angles, and aerofoil profiles for the definition of the aerodynamic surfaces of the blade. Other inputs control the definition of the internal structure, mesh size and type, materials, loads, analysis options and type and post-processing tasks.

This strategy has several interlinked benefits. No geometry entity needs to be defined in Abaqus. The whole model is defined as a function of a series of nodes and elements – the nodal coordinates are calculated from the user-defined geometry parameters and desired element sizes. The total control over the meshing algorithm enables a fully structured mesh to be created and the mesh connectivity between the various sub-components is produced automatically. To replace the presence of geometry entities used for the application of materials and boundary conditions, a list of element and node sets is defined. Again, the swept character of the mesh and the data structure in Python largely facilitate this.

A further requirement for a powerful modelling strategy is a built-in parameter sweep functionality. With the native Abaqus parametric study commands, only input file data can be the object of a study. This proscribes conducting analyses on parameters that affect the model geometry, and therefore the mesh coordinates. It was therefore decided to also code the parameter sweep function into the Python script, as an additional outer loop.

2.3 Abaqus v6.7 composite features

An Abaqus v6.7 *composite layup* can be defined (see Figure 3) as composed of plies in the same way that a *composite shell section* is in v6.6. However, because a layup can affect each ply to a specific region, a single layup can be created to define the stack of layers that applies on a whole area. It better reflects the way that composite construction layups are typically documented and is therefore a more user-friendly approach when complex composites need to be modelled. Internally, a *composite layup* is converted into *composite shell sections* before solving.

A ply-stack plot feature is also available in v6.7, allowing the easy visualisation of layups applied to the structure. Figure 11 presents the ply-stack plot of a layup used in this work. Moreover, when post-processing the results, v6.7 also allows plotting through-thickness outputs in XY plots.

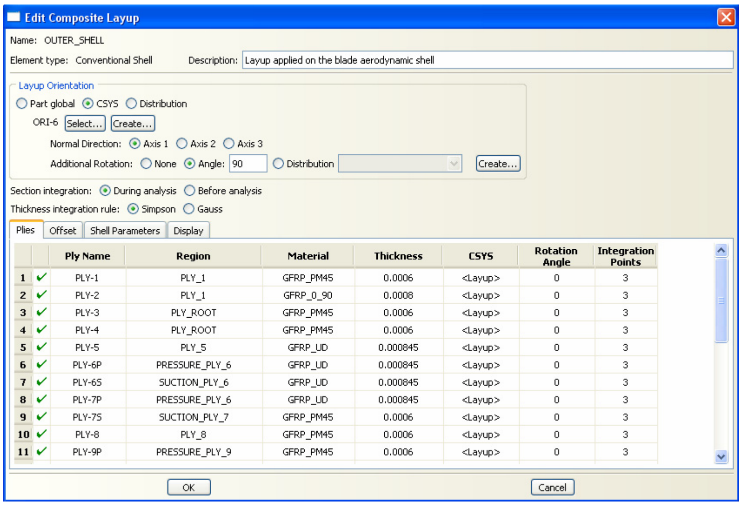

Figure 3. Composite layup editor dialog.

2.4 Other features of the script

The script also gives the user the choice of using linear (S4R and C3D8R) or quadratic elements (S8R and C3D20R). The linear and quadratic shells in Abaqus have slightly different behaviours. The linear S4R element is a general-purpose shell while the quadratic S8R features a thick-shell approximation and a constant thickness. This can be excessive in certain cases of wind turbine blade applications.

Another user parameter provides for the possibility of taking into account non-linear geometric effects. This would be necessary when large deflections occur or, for instance, in the case of modelling the centrifugal stiffening.

Even though not required at this stage of our research, damage features such as composite debonds or delaminations, together with the necessary contacts, can easily be added to the script.

Analysis post-processing is also largely automated. Key results are output in a text file (e.g. position of the blade tip, model mass, or maximum deflection) and Abaqus plots are produced and saved as TIF images. Field outputs from fields and frames can be straightforwardly created, as well as specific colour spectra for field output plotting. For instance, the variation over cyclic loading of the sum of the two surface principal stresses was post-processed as a field output and compared with the experimental thermo-elastic stress analysis measurements of a real wind turbine blade under cyclic loading.

However, automated post-processing can be very time consuming if care is not taken in the way the results are accessed. To obtain acceptable execution times, the number of accesses to the output database (ODB) file should be minimised.

3. Examples of parametric analyses

To illustrate the modelling strategy adopted, four parametric analyses are presented below. For each of these studies, once the sweep is set-up and the script execution started, no more user interaction is necessary to produce or compile the complete set of results presented. In these analyses, the blades are simply loaded at the tip in the direction of the wind, which is also the most critical loading direction given the section orientation of a wind turbine blade. The results presented do not constitute significant advances in wind turbine blade engineering, but are here to display typical results of the script model.

3.1 Mesh convergence analysis

A typical but necessary optimisation problem when conducting FEA work is that of the mesh resolution convergence. How refined does the mesh need to be? How many degrees of freedom can be solved in the time frame available? A mesh convergence analysis was conducted on a typical blade model. Five meshes of increasing resolution were produced. The meshes are shown in Figure 4 as viewed from the root, looking inside the blade – note that this view point means that the mesh elements mostly look stretched, even though they have in fact good proportions.

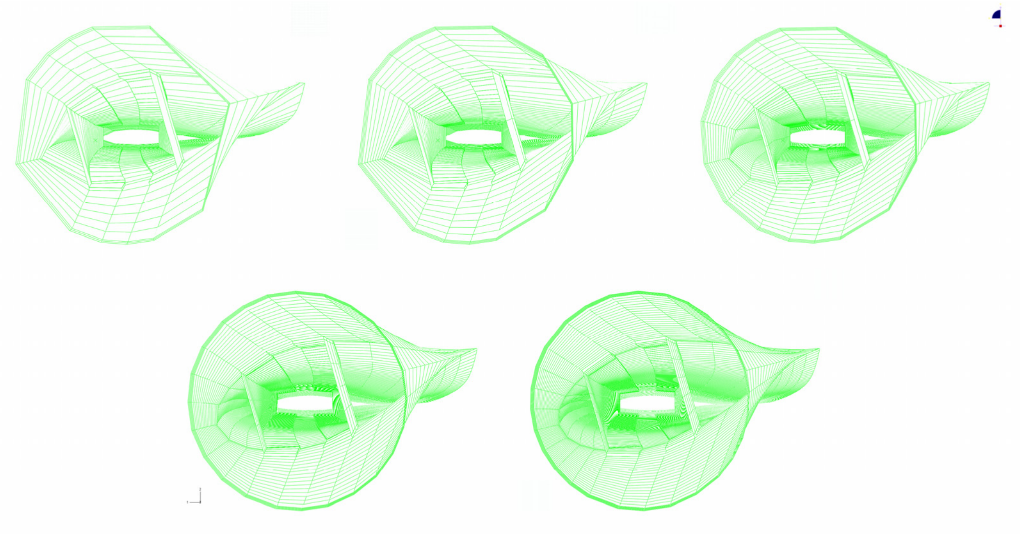

Figure 4. Meshes obtained in mesh size sweep analysis.

These meshes were analysed consecutively and the results presented in Figure 5 were obtained. With the smallest 0.03 m mesh size parameter, the mesh refinement process has converged for general stiffness analyses, as shown by the slope of the tip deflection curve on figure 5. This mesh will be used in the following sections.

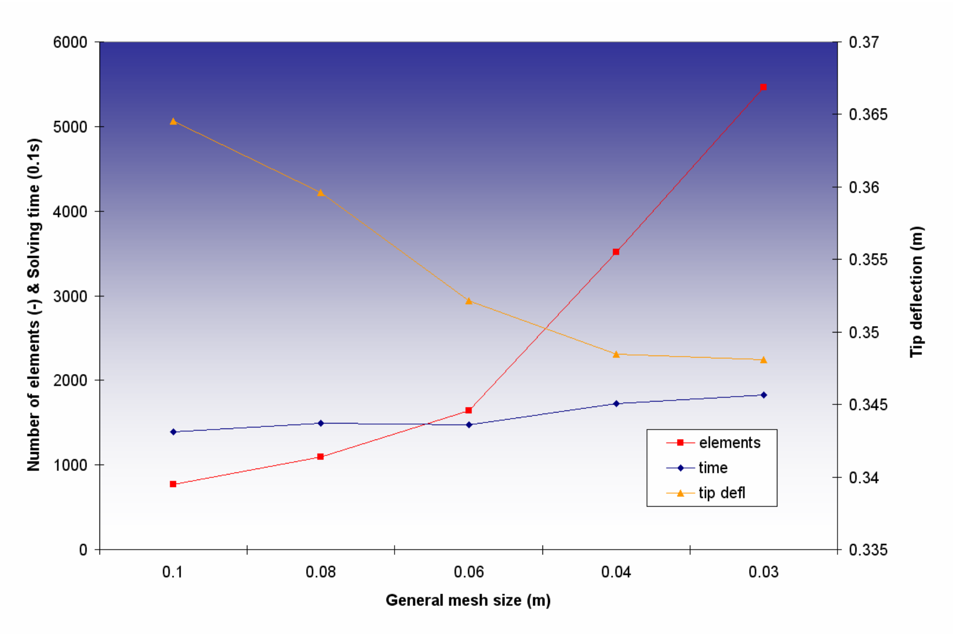

Figure 5. Mesh size sweep results.

3.2 Shear-web transverse placement

The transverse placement of the shear-webs within the aerofoil sections naturally influences the structural properties of the assembly. Due to the twist angle variations along the blade, a bending loading state always introduces torsion in some sections of the blade. Such torsion modifies the angle of attack of the aerodynamic surfaces, in turn causing a modification in loading – such a phenomenon is generally described as aero-elastic. The influence of the shear-web placement on the blade mass, bending stiffness and bending-torsion coupling is studied in this example. Six different geometries are examined as shown in Figure 6. In each one, the shear-webs are moved in opposite directions, from being very close to each other to being near the leading and trailing edges. The various geometries are analysed and the results are presented in Figure 7.

Note that, as the shear-web spacing increases, the blade mass grows significantly. This is because in the particular layup used, the composite stack applied in the outer skin in-between the two shear-webs is significantly heavier than that applied near the leading and trailing edges. In terms of blade stiffness, it is found that the increase in shear-web spacing also brings higher stiffness values, both in terms of linear and angular displacements. Although the analysis results presented here are valid, they raise the question of cross-coupling between the various design parameters, and it emerges that parameter sweeps with cross coupling of several independent parameters would be necessary in this case. Alternatively, and perhaps more comprehensively, optimisation graphs could be produced from the script where iso-parameter contours can be interpolated, improving the understanding of the interdependency between various design parameters.

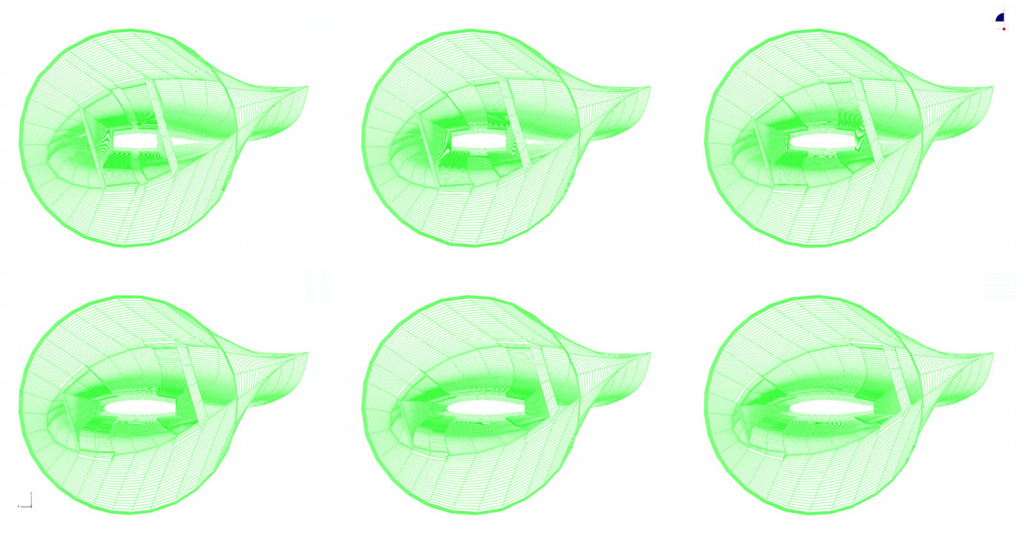

Figure 6. Meshes obtained in shear-web transverse placement analysis.

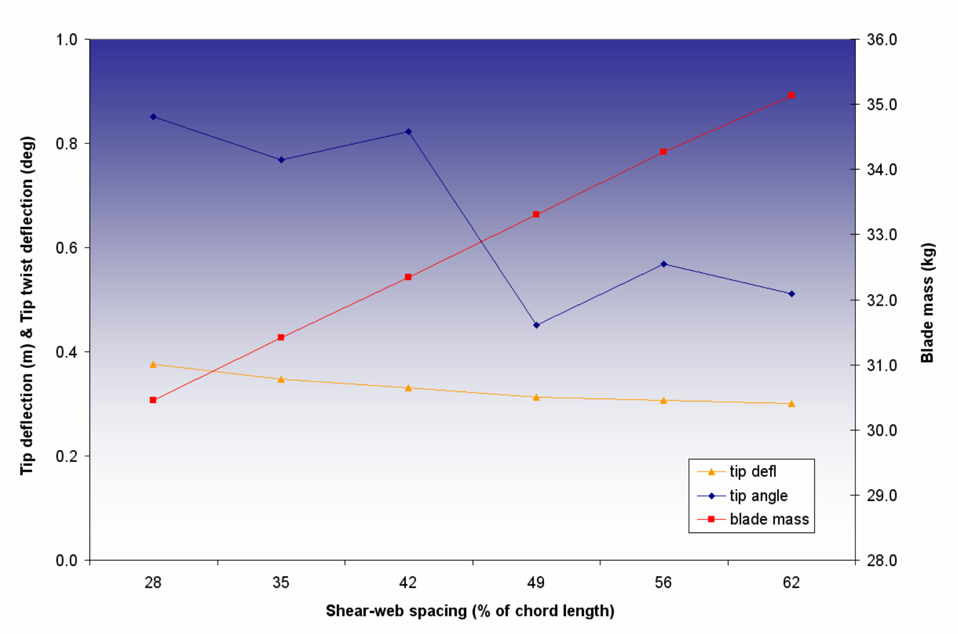

Figure 7. Shear-web placement sweep results.

3.3 Aerofoil shape

The parametric blade geometry definition also enables a study into the effect of the aerofoil shape on the structural properties of the finished component. Here, we have compared the baseline model used so far with two variations that use different profile series: the Wortmann FX84W and the Althaus AH93W (see Selig, 2006 for full coordinate listing). The blade produced with the Wortmann aerofoils uses a round root section, while a specific root profile, wider but a lot thinner, is provided in the Althaus aerofoil series – note that such a profile would bring significant construction constraints in the case of a pitch operated blade. The aerofoil coordinates are contained in plain text files (one file for each blade section), referred to from within the Python script. Figure 8 shows the meshes obtained for the three aerofoil types. These meshes show once again the validity of the mathematical geometry and mesh generation and its adaptability to substantially different designs.

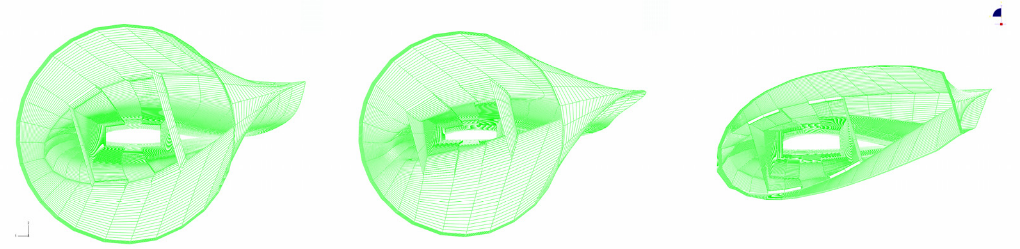

Figure 8. Meshes obtained with three different aerofoil series.

The static analysis results obtained for the three aerofoil series are displayed in Figure 9. It can be seen that a change from the baseline aerofoils to the Wortmann series, all other parameters constant, would be beneficial in terms of blade mass but detrimental in terms of stiffness. A change for the Althaus series seems very beneficial in mass terms and only a little detrimental in terms of stiffness. These results variations are mostly due to the relatively stiff and heavy sections near the root and the maximum chord area – the Wortmann series has a significantly thinner maximum chord profile while the root section of the Althaus blade has a much smaller circumference, but is still stiff enough compared to the rest of the blade, perhaps providing a better compromise.

3.4 Fibre orientation angle for passive aero-elastic tailoring

The orientation of structural fibres at an angle between 0 and 90 deg from the blade's longitudinal direction induces the creation of surface shear stresses. In the case of a circumferentially asymmetric stiffness layup, de Goeij *et al.* (de Goeij, 1999) have shown that an aero-elastic twist is produced. A parameter sweep analysis can be conducted on the composite material fibre orientation angle and the tip section angle can be post-processed to monitor the aero-elastic effect. To model this aero-elastic design effect, some of the uni-directional (UD) plies in the model had their fibre angle adjusted away from the pure longitudinal axis. The positive UD angle convention, as shown in Figure 10 for the example of +30 deg, is such that fibres go from the TE towards the LE while going from root to tip.

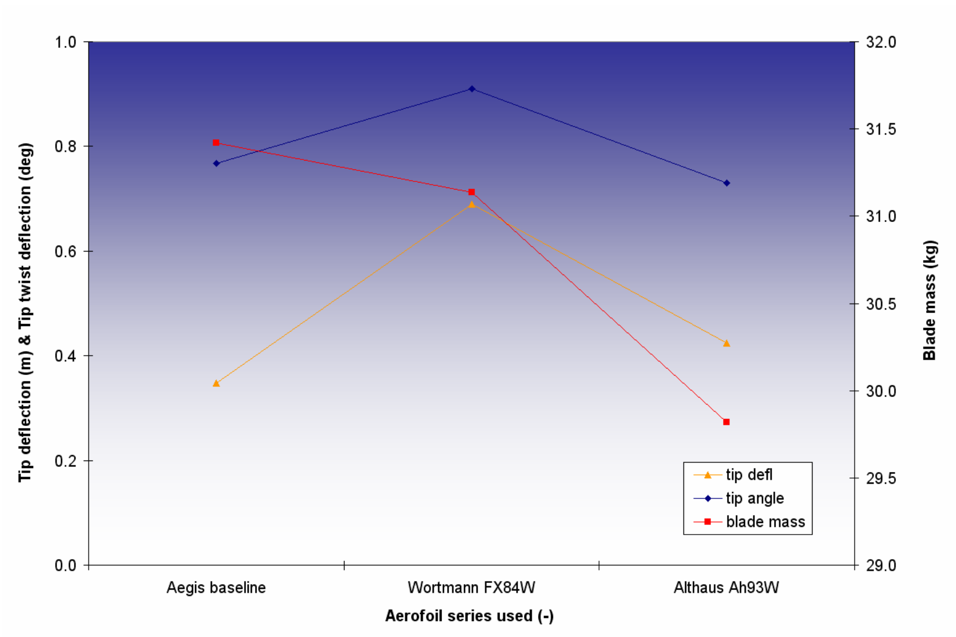

Figure 9. Aerofoil series sweep results.

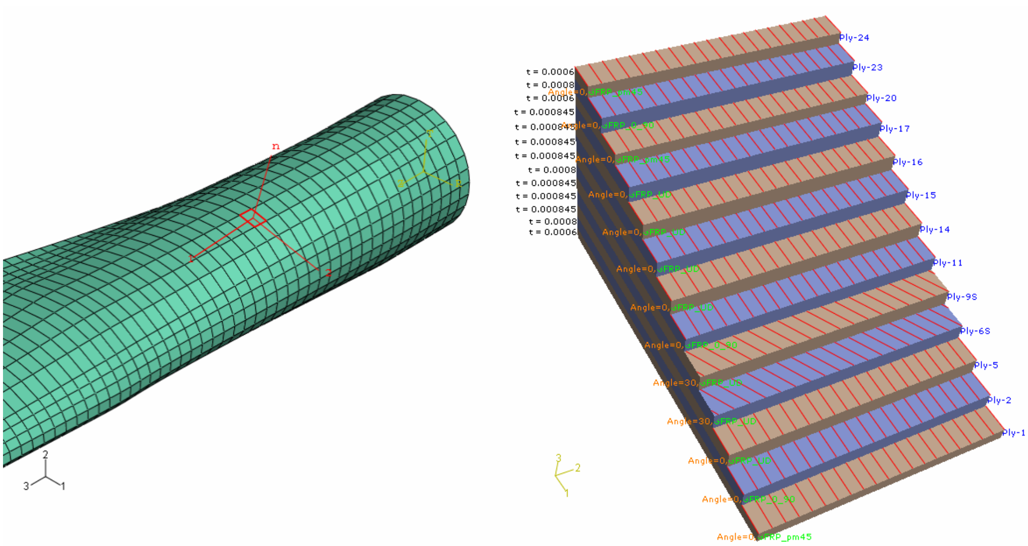

Figure 10. Layup affected by a +30 deg fibre direction shift on two local plies.

The UD fibre angle was only adjusted on four plies, all in the blade outer shell composite. Because these plies are located near the blade root, the fibre misalignment is only likely to have a subtle bending-torsion coupling impact. Still, the analysis shows that the model is sensitive enough to confirm the validity of this aero-elastic design technique – Figure 11 shows a clear correlation between the fibre alignment angle and the elastic twist at the blade tip. The aerodynamic effect, for a fibre orientation of +22.5 deg, is an angle of attack reduction at the blade tip of 0.25 deg from the value obtained with no misalignment, resulting in a passive load reduction device. This is obtained at the cost of a small stiffness reduction (also shown on Figure 11). Note however that this is only a partial validation since the change in dynamics due to the stiffness reduction should also be studied.

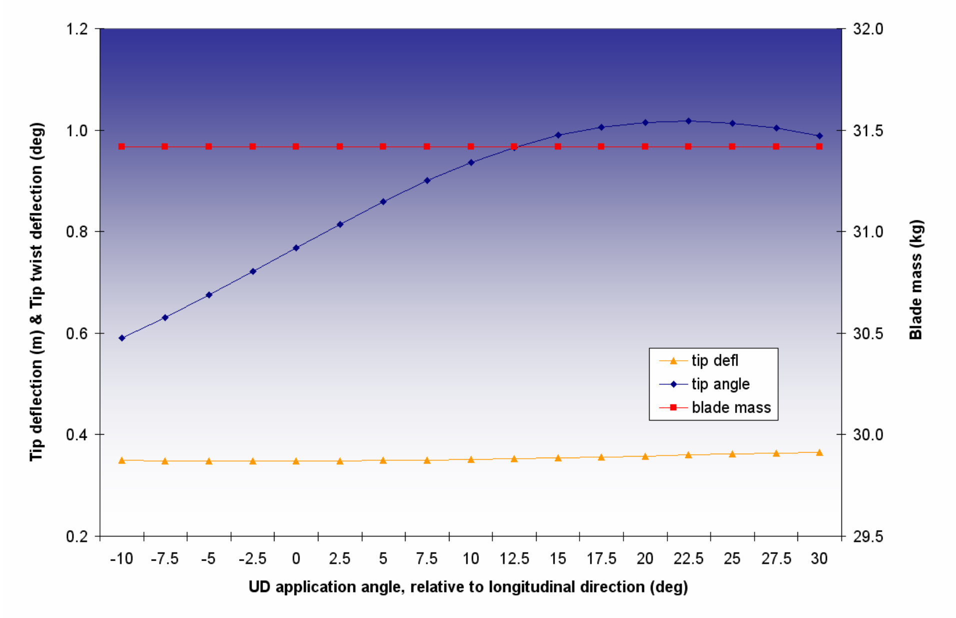

Figure 11. UD fibre application angle sweep results.

4. Blade loading

Even though this is not yet fully implemented in the parametric blade model presented in the section above, this section presents the process that will be followed to obtain accurate loading conditions, crucial for our work. The generation of a proper load case for wind turbine blades typically requires two stages – an aerofoil analysis and a rotor aerodynamics integration. The aerofoil analysis can be either experimental (using a wind tunnel) or computational and aims at providing performance data in the form of lift and drag coefficients as functions of the angle of attack. In this work, a 2D panel code solves the flow equations over an aerofoil by implementing the boundary integral method. The code used – Javafoil (Hepperle, 2005) – can either produce profile coordinates for the most common aerofoil profiles or can use data stored elsewhere, as shown in Figure 12.

The second stage of the load calculation is the rotor integration – it is necessary because the wind flow through the rotor disc is significantly affected by the presence of the rotor. Again, experimental and computational methods exist. A typical choice – the blade element momentum (BEM) theory – will be used in this research.

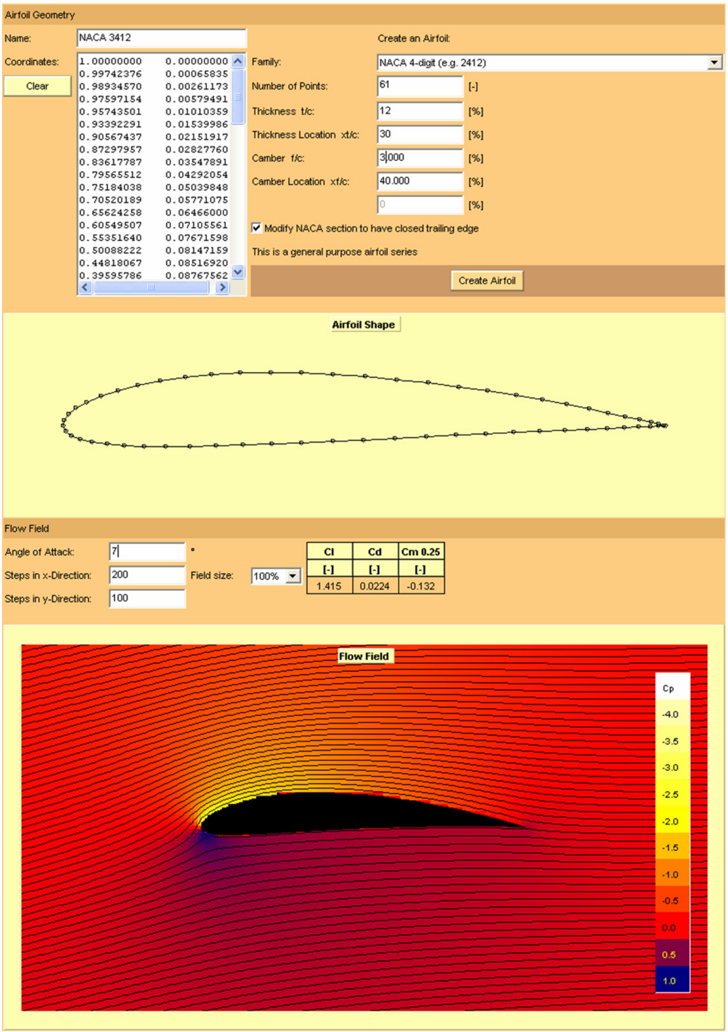

Figure 12. Aerofoil geometry and pressure distribution processed by panel code.

Once the BEM solver has produced the aerodynamic flow over the blade sections, a detailed load field over the FEA mesh can be generated by applying the pressure distribution over each section given by the aerofoil data (see Figure 12). An interpolation fitting is typically used in this process to obtain the mesh nodal loads – such an interpolation process is largely facilitated by the structured characteristic of the mesh.

5. Future developments

The future work in this research will be articulated over three main drivers. First, the blade model described in this paper needs to be developed with known and potential improvements. The blade loading, as described above, will be fully parameterised and implemented within the blade model script. This will imply that the aerofoil performance data is accessed from external files produced by the panel code tool. Later, this can be developed into an aero-elastic analysis tool by creating a feedback loop from the calculated blade deflections at various sections to the generation of an updated load field. At this stage, fully dynamic blade analyses can be conducted and the structural dynamics of whole wind turbine machines be investigated.

Also, the main composite mesh elements, currently described by 2D shells, would perhaps more accurately be modelled by 3D continuum shells. This investigation will involve a comprehensive model calibration, which can be conducted at two levels:

- On a sub-component level, in collaboration with materials research specialists within our Supergen consortium, bespoke tests of manufactured components will be carried out and reproduced computationally.
- On the whole blade level, full blade certification test data from our industrial partner collaborators will also enable crucial model validation and calibration.

And of course substantial work will be devoted in parallel to address the project questions regarding future generations of wind turbines. The current blade technology will first be replicated and potential improvements will be investigated. To this end, the material data such that collected in the Optimat Blades Project (Knowledge Centre Wind Turbine Materials and Constructions , 2006), in which the second author was an active partner, will be an invaluable knowledge source for this project.

6. Conclusions

In this paper, a generic model was presented to generate and analyse the structural behaviour of large composite wind turbine blades. The model is programmed in the Python scripting language and the analysis process is entirely automated and parameterised. Typical potential applications to wind turbine blade structural engineering issues were presented and the model showed good coherence and extreme adaptability. Special considerations were then presented regarding the generation of accurate aerodynamic loading procedures, and future developments of the model and of the current research were described.

7. References

1. Bechly, M. E., and P. D. Clausen, "Structural Design of a Composite Wind Turbine Blade Using Finite Element Analysis," Computers & Structures, vol. 63, pp. 639-646, 1997.
2. Burton, T., D. Sharpe, N. Jenkins, and E. Bossanyi, "Wind Energy Handbook," John Wiley & Sons Ltd, 2001.
3. de Goeij, W. C., M. J. L. van Tooren, and A. Beukers, "Implementation of bending-torsion coupling in the design of a wind turbine rotor blade," Applied Energy, vol. 63, pp. 191-207, 1999.
4. European Commission, "European Wind Energy at the Dawn of the 21st Century," http://ec.europa.eu/research/energy/index_en.htm, 2005.
5. Gardner, P., A. Garrad, P. Jamieson, H. Snodin, and A. Tindal, "Wind Energy – The Facts – Volume 1 – Technology," European Wind Energy Association, http://www.ewea.org, 2004.
6. Hepperle, M., "JavaFoil Users Manual," http://www.mh-aerotools.de/airfoils/javafoil.htm, 2005.
7. Knowledge Centre Wind Turbine Materials and Constructions, "Optimat Blades," http://www.wmc.eu/optimatblades.php, 2006.
8. Selig, M., "UIUC Airfoil Coordinates Database - Version 2.0," http://www.ae.uiuc.edu/m-selig/ads/coord_database.html, 2006.

8. Acknowledgments

The authors wish to acknowledge the support of the EPSRC under the Sustainable Power Generation and Supply (SUPERGEN) Wind Energy Technologies grant no. EP/D034566/1, 2006. The High Performance Computing Services Group (HPCSG), eScience Department at STFC also significantly contribute to this work by providing the SCARF cluster with Abaqus solving capabilities.