

# **Low Wind Speed Turbine Project Conceptual Design Study: Advanced Independent Pitch Control**

**July 30, 2002–July 31, 2004**

T. Olsen  
*Advanced Energy Systems, Inc.*

E. Lang  
*E3-Design*

A.C. Hansen  
*Windward Engineering*

M.C. Cheney  
*P.S. Enterprises*

G. Quandt  
*Gene Quandt Consulting*

J. VandenBosche and T. Meyer  
*Chinook Wind*



**NREL**

**National Renewable Energy Laboratory**  
1617 Cole Boulevard, Golden, Colorado 80401-3393  
303-275-3000 • [www.nrel.gov](http://www.nrel.gov)

Operated for the U.S. Department of Energy  
Office of Energy Efficiency and Renewable Energy  
by Midwest Research Institute • Battelle

Contract No. DE-AC36-99-GO10337

# Low Wind Speed Turbine Project Conceptual Design Study: Advanced Independent Pitch Control

**July 30, 2002–July 31, 2004**

T. Olsen  
*Advanced Energy Systems, Inc.*

E. Lang  
*E3-Design*

A.C. Hansen  
*Windward Engineering*

M.C. Cheney  
*P.S. Enterprises*

G. Quandt  
*Gene Quandt Consulting*

J. VandenBosche and T. Meyer  
*Chinook Wind*

NREL Technical Monitor: A.D. Wright

Prepared under Subcontract No. LAM-2-31235-02



**NREL**

**National Renewable Energy Laboratory**  
1617 Cole Boulevard, Golden, Colorado 80401-3393  
303-275-3000 • [www.nrel.gov](http://www.nrel.gov)

Operated for the U.S. Department of Energy  
Office of Energy Efficiency and Renewable Energy  
by Midwest Research Institute • Battelle

Contract No. DE-AC36-99-GO10337

## NOTICE

This report was prepared as an account of work sponsored by an agency of the United States government. Neither the United States government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States government or any agency thereof.

Available electronically at <http://www.osti.gov/bridge>

Available for a processing fee to U.S. Department of Energy and its contractors, in paper, from:

U.S. Department of Energy  
Office of Scientific and Technical Information  
P.O. Box 62  
Oak Ridge, TN 37831-0062  
phone: 865.576.8401  
fax: 865.576.5728  
email: <mailto:reports@adonis.osti.gov>

Available for sale to the public, in paper, from:

U.S. Department of Commerce  
National Technical Information Service  
5285 Port Royal Road  
Springfield, VA 22161  
phone: 800.553.6847  
fax: 703.605.6900  
email: [orders@ntis.fedworld.gov](mailto:orders@ntis.fedworld.gov)  
online ordering: <http://www.ntis.gov/ordering.htm>



**ABSTRACT**

Advanced Energy Systems (AES) Inc. conducted a conceptual study of independent pitch control using inflow angle sensors. The control strategy combined input from turbine states (rotor speed, rotor azimuth, each blade pitch) with inflow angle measurements (each blade angle of attack at station 11 of 15) to derive blade pitch demand signals. The controller reduced loads sufficiently to allow a 10% rotor extension and to reduce cost of energy 6.3%.

## **ACKNOWLEDGMENTS**

This study was performed by a team of wind energy consultants, including Tim Olsen of Advanced Energy Systems (project manager), Eric Lang of E3-Design (principal investigator), Craig Hansen of Windward Engineering, Marvin Cheney of PS Enterprises, John VandenBosche and Terrance Meyer of Chinook Wind, and Gene Quandt. Technical and management support were provided by a National Renewable Energy Laboratory (NREL) technical review team, led by Alan Wright and Dave Simms (technical monitors), with Neil Wikstrom serving as contract administrator. David Malcolm and Dayton Griffin of Global Energy Concepts also provided substantial support in establishing and understanding the WindPACT baseline turbine and its associated design and analysis tools. Funding was made possible through the NREL Low Wind Speed Turbine Project, subcontract number RAM-2-31235 (84%), and cost-sharing by the project team members (16%).

# TABLE OF CONTENTS

<b>Abstract</b> .....	<b>iii</b>
<b>ACKNOWLEDGMENTS</b> .....	<b>iv</b>
<b>List of Figures</b> .....	<b>vi</b>
<b>List of Tables</b> .....	<b>vi</b>
<b>1.0 Introduction</b> .....	<b>1</b>
<b>2.0 Concept Description</b> .....	<b>1</b>
2.1 Design Philosophy and Approach.....	1
2.2 Rationale for Major Design Features.....	1
2.3 Operational Strategies.....	2
2.4 Critical Innovations.....	3
2.5 Advantages and Disadvantages.....	3
<b>3.0 Methodology</b> .....	<b>3</b>
3.1 Design Constraints and Alternatives.....	3
3.2 Software Tool Development.....	4
3.2.1 Control System Model: SymDyn.....	4
3.2.2 Aero-Structural Model: ADAMS.....	7
3.2.3 WindPACT Design and Analysis Tools.....	13
3.3 Analysis Methodology.....	14
3.3.1 Baseline Turbine.....	14
3.3.2 Preliminary Design.....	15
3.3.3 Final Design.....	16
3.4 Assumptions and Limitations.....	16
<b>4.0 Results</b> .....	<b>17</b>
4.1 Preliminary Design: System Performance and Influential Design Parameters.....	17
4.2 Final Design: System Performance and Influential Design Parameters.....	20
4.2.1 Structural Design Results.....	21
4.2.2 Performance Results.....	25
4.3 Figure-of-Merit Impacts.....	25
4.3.1 Reliability, Maintainability, System Lifetime.....	25
4.3.2 Cost of Energy Impact.....	26
<b>5.0 Market Integration</b> .....	<b>27</b>
5.1 Enabling Technology.....	27
5.1.1 Controller Hardware.....	28
5.1.2 Angle of Attack Sensors.....	28
5.1.3 Pitch Drives.....	34
5.1.4 Backup Power for Controller and Pitch System.....	34
5.2 Performance, Costs, and Figures of Merit.....	35
5.3 Other Market Considerations.....	37
5.4 Turbine Development Plan.....	37
5.4.1 Detailed Engineering Design.....	37
5.4.2 Proof-of-Concept Testing.....	37
5.4.3 Prototype and Demonstration.....	38
5.4.4 Market Introduction.....	38
<b>6.0 Conclusions</b> .....	<b>38</b>
<b>References</b> .....	<b>40</b>
<b>Appendices</b> .....	<b>41</b>

## LIST OF FIGURES

Figure 1: SymDyn block diagram .....	7
Figure 2: Blade 1 pitch with 16 m/s wind and IEC category A turbulence.....	10
Figure 3: Blade 1 pitch rate with 16 m/s wind and IEC category A turbulence.....	10
Figure 4: Shaft speed with 16 m/s wind and IEC category A turbulence.....	11
Figure 5: Generator power with 16 m/s wind and IEC category A turbulence .....	11
Figure 6: Generator torque with 16 m/s wind and IEC category A turbulence.....	12
Figure 7: Blade 1 root edge moment with 16 m/s wind and IEC category A turbulence.....	12
Figure 8: Blade 1 root flap moment with 16 m/s wind and IEC category A turbulence .....	13
Figure 9: Blade weight .....	22
Figure 10: Flatwise stiffness.....	23
Figure 11: Edgewise stiffness.....	23
Figure 12: Torsional stiffness.....	24
Figure 13: Moment of inertia .....	24
Figure 14: Leading edge sensor booms (NREL).....	30
Figure 15: Hot wire flow sensor.....	31
Figure 16: Pressure probe sensor (NREL).....	31
Figure 17: Particle image velocimetry system .....	32
Figure 18: Laser Doppler velocimetry system .....	32
Figure 19: Rotating vane sensor.....	33

## LIST OF TABLES

Table 1: Load Cases Modeled .....	4
Table 2: Loads Summary: Preliminary Design Flexible and Rigid Tower Models .....	18
Table 3: Load Ratio Summary: Flexible and Rigid Models, AOA and PID Controllers .....	19
Table 4: Design Load Cases Producing Maximum Loads .....	20
Table 5: Baseline and Final Wind Turbine Specifications .....	21
Table 6: Strain vs. Moment Relationships .....	25
Table 7: Component Cost Impacts .....	27
Table 8: AOA Sensor Requirements .....	29
Table 9: Angle-of-Attack Sensor Options.....	30
Table 10: Baseline and Final Turbine Cost Breakdown.....	36

## **1.0 INTRODUCTION**

As the wind energy industry strives to improve its competitiveness in energy markets, the primary objective is to reduce the cost of energy (COE). To that end, the U.S. Department of Energy (DOE) and the National Renewable Energy Laboratory (NREL) have implemented the WindPACT and Low Wind Speed Turbine (LWST) programs. WindPACT's main purpose is to examine a variety of wind turbine configurations and design and logistic issues related to scaling up to 10-MW units.

The LWST program seeks to develop a new generation of techniques, components, and turbine systems to capture energy competitively at lower wind speeds found more readily near major electric load centers. Specifically, the ultimate objective of the LWST program is to produce wind turbines that generate electricity at \$0.025/kWh at sites demonstrating 5.8 m/s annual average wind speeds at 10 m above ground.

## **2.0 CONCEPT DESCRIPTION**

### ***2.1 Design Philosophy and Approach***

As wind energy technology matures, it becomes increasingly difficult to identify areas of improvement that will further reduce the COE. This project's objective was to reduce system loads through advanced blade pitch control techniques and then increase rotor area and energy capture accordingly, for a net reduction in the life cycle of the COE. The state-space control system receives measurements of rotor azimuth position, rotor speed, and angle of attack (AOA) sensor inputs and then controls each blade pitch independently according to a weighting function that optimizes power output and load reduction throughout the system. AOA sensing, combined with pitch control, are of particular interest because the AOA provides information that is highly correlated with the aerodynamic loads on the turbine.

The general design approach is to reduce loads for a given turbine design through advanced independent blade pitch control and then take advantage of the excess design margins by increasing blade length to the extent possible while staying within the original loads envelope.

Alternative approaches include allowing the rated power of the turbine to increase without increasing the blade length or reducing the COE by reducing component sizes, thereby reducing turbine capital cost.

### ***2.2 Rationale for Major Design Features***

The new design features inherent to this project are independent blade pitch, state-space control, AOA sensors, lengthened blades, and a backup power system. Each blade experiences different wind, so each blade should pitch independently to optimize the resulting structural loads on that blade and the total turbine system. A robust way to control blade pitch is to use a state-space controller. This requires a complex algorithm, but turbine controllers should be capable of handling the complexity and requisite processing.

AOA sensors provide the earliest information available for control decisions, and that information lies at the source of loading for each blade. Instead of waiting for the changing wind conditions to create disturbing forces on the blades and turbine and for these loads to cause deformation and vibration, the AOA sensors provide immediate information that can be used by the control system to determine blade pitches that will counteract the effect of the changing wind conditions.



This AOA sensor approach requires new sensors and appropriate pitch actuators, as well as a backup power system to ensure controller operation during grid outage and other fault conditions. It is unlikely to be able to directly measure the true aerodynamic angle of attack on a wind turbine blade. However, there are many ways to measure the inflow angle immediately ahead of a given blade station. This angle is equally suitable for control purposes because load fluctuations will be highly dependent on this angle. In this report, we will refer to the AOA, but we are referring to this measured inflow angle as a surrogate for the AOA.

Finally, the blades can be lengthened to increase the rotor-swept area and energy capture without significantly increasing other component sizes or costs. This approach was chosen over component size reduction for logistics purposes: It allows blade redesign, rather than the redesign of all the other components to be the focus.

Lengthening the blade raises questions about the approach to extension. Full scaling in all dimensions would maintain the original aerodynamic plan form, but it would improve structural properties and increase blade weight unnecessarily, and then require significant structural redesign. Stretching the blade lengthwise would retain original section properties at scaled-up radial locations, requiring no substantial redesign work but modifying the original aerodynamic plan form. In addition, stretching the blade would impact only the length aspect of blade shipping, whereas full scaling would increase height and girth, which are already pushing shipping limitations on a 1.5-MW turbine. We chose to stretch the blade in length only, after demonstrating an insignificant change in the aerodynamic power coefficient.

### **2.3 Operational Strategies**

The primary operational strategy for this project is to produce blade pitch commands that maximize energy capture and minimize structural loads. In collective pitch control, the blades are pitched in unison. This can be effective in controlling rotor speed and responding to changes in wind speed. In this study, we explored independent blade pitch control. Independent pitch control allows the turbine to respond to asymmetric loads such as vertical wind shear or changes in wind direction. Independent blade pitch control also allows the vibration of each blade to be reduced through dynamic pitch angle adjustments. In general, individual blade vibration cannot be addressed through collective pitch schemes.

For the power curve in Region II (above cut-in and below rated wind speeds), the objective is to maximize the power coefficient ( $C_p$ ). This was accomplished by using a variable-speed operation in which the rotor speed increases with increasing wind speed, maintaining a nearly constant tip speed ratio. Essentially, the generator applies a variable torque that is a function of rotor speed. This strategy is similar to the strategy for the baseline WindPACT turbine. In this study, the improved pitch controller reduced loads, which in turn allowed the blade length to increase. The longer blades allow more energy capture in Region II. Additionally, rated power is reached at a lower wind speed.

For the power curve in Region III (above rated and below cut-out wind speeds), the objective is to independently vary the pitch of each blade to maintain a constant rotor speed and to minimize blade-flapping motion, and thereby reduce turbine loads. In Region III, the generator maintains a constant torque and allows rotor speed to vary in response to changes in rotor power. Since rotor power is the product of torque and angular velocity, controlling rotor speed maintains rotor power at a rated power.

The strategy for wind speeds above cut-out is for each blade to pitch independently so as to minimize the angle of attack variations at blade station number 11 (of 15).

## **2.4 Critical Innovations**

This project relies on several critical innovations that are developed in further detail in Sections 3.2.1 and 5.1.2. They include AOA measurement and feedback, a state-space controller that utilizes a linearized turbine model with the AOA information to make blade-pitch decisions, and tuning of the control parameters and input weighting.

## **2.5 Advantages and Disadvantages**

This project's advanced control concept has a number of advantages: smart loads control and reduced structural loads, increased blade length and rotor area for low-wind-speed sites, higher annual energy production (AEP), and lower COE.

Disadvantages include complex controller design and tuning and a reduced failure tolerance. This study's approach includes a backup power system to mitigate the reduced failure tolerance. This system may actually improve overall system robustness, making it a net advantage. These issues will be discussed in greater detail.

## **3.0 METHODOLOGY**

### **3.1 Design Constraints and Alternatives**

Design studies with numerous parameters face a significant challenge of maintaining experimental control to ensure fair comparison of alternatives. After substantial discussion about consistency in comparisons and concerns with the WindPACT tools, the team decided on the following design constraints:

1. Hold to the WindPACT baseline design tools and analysis procedures
2. Constrain rated power to the baseline 1.5 MW
3. Constrain maximum tip speed to the baseline 75 m/s (results in slower rated rotor speed for longer blades)
4. Hold hub height to the baseline 84 m
5. Allow only minor alterations to the aerodynamic design, retaining the maximum chord, the same airfoil distribution, and keeping  $C_{pmax}$  within 0.5% of the baseline value
6. Model the WindPACT subset of the IEC [1] load cases (Table 1) and address fault conditions as appropriate
7. Require component design margins similar to the baseline.

**Table 1: Load Cases Modeled**

Type of Load	Acronym	Mean or Initial Wind Speeds (m/s)	Directions	Return Periods (Years)
Normal turbulence model, IEC class 2a, Kaimal spectrum	NTM	8, 12, 16, 20, 24 (@ 3 seeds each)	N/A	N/A
Extreme coherent gust with direction change	ECD	12, 16, 20, 24	Positive, negative	N/A
Extreme coherent gust	ECG	12, 16, 20, 24	N/A	N/A
Extreme direction change	EDC	12, 16, 20, 24	Positive, negative	1, 50 years
Extreme operating gust	EOG	12, 16, 20, 24	N/A	1, 50 years
Extreme vertical wind shear	EWSV	12, 16, 20, 24	N/A	N/A
Extreme horizontal wind shear	EWSH	12, 16, 20, 24	Positive, negative	N/A
Extreme wind model	EWM	42.5 (@ 5 seeds)	Turbulent	50
Emergency stops	E-stop	12, 16, 20, 24	N/A	N/A

### 3.2 Software Tool Development

The project's primary tools include revised SymDyn on the MATLAB/Simulink platform, ADAMS with control matrices inserted from SymDyn, and revised WindPACT design and analysis tables, with the addition of load case identification.

#### 3.2.1 Control System Model: SymDyn

SymDyn [2] is a computer-based toolset for creating dynamic models of wind turbines and for developing control systems for wind turbines. This section discusses the application of SymDyn for controller design and modeling; a history of SymDyn's development is available in Appendix A.1. Input data files used by SymDyn are included in Appendices A.2, A.3, and A.4.

Creating Dynamic Models Using SymDyn creates a limited degree of freedom (DOF), nonlinear dynamic model of a wind turbine. For this study, the active DOFs were rotor azimuth (rotor speed) and blade flap on each of the three blades, for a total of four DOFs. In contrast, modeling a wind turbine in ADAMS uses many more DOFs. A higher number of DOFs is useful during simulations. However, a low DOF model is better for developing a control system. As the number of DOFs increases, the controller rapidly becomes more complex and numerically difficult or impossible. SymDyn calculates a periodic, linearized, state-space model from the nonlinear model. This concept will be developed in greater detail.

First, it should be recognized that the dynamic equations of a wind turbine are periodic. In other words, the coefficients in the coupled system of dynamic equations of motion are not constant. Rather, the coefficients are a function of the azimuth of the rotor. This may be more obvious for a two-bladed turbine than for a three-bladed turbine. In a two-bladed turbine, the moment of inertia of the rotor about the yaw axis is a function of the azimuth. In a three-bladed turbine with shaft tilt, gravity loads are periodic; thus the dynamic equations are periodic. Also, if asymmetrical wind loads such as wind shear exist, the equations are also periodic. In this study, the periodic coefficients were approximated by calculating the coefficients at many azimuthal positions of the rotor. Typically, 200 sets of matrices were calculated for a single wind speed. The coefficients

were also a function of hub height or average horizontal wind speed. Thus, the coefficients were calculated at 12, 14, 16, 18, 20, 22, 24, and 26 m/s for each of 200 azimuth positions.

Second, linear control theory is a well-developed topic while nonlinear control theory is less developed and difficult to implement. Thus, for the purpose of creating the control system, the nonlinear dynamic equations are approximated by linear equations, and linear control theory is applied. However, for the simulations, the nonlinear equations were used to represent the wind turbine. Only the controller and estimator equations were linearized.

Third, most wind turbine control systems to date have used classical control theories such as Proportional-Integral-Derivative (PID) controllers. Although these classical methods have been successful, they are also limited and problematic when extended to consider multiple controlled variables, such as controlling tower vibration, rotor speed, and blade vibration simultaneously. In “modern” control theory, the dynamic equations of motions are written as a system of coupled first-order equations. Each of the variables in the system of equations is called a state. Thus, the four DOFs of azimuth and blade flap on each of three blades become eight states: rotor azimuth, rotor angular velocity, three blade flap positions, and three blade-flapping angular velocities.

SymDyn relies on AeroDyn [3] subroutines for the aerodynamic calculations. The possible wind field inputs are the same as for AeroDyn: various combinations of steady and/or ramped wind fields described by hub-height wind speed, vertical and horizontal wind shear, vertical wind speed, and horizontal wind direction. AeroDyn also supports the use of full-field turbulence, which is a more realistic representation of the wind as a complex, time-varying, and spatially varying input function.

Part of the philosophy of this study is that a better controller is possible if the complex wind field is approximated in the control system by a few well-chosen parameters and the controller uses this approximate wind field to choose better blade pitch commands. Specifically, the bulk of this study focuses on approximating the incoming wind using three time-varying parameters: hub-height horizontal wind speed, linear vertical wind shear, and linear horizontal wind shear. The wind fields that are used in the simulations are not limited to this parameterization; only the controller’s representation of the wind field is limited to this parameterization. The actual wind fields used for the simulations vary depending on the purpose of the simulation, but they include full-field turbulence in most cases.

There are many other possible parameterizations of the incoming wind field. Specifically, it would be possible to add vertical wind and horizontal wind direction. However, it was found that vertical wind introduces perturbations very similar to horizontal wind shear and that horizontal wind direction was mostly redundant to vertical wind shear. In fact, including “redundant” parameters made the control system less effective. Different wind field parameterization methods are possible, such as a grid of wind speed values in the rotor plane with an appropriate interpolation scheme. This later method of wind field parameterization was not investigated because it would have required a major rewrite of SymDyn and its interface with AeroDyn and because the simple three parameters of hub-height wind speed, linear vertical shear, and linear horizontal shear worked well.

These wind parameters were added to the dynamic model of the wind turbine as “disturbance” states [2]. Thus, the dynamic equations used eight model states plus three wind disturbance states (11 states total).

**Creating Control Systems Using SymDyn:** Once SymDyn is used to create the periodic, linearized, state-space model from the nonlinear model, the next step is to create the control system. One of the major purposes of this study was to investigate the use of sensors capable of measuring the wind as inputs to the control system. Specifically, the use of AOA or angle of incidence sensors located on the blades was investigated. Another major purpose of this study was to investigate the effect of independent pitch control on turbine performance. The control system consists of a periodic state estimator and a periodic controller with disturbance accommodation. This concept will be developed in greater detail.

If the state of the turbine and the wind were (somehow) known directly without errors or time delays, there would be no need for an estimator. However, in most cases, the time varying values of eight turbine states and three wind states (see above) are not known and must be estimated before the controller can calculate the desired pitch angles. Part of the control system, called the “estimator,” uses measurements of the rotor speed, the azimuth, and the three AOA sensors (one on each blade at 73% span) to estimate the states of the wind turbine and the wind disturbances. The controller uses these estimates to determine the desired blade pitches.

The estimator is a periodic linear system of dynamic equations. It is periodic because the dynamic state-space model of the turbine was periodic and because the relationship between the AOA sensors and the wind disturbance states is periodic. This second point can be understood by considering how the AOA would change at the 70% span position of a blade as a function of azimuth even for the case of a rigid turbine in steady wind with a constant vertical wind shear and no pitching action. The AOA would change as the blade moved because the incoming wind speed would change depending on the azimuth of the blade.

In this study, the periodic equations for the estimator were approximated by calculating the coefficients of the equations at many points in the 360 degrees of possible rotor azimuth positions. Typically, 200 sets of matrices were calculated for a single wind speed. The coefficients were also a function of hub height or average horizontal wind speed. Thus, the coefficients were calculated at 12, 14, 16, 18, 20, 22, 24, and 26 m/s for each of 200 azimuth positions.

The controller uses the estimates of the turbine states and the wind states to determine the best blade pitch commands. The linearized dynamic model of the turbine is periodic, so the controller should also be periodic. An easy way to understand this is to consider the pitch commands needed to correct for a vertical wind shear. Clearly, the amount of pitch for a given blade depends (in part) upon the azimuth of that blade. When the blade points up, the blade pitch increases; when the blade points down, the blade pitch decreases.

One of the features of the controller used in this study is the disturbance accommodating control (DAC) [2]. In a control system without DAC, the controller calculates the pitch angles that minimize the deviation of the turbine states from a chosen operating point. This means that the controller uses pitch angle commands to try to maintain the proper rotor speed and to minimize blade flapping. However, without DAC, the controller does not make use of the estimated wind field contained in the parameterized wind estimates. In a control system with DAC, the controller uses the estimated wind shears (vertical and horizontal), horizontal wind speed, and the estimated turbine states to come up with an improved set of commanded blade pitch angles.

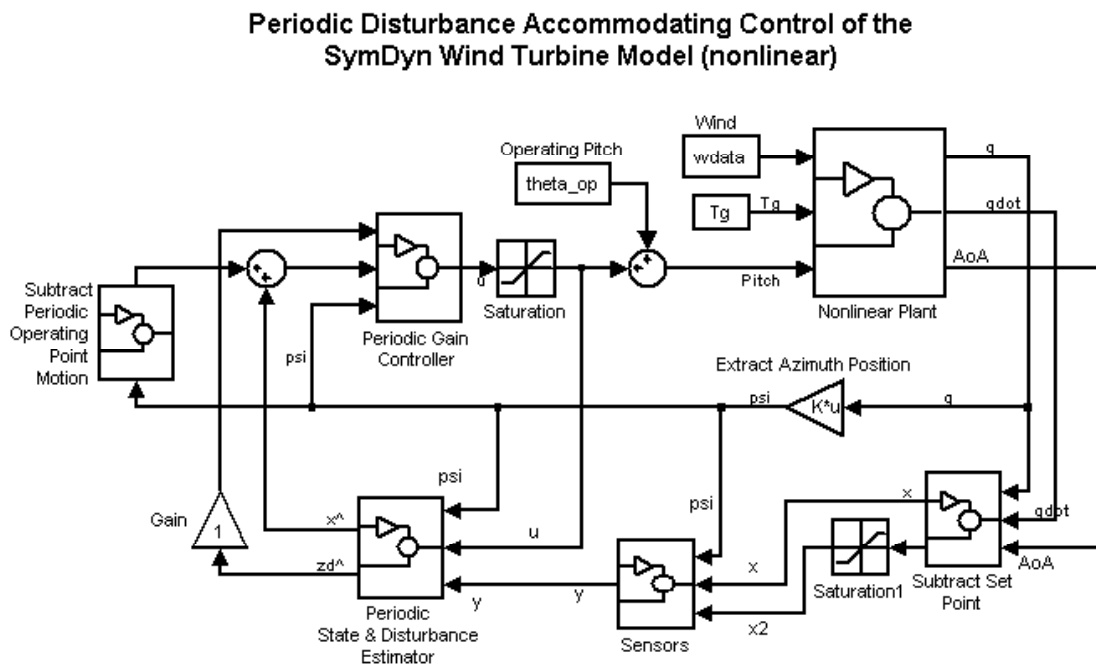
**SymDyn Block Diagram:** Figure 1 depicts a control block diagram. Starting in the upper right corner, there is a block labeled Nonlinear Plant. The inputs to this block are the wind data, the generator torque, and the pitch angles of each blade. The outputs are the flapping angular

positions and velocities of the blades and azimuth and rotor speed, as well as the angles of attack for each station of each blade. The outputs are calculated from the inputs by integrating the nonlinear equations of motion of the wind turbine and by using calls to AeroDyn to calculate the aerodynamic loads.

The next block, called **Subtract Set Point**, subtracts the steady state operating conditions from the output of the previous block. The outputs are the deviation of the state variables from the periodic steady state operating point. The AOA data pass through a saturation-limiting filter to prevent unrealistic angles of attack from propagating through the system at startup.

The next block, called **Sensors**, simulates the sensors available to the control system. This flexible block allows the number and type of sensor to be varied. After debugging, this block remains fixed. It provides rotor speed, rotor azimuth, and AOA values at station 11 of 15 for each blade to the next block, called **Periodic State & Disturbance Estimator**.

The **Periodic State & Disturbance Estimator** block uses sensor outputs and pitch angle for each blade to estimate the state of the wind and the state of the turbine. These estimates are used in the next block, called **Periodic Gain Controller**, which calculates the optimal pitch angles based on the state estimates. The pitch commands are filtered to ensure the pitch angles and pitch rates fall within acceptable ranges.



**Figure 1: SymDyn block diagram**

### 3.2.2 Aero-Structural Model: ADAMS

All structural loads and dynamics were estimated for the Baseline and Advanced turbines using ADAMS models. This section describes the common characteristics of the models and details the pitch control algorithm for the advanced system.

The models were developed and run using the same basic techniques employed in the WindPACT Rotor study [4, 5]. Every effort was made to limit changes in the modeling to those associated with the pitch control method and the redesign of the structure that resulted from the load mitigation. The structural degrees of freedom, aerodynamics model, wind inputs, and drive train characteristics were the same for all models that we ran. This was done to ensure that results could be evaluated and compared on a fair basis. These characteristics are described in greater detail in the following paragraphs.

Each blade was modeled as an elastic structure with 15 elements connected by six DOF “FIELDS”, for a total of 90 DOFs per blade. The drive train had one elastic (torsion) DOF plus the rigid body generator rotation DOF. The Baseline pitch system used a coupled pitch of all three blades (1 DOF), while the Advanced system required independent actuators on each blade (3 DOFs). The tower was rigid and the yaw position was fixed in all models.

Future work should extend the control system to include a flexible tower in the dynamic model and the control and estimation system.

AeroDyn subroutines [3] were used to calculate aerodynamic forces on the blades. No aerodynamic force was applied to the tower. The AeroDyn routines use dynamic inflow and dynamic stall models to calculate the unsteady, aeroelastic forces on each blade element (15 per blade).

Loads were calculated for a subset of loads cases as defined by the large turbine safety standard, IEC 61400-1 [1]. Normal power production in turbulence, normal power production in extreme gusts, and parked rotor in 50-year extreme winds were simulated. Fault conditions were not simulated. The same conditions were considered in the WindPACT Rotor study.

**Pitch Control Algorithm in ADAMS:** A mathematical description of the control algorithm is found in Appendix B.1. This section describes the implementation of this method in the ADAMS model of the turbine. The method uses state-space equations to calculate the pitch demand. This can be readily accomplished in ADAMS using a “user-written” Fortran subroutine named GSESUB. This subroutine receives inputs from the ADAMS numerical solver, integrates the state-space equations specified by the user, and returns the pitch demand value for each blade to the turbine model. This is transparent to the user, provided the subroutine is written according to ADAMS specifications.

A listing of the subroutine that was written for this purpose is provided in Appendix B.2. Appendix B.3 contains the essential portions of the ADAMS model dataset (adm file) that communicate with the GSESUB subroutine. ADAMS analysts who are familiar with user-written subroutines will be able to extract all details of the model implementation from these appendices.

The strong non-linearities of a wind turbine system require that the controller be designed for a series of different 10-minute average wind speeds. This is done using the SymDyn model described earlier for wind speeds from 12 to 26 m/s in 2 m/s increments (12 to 22 m/s for the advanced turbine configuration). For each mean wind speed, a set of operating points (pitch angle, rotor speed, and angle of attack at a specific spanwise station) is determined from the models.

However, it is unlikely that the actual operating turbine will have a reliable estimate of the current wind speed. Instead of using wind speed as an input to the controller, it is estimated from a running average of the pitch demand (which is uniquely related to wind speed over the range of

interest). The controller determines the state-space equation coefficients by linear interpolation as a function of the estimated wind speed. This allows the controller to work over the entire range of operating wind speeds, even though the estimator and controller “gains” vary significantly.

Much of the GSESUB is devoted to estimating the current coefficients and operating points for the state-space equations. The remainder of the subroutine receives the inputs and calculates the outputs that are described below.

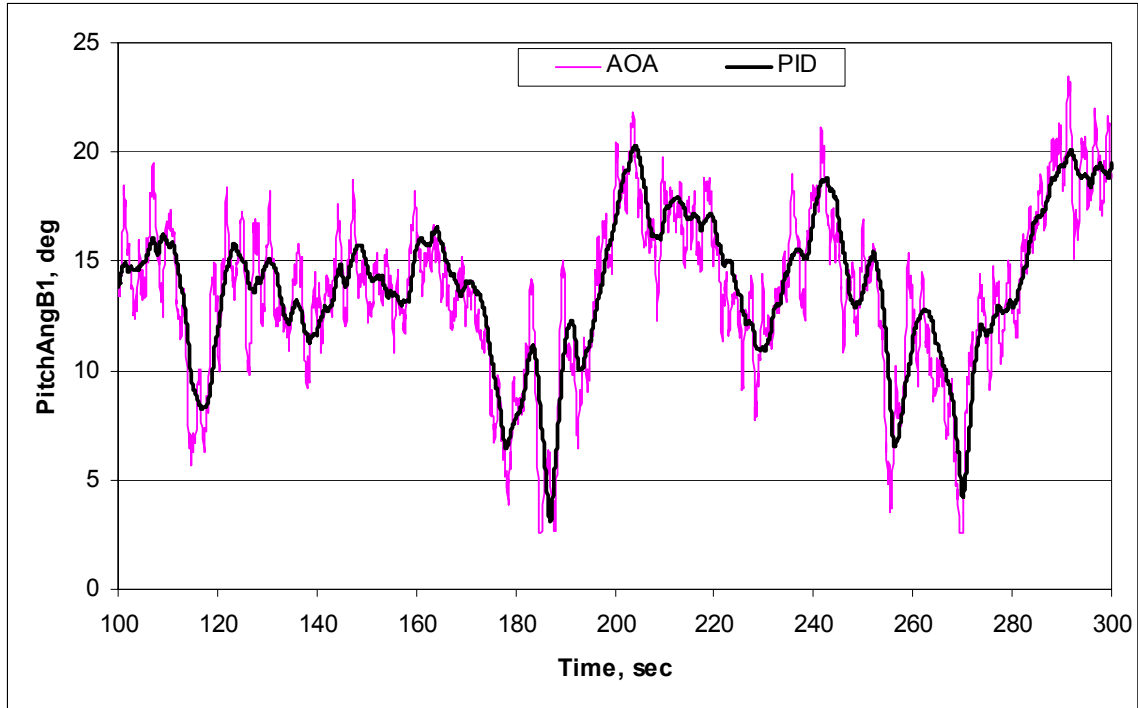
The inputs to the calculation are the current values of rotor speed, rotor azimuth position, and pitch angle of each blade. The outputs of the calculation are the pitch demand for each blade. Pitch demand is calculated by the state-space controller, and then limits on pitch angle and pitch rate are applied before the value is sent to the pitch actuator in the ADAMS model.

Difficulties were encountered initializing the state-space controller in the ADAMS model. These problems are artifacts of the way the simulation starts with the system at rest. Aerodynamic and gravitational loads are applied abruptly at the start of the simulation, and some time is required for these impulsive load transients to dampen. During this time, the state-space controller is unable to function properly because it does not include a startup algorithm and there are unrealistic numerical transients as ADAMS seeks a solution. This problem was encountered and solved during the WindPACT Rotor study with the PID pitch controller. For the present study, we started the simulations with the PID controller and then gradually transitioned to the state-space controller as the simulation found the quasi-steady solution. This worked quite well for the simulations. Of course, an actual turbine would require startup, shutdown, and protection system controls that were not included in these simulations.

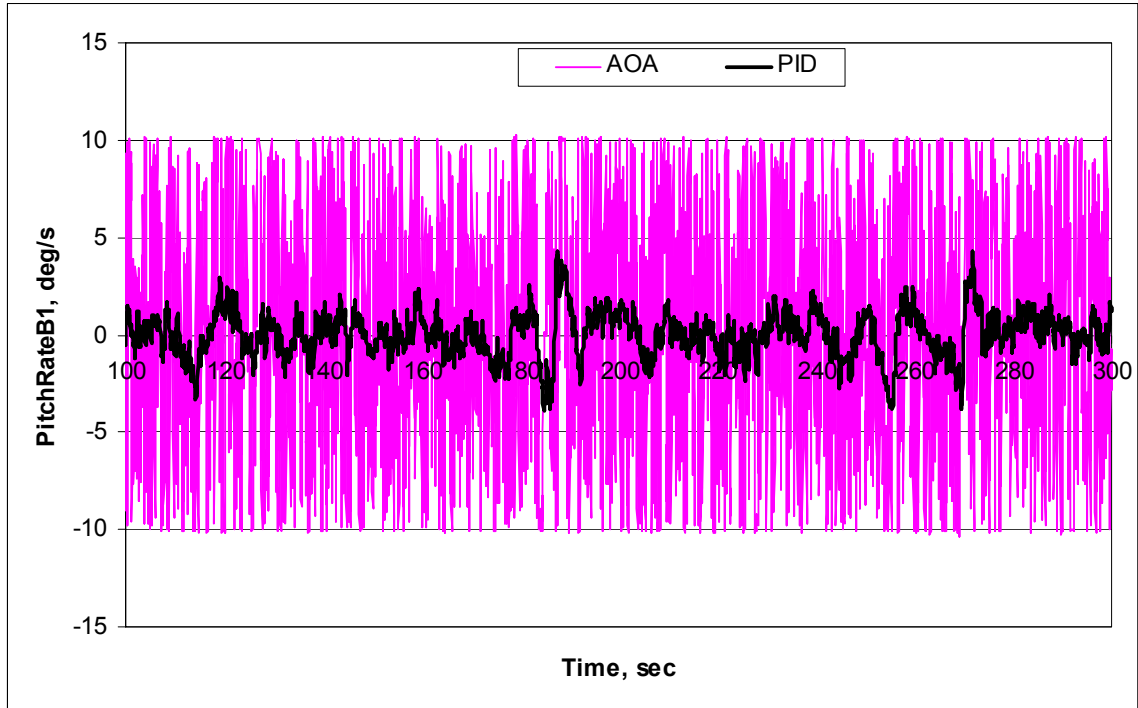
The early ADAMS simulations exhibited strong periodic responses in pitch demand that were not present in the simpler SymDyn model used to design the controller. This type of result is not unexpected when there are many more structural DOFs in the operating system than were present in the model used for control design. Increasing the penalty function for pitch activity in the controller design solved this problem. This reduced the pitch accelerations and actuator loads and resulted in smooth operation of the complete system. We also found it necessary to reduce the weighting factor for rotor azimuth error in the design of the controller. Since azimuth is a measured input to the control, the estimator need not determine the value. Furthermore, though it is important that the rotor speed be closely controlled, there is no need for tight control of rotor azimuth position.

Figures 2-8 compare the PID and AOA controller responses of various signals in ADAMS using the IEC-defined 16 m/s wind speed with category A turbulence [1].

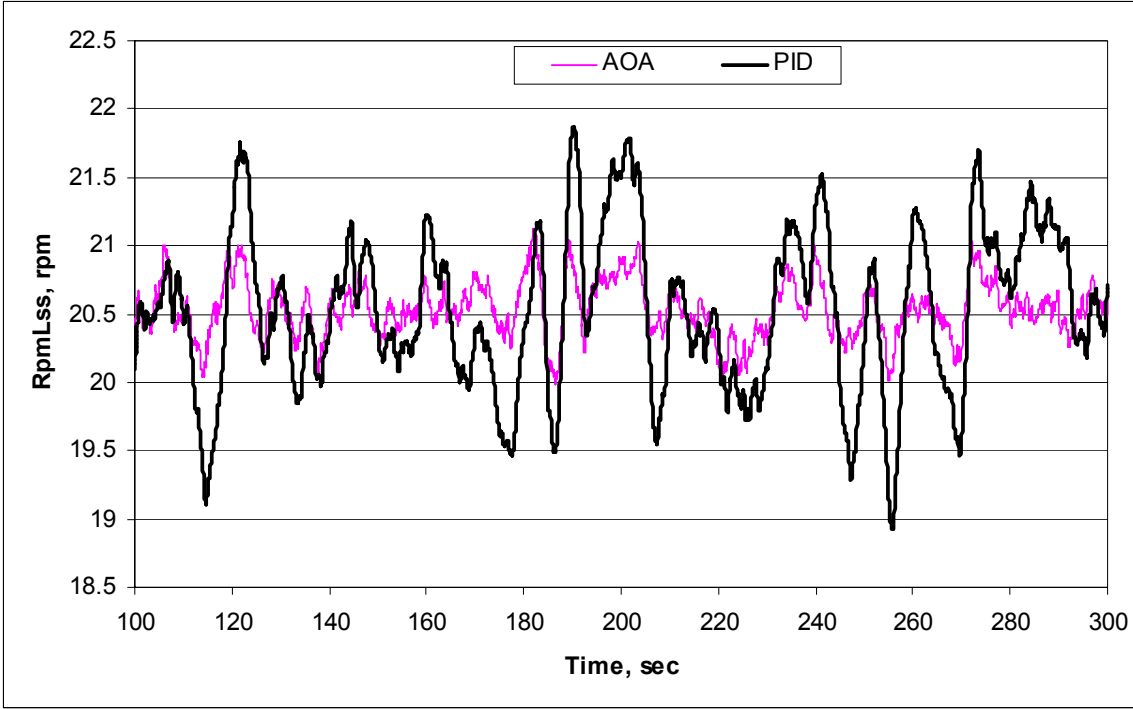




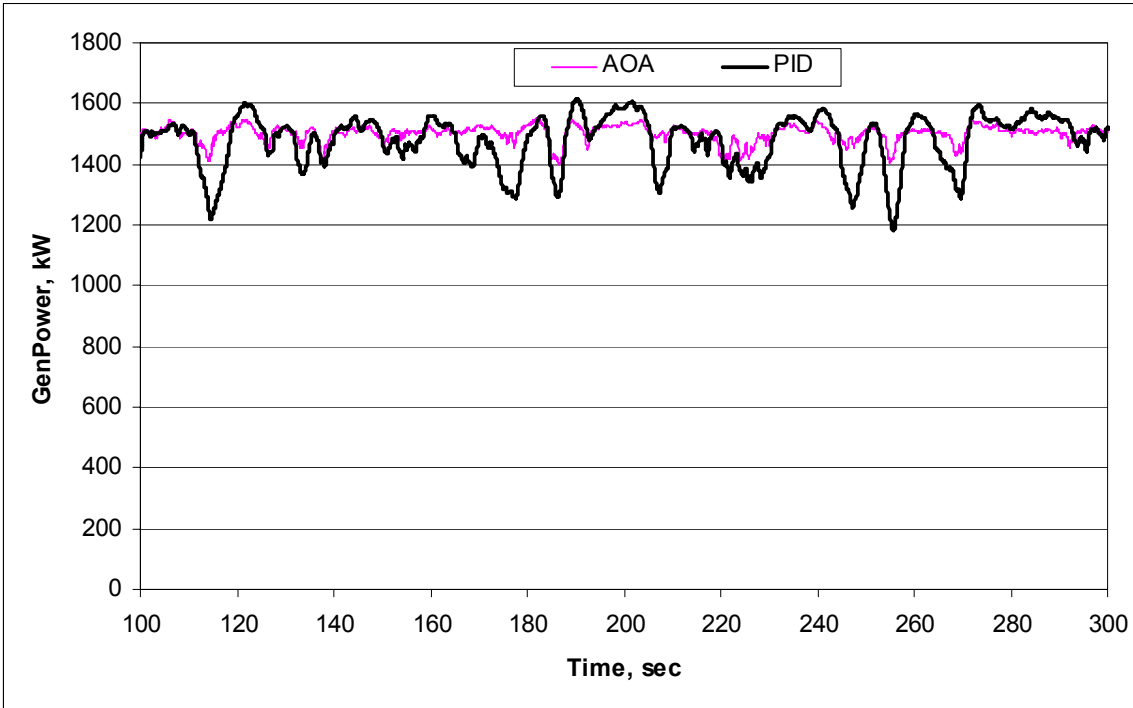
**Figure 2: Blade 1 pitch with 16 m/s wind and IEC category A turbulence**



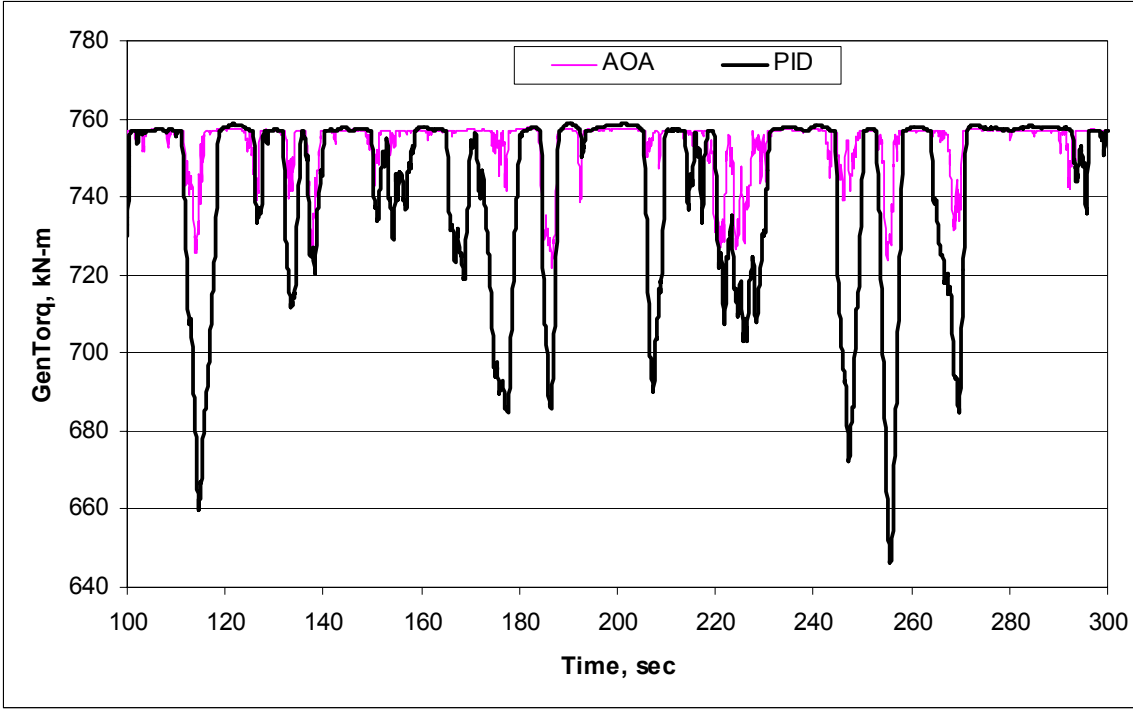
**Figure 3: Blade 1 pitch rate with 16 m/s wind and IEC category A turbulence**



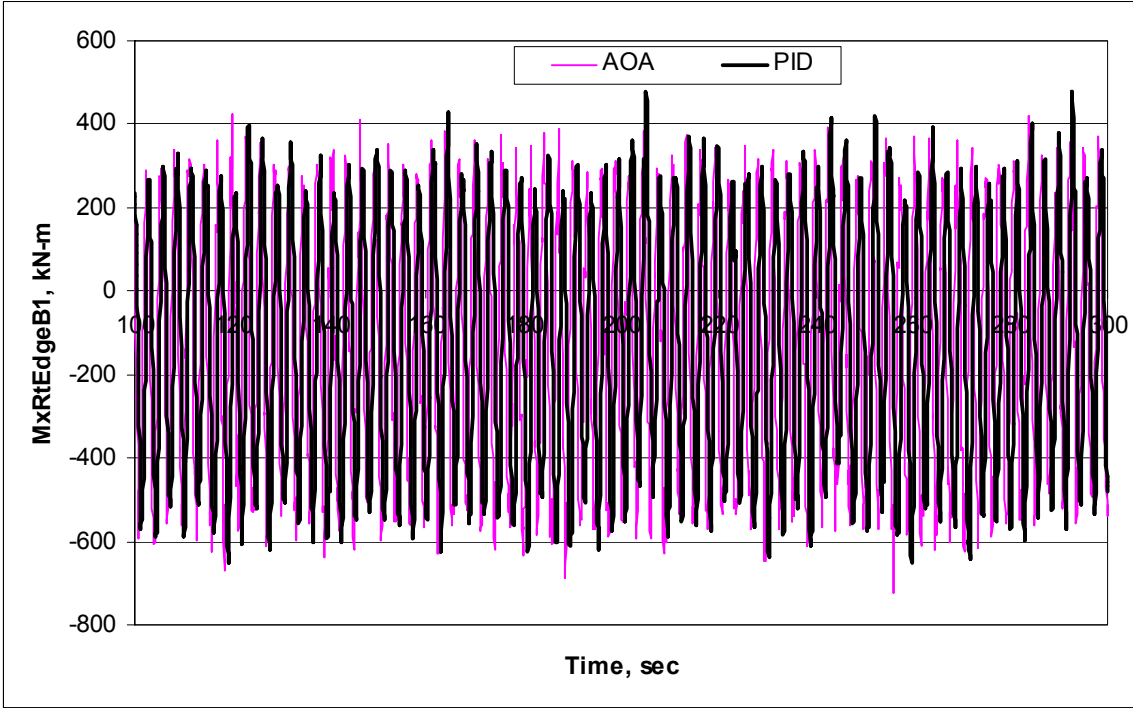
**Figure 4: Shaft speed with 16 m/s wind and IEC category A turbulence**



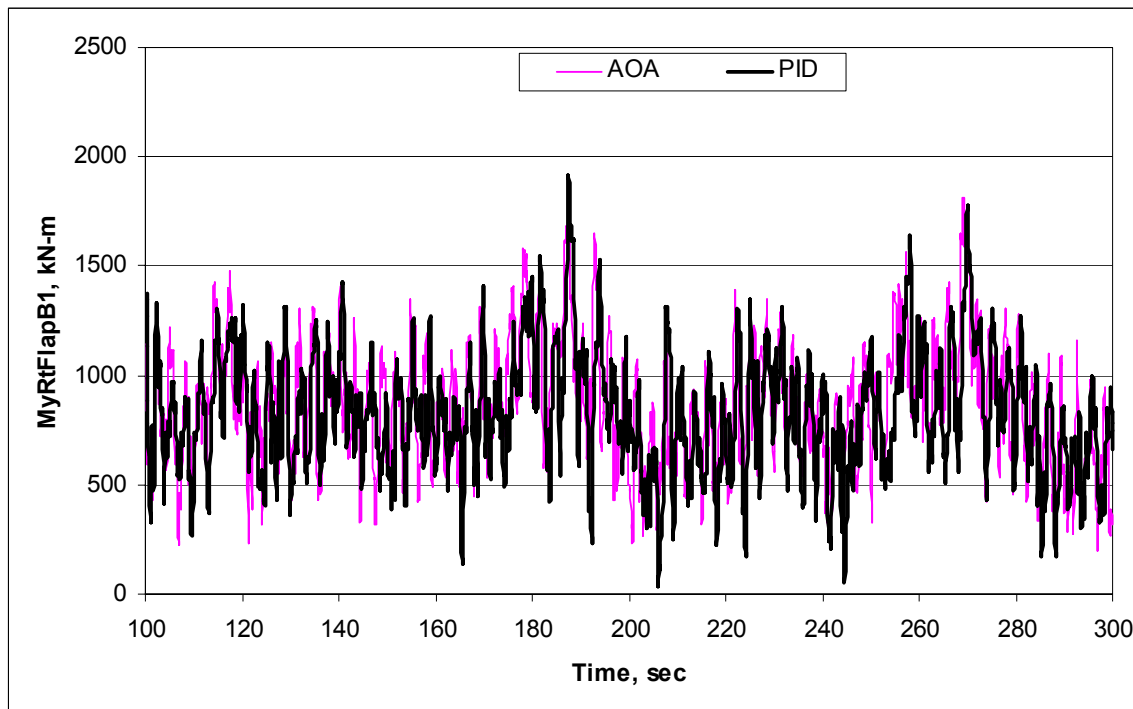
**Figure 5: Generator power with 16 m/s wind and IEC category A turbulence**



**Figure 6: Generator torque with 16 m/s wind and IEC category A turbulence**



**Figure 7: Blade 1 root edge moment with 16 m/s wind and IEC category A turbulence**



**Figure 8: Blade 1 root flap moment with 16 m/s wind and IEC category A turbulence**

Notice the increased blade pitch activity with the AOA controller in Figures 2 and 3, which led to finer control of the generator speed, power, and torque in Figures 4, 5, and 6. The blade root bending moment differences are hard to distinguish in the time series plots of Figures 7 and 8, but rainflow cycle counting revealed significant fatigue life savings in the flatwise direction with the AOA controller. These results are discussed further in Section 4.1.

### ***3.2.3 WindPACT Design and Analysis Tools***

We used the WindPACT design spreadsheets developed by Global Energy Concepts, Inc. to produce the initial input files for ADAMS structural modeling and post-processing of turbine models. The WindPACT 1.5-MW configuration (Excel file 1.5A08CO1V07) turbine model input was used as the baseline for this concept design study. The effort required that the team use the WindPACT blade design spreadsheets to create a known baseline that could be used as a starting point for the study. The effort required to develop a new baseline model was beyond the scope of the study.

The spreadsheets used in this work were:

- BladeDesign1.5A08V07.xls
- InputData1.5A08V07adm.xls
- Loads\_1.5A08V07adm.xls
- DesignEval&cost1.5A08C01V07cAdm.xls.

A detailed presentation of the WindPACT approach to rotor design is presented in Appendix C.1. The capabilities, limitations, and application of these tools are presented in Appendix C.2.

### 3.3 Analysis Methodology

The analysis methodology for this project involves establishing a baseline, developing a new design with controlled alterations to the baseline, and comparing the resulting figures of merit for the two cases. The analysis procedure follows:

1. Define baseline
2. Design and tune controller on baseline; optimize for loads reduction
3. Import controller into ADAMS
4. Test different baseline configurations (corrected torsional stiffness [GJ]), flexible tower)
5. Quantify loads reductions
6. Design extended rotor
7. Re-tune controller for extended rotor
8. Run ADAMS and check new loads
9. Fine-tune components to meet design margins
10. Generate Cp-TSR curve and AEP for extended rotor
11. Compare resulting costs, AEP, and COE.

#### 3.3.1 Baseline Turbine

The baseline turbine analysis involved checking the WindPACT turbine input files (BladeDesign1.5A08V07.xls, InputData1.5A08V07adm.xls), duplicating the loads files from the ADAMS model (Loads\_15A08V07C01\_GEC\_FlexibleTower\_PID.xls), and checking the analysis and energy production in the DEC (DesignEval&cost1.5A08C01V07adm.xls), and AEP files (Perf&Torque1.5A08.xls). This task proved to be much more difficult than expected, as it took numerous iterations over 13 months to correctly identify and obtain a complete set of the correct baseline files.

**Versions:** The WindPACT blade studies completed by the GEC team in June 2002 [5] included a fast-evolving baseline with at least seven main versions and more sub-versions, plus some discrepancy in the final report on the accepted version number. The project started with a version 03 baseline provided by the GEC team and referred to in the main body of their final report. Subsequent version 07 files provided by Windward Engineering caused some confusion before getting some assurance that 07 is the correct and final version. Later work on the analysis procedures revealed discrepancies between the design margins and the margin requirements, leading to the further discovery of a properly tuned version 07c.

**Improvements:** The team also identified several improvements in the baseline structural property and axis calculations as detailed in Section 3.4. This work raised a problematic question: Do the improvements have enough impact on baseline loads and design to justify shifting to a revised LWST baseline, or is the impact modest enough to justify holding to the GEC baseline? This dilemma created substantial unexpected work, as it was deemed prudent to test the effect of using the corrections before making a choice. Although these changes allowed the blade weight to drop 15%, they were not utilized so that the project could stay consistent with the original WindPACT method.

Finally, armed with a full understanding of the WindPACT baseline turbine and some desired improvements, the project moved ahead in the Preliminary Design phase, examining the impact of several baseline alternatives, as detailed below.

### 3.3.2 Preliminary Design

The preliminary design involved developing an advanced AOA-based controller, augmenting the baseline turbine with the AOA controller, and running a new set of structural loads and quantifying the difference with the baseline loads. The AOA controller was developed using a SymDyn model of the WindPACT Baseline turbine, confirming the model by comparison with the baseline ADAMS model results, and then Modifying SymDyn's interface with AeroDyn to make the blade angle of attack array available as a control input in the Simulink program.

Early efforts in the AOA controller design did not include modeling of the tower motion. As a result, runs with the tower elastic DOFs activated were unstable. Subsequent work re-tuned the controller to handle tower motion with stability; however, full tower flexibility will require further development of the controller model.

The baseline turbine issues led us to evaluate several different configurations for load impacts before and after replacing the PID control with AOA control. They included:

1. Original GEC baseline model with flexible tower
2. Original GEC baseline model with rigid tower
3. Revised baseline model with corrected blade properties and flexible tower
4. Revised baseline model with corrected blade properties and rigid tower.

Substantial review of the load data resulting from this matrix of eight design options revealed minimal differences between the original and revised baselines. However, the differences between rigid and flexible tower results were more subtle, requiring completion of the design/analysis process. Therefore we chose to stick with the original baseline and focus on those two configurations for evaluating the AOA controller benefits over a traditional PID controller:

1. Original GEC baseline model with flexible tower
2. Original GEC baseline model with rigid tower.

Further tuning of the AOA controller brought the flexible tower results in line with those of the rigid tower, thereby validating the use of the original GEC baseline with flexible tower for the rest of the project:

1. Final Baseline: Original GEC baseline model with flexible tower.

The selection of the GEC baseline over the modified version was not without difficulty, given the incorrect blade torsional stiffness values noted below in Section 3.4. However, the impact of corrected values on the load results was minimal in the cases examined. This result, combined with a desire to minimize confusion when presenting results to the broader wind turbine design community, created a compelling case to adhere to the original baseline with its flaws. This choice may present future challenges however, as longer blades with lower blade torsional frequencies are explored.

The WindPACT Baseline followed these general design requirements:

- Three blades
- Upwind
- Full-span variable-pitch control
- Rigid hub

- Blade-first flatwise natural frequency between 1.5 and 3.5 per revolution
- Blade-first edgewise natural frequency greater than 1.5 times flatwise natural frequency
- Rotor solidity between 2% and 5%
- Variable-speed operation with maximum power coefficient = 0.50
- Maximum tip speed  $\leq 85$  m/s
- Air density = 1.225 kg/m<sup>3</sup>
- Turbine hub height = 1.3 times rotor diameter
- Annual mean wind speed at 10-m height = 5.8 m/s
- Rayleigh distribution of wind speed
- Vertical wind shear power exponent = 0.143
- Rated wind speed = 1.5 times annual average at hub height
- Cut-out wind speed = 3.5 times annual average at hub height
- Dynamically soft-soft tower (natural frequency between 0.5 and 0.75 per revolution)
- Yaw rate less than 1 degree per second.

### 3.3.3 Final Design

Rotating blade moments are proportional to radius cubed, so one can conversely use the inverse cube of the reduction in moment (achieved through the AOA controller) to determine the increase in radius to produce the original moment. For the parked rotor, a square function would apply. Therefore the Final Design analysis procedure is as follows:

1. Select baseline
2. Run all the loads with the advanced controller applied to that baseline
3. Quantify the loads reductions
4. Determine rotor extension limits: inverse cubed root of operating loads reductions and inverse square root of parked loads reductions
5. Design a longer blade to these limits
6. Run the loads with the longer blade
7. Fine-tune the component sizing to meet the margin requirements
8. Estimate the new blade costs
9. Estimate the new blade performance
10. Tally the new COE figures of merit.

Step 7 is open to interpretation, as it tries to duplicate margins from a baseline in which many are much higher than needed, and the margin requirements themselves are very coarsely defined. Full 3-D blade scaling was not used because the baseline section strengths were considered appropriate for loads brought back up to the baseline envelope.

### 3.4 Assumptions and Limitations

A list of assumptions and limitations is presented here, with more detailed discussions included in Appendix C.2:

- A reduced set of IEC load cases includes all that are readily modeled but omits fault conditions.
- Fatigue distribution: The analysis procedure used large design margins instead of extrapolating the fatigue curves.

- A form of engineering judgment was introduced through “scale factors” hidden in the component stress tables; they can be used to apply stress concentration factors and other derating considerations.
- Full 3-D stress states were not used in the component stress analyses; simple beam theory is used instead for a single loading direction for each analysis.
- Loads were not included for 25%, 50%, and 75% span locations of Blades 2 and 3 in the WindPACT approach, but these blades’ loads could dominate under certain conditions.
- Negative margins were accepted for the main shaft to maintain a design similar to known field dimensions.
- Several blade structural property calculation errors were discovered, checked, shown to have minimal loads impact, and then retained in this study.
- Torque and gravity loads were summed for trailing-edge compression, but torque minus gravity would be the correct application unless the rotor spins backward.
- Inaccurate ‘c’ values were used in the strain-moment equations.
- Blade fatigue strain limits and S-N curves were extra conservative, as explained below.
- The handling of the trailing-edge spline was awkward and probably unnecessary.

## **4.0 RESULTS**

### ***4.1 Preliminary Design: System Performance and Influential Design Parameters***

The preliminary design effort involved comparing loads from several configurations in an effort to capture the controller capability to reduce turbine loads on the baseline turbine. Table 2 is a summary loads output created in the DesignEval&cost1 spreadsheets. The configurations were the baseline turbine with the noted exceptions.

Four ADAMS models are represented in the results in Table 2. The ADAMS models are: 1) WindPACT Baseline, 2) WindPACT Baseline with a rigid tower, 3) AOA controller on the WindPACT Baseline with a flexible tower, and 4) AOA controller implemented with a rigid tower. The rigid tower runs were performed because the AOA controller had not implemented tower feedback and it was thought that controller performance would suffer from the lack of tower dynamic modeling. The relative loads data with a rigid tower are similar to the more correct flexible tower data with few exceptions. The most significant design load reduction of the AOA controller vis-à-vis the PID base line is in the root flap bending load case.



**Table 2: Loads Summary: Preliminary Design Flexible and Rigid Tower Models**

			Baseline	PID	Baseline PID (RIGID TOWER)	AOA CONTROLLER	AOA (RIGID TOWER)
tilt angle		deg	5		5	5	5
coning angle		deg	-		-	-	-
angle of first contact		deg	-		-	-	-
max tip out of plane displt	max abs	m	2.82		2.90	2.76	2.76
min tip out of plane displt	max abs	m	-1.71		-2.14	-1.74	-1.77
tip-tower clearance margin			0.14		0.11	0.16	0.16
blade rt flap mt	max abs	kN m	3298		2472	2058	2061
	equiv fatigue *		1237		1211	1045	1059
blade rt edge mt	max abs		815		997	890	916
	equiv fatigue		847		861	858	874
blade 25% flap mt	max abs		1350		1471	1219	1213
	equiv fatigue		733		725	646	636
blade 25% edge mt	max abs		412		488	457	449
	equiv fatigue		382		397	400	412
blade 50% flap mt	max abs		553		618	546	544
	equiv fatigue		338		338	317	304
blade 50% edge mt	max abs		146		181	152	166
	equiv fatigue		126		135	135	140
blade 75% flap mt	max abs		133		149	135	135
	equiv fatigue		87		90	84	81
blade 75% edge mt	max abs		33		39	33	37
	equiv fatigue		23		25	25	26
shaft/hub My	max abs		2308		2408	1811	1843
	equiv fatigue		583		581	467	458
shaft/hub Mz	max abs		2210		2354	1841	2407
	equiv fatigue		579		582	463	462
shaft thrust	max abs	kN	324		336	279	270
	equiv fatigue		44		45	40	41
shaft Mx	max abs	kN m	3424		3418	3227	3227
	equiv fatigue		83		75	82	84
yaw brg My	max abs		3079		2923	2083	2147
	equiv fatigue		468		469	417	406
yaw brg Mz	max abs		1966		1487	1708	1869
	equiv fatigue		473		468	424	410
tower base Mx	max abs		31951		16060	27878	27878
	equiv fatigue		2058		1090	1693	1120
tower base My	max abs		27390		24924	29130	27210
	equiv fatigue		5482		3762	5594	3407

\* Equivalent fatigue is the once per rev equivalent load

Table 3 presents the loads ratios between baseline AOA and PID cases. Ratios are shown for both the flexible and the rigid tower models. However, the rigid tower will not be considered for further analysis as the flexible tower model represents a more realistic configuration. Tower feedback will require future study.

The maximum root flap bending moments for the flexible rotor with AOA and PID controllers have a ratio of 62%. The load reduction represents a substantial benefit from the AOA controller. The blade root edge moment at 25% span has the highest load ratio (111%); this load may require some extra attention to design details with the AOA controller. Blade 3 had the highest root flap bending moments, but only Blade 1 spanwise load data were tracked. Another load increase was tower base moment, with a ratio of 106%. This load should be mitigated with tower feedback in the controller.

**Table 3: Load Ratio Summary: Flexible and Rigid Models, AOA and PID Controllers**

			<b>FLEXIBLE LOAD RATIO AOA / BASELINE</b>	<b>RIGID LOAD RATIO AOA / BASELINE</b>
tilt angle		deg	-	-
coning angle		deg	-	-
angle of first contact		deg	-	-
max tip out of plane displt	max abs	m	98%	95%
min tip out of plane displt	max abs	m	102%	82%
tip-tower clearance margin			-	-
blade rt flap mt	max abs	kN m	62%	83%
	equiv fatigue		85%	87%
blade rt edge mt	max abs		109%	92%
	equiv fatigue		101%	102%
blade 25% flap mt	max abs		90%	83%
	equiv fatigue		88%	88%
blade 25% edge mt	max abs		111%	92%
	equiv fatigue		105%	104%
blade 50% flap mt	max abs		99%	88%
	equiv fatigue		94%	90%
blade 50% edge mt	max abs		105%	91%
	equiv fatigue		107%	104%
blade 75% flap mt	max abs		102%	91%
	equiv fatigue		96%	89%
blade 75% edge mt	max abs		100%	93%
	equiv fatigue		108%	102%
shaft/hub My	max abs		78%	77%
	equiv fatigue		80%	79%
shaft/hub Mz	max abs		83%	102%
	equiv fatigue		80%	79%
shaft thrust	max abs	kN	86%	80%
	equiv fatigue		91%	91%
shaft Mx	max abs	kN m	94%	94%
	equiv fatigue		98%	111%
yaw brg My	max abs		68%	73%
	equiv fatigue		89%	87%
yaw btg Mz	max abs		87%	126%
	equiv fatigue		90%	88%
tower base Mx	max abs		87%	174%
	equiv fatigue		82%	103%
tower base My	max abs		106%	109%
	equiv fatigue		102%	91%

Table 3 compared loads reductions by showing the ratio of AOA to PID controller loads using flexible towers at the most critical location with near-zero design margin. Values below 100% imply the associated loads were lower with the AOA controller and vice versa. The blade root flap extreme load dropped 37.5% and the fatigue dropped 15%, which allow blade extensions of 11% and 5% respectively based on cubic load-vs-diameter scaling.

As might be expected, the design process is wrought with complexity. Other component locations showed more modest load reductions plus some increases. This meant that the resulting design might require some component cost increases unless taking advantage of ample original margins.

The designations “rigid” and “flexible” refer to two variations on the tower modeling in ADAMS: one with the tower essentially omitted (represented as a rigid body with rigid foundation joint) and the other using several flexible elements. Including tower flexibility allowed a new set of interactions that could reduce or augment any of the loads. The controller did not model tower flexibility in either case, but this step would be essential in future studies.

The design load cases responsible for the maximum model loads are shown in Table 4. The load case responsible for the root flap bending design load is the EWM  $V_{e50}$  for the IEC wind class 2. It was run as a turbulent inflow case.

**Table 4: Design Load Cases Producing Maximum Loads**

			Baseline PID	Baseline PID (RIGID TOWER)	AOA CONTROLLER	AOA (RIGID TOWER)
blade rt flap mt	max abs	kN m	Turb42.5ms_S5	Turb42.5ms_S2	edc_50n16.wnd	edc_50n16.wnd
blade rt edge mt	max abs		eog_50_12.wnd	Turb24ms_S1	Turb24ms_S3	Turb24ms_S3
blade 25% flap mt	max abs		Turb42.5ms_S1	Turb42.5ms_S2	edc_50n16.wnd	edc_50n16.wnd
blade 25% edge mt	max abs		eog_50_16.wnd	Turb24ms_S2	Turb24ms_S1	Turb20ms_S3
blade 50% flap mt	max abs		Turb42.5ms_S3	Turb42.5ms_S2	edc_50n16.wnd	edc_50n16.wnd
blade 50% edge mt	max abs		Turb24ms_S1	Turb24ms_S2	Turb24ms_S1	Turb24ms_S1
blade 75% flap mt	max abs		ecd_00p24.wnd	ecd_00p24.wnd	edc_50n16.wnd	edc_50n16.wnd
blade 75% edge mt	max abs		Turb24ms_S1	Turb24ms_S2	Turb24ms_S3	Turb24ms_S1
shaft/hub My	max abs		ecd_00n24.wnd	ecd_00n24.wnd	ecd_00p12.wnd	ecd_00p12.wnd
shaft/hub Mz	max abs		ecd_00n24.wnd	ecd_00n24.wnd	ecd_00p16.wnd	ecd_00p16.wnd
shaft thrust	max abs	kN	eog_50_12.wnd	eog_50_12.wnd	Turb16ms_S2	Turb16ms_S2
shaft Mx	max abs	kN m	Turb42.5ms_S5	Turb42.5ms_S4	Turb42.5ms_S4	Turb42.5ms_S4
yaw brg My	max abs		ecd_00n24.wnd	ecd_00n24.wnd	ecd_00p12.wnd	ecd_00n16.wnd
yaw bfg Mz	max abs		Turb42.5ms_S4	ecd_00n24.wnd	Turb24ms_S1	ecd_00p16.wnd
tower base Mx	max abs		Turb42.5ms_S4	Turb42.5ms_S2	Turb42.5ms_S1	Turb42.5ms_S1
tower base My	max abs		eog_50_12.wnd	eog_50_12.wnd	Turb24ms_S1	Turb16ms_S2

As can be seen in the table above, several of the design load cases producing the maximum loads involve wind direction changes. The control system and the simulations that were used did not allow for yaw control. However, given that the estimator produces estimates of the horizontal wind shear (that is closely correlated with horizontal wind direction change), future work should investigate the potential for loads reduction by incorporating active yaw control.

#### **4.2 Final Design: System Performance and Influential Design Parameters**

Working with the flexible PID and AOA cases, the greatest margin relief was found at the blade root bending compression surface (static load): from -0.2% margin to 60.0%. Using the defining loads of 3300 Nm (PID) and 2060 Nm (AOA) produced a 37.6% load decrease; the cube root of 1.376 would allow an 11.2% diameter increase. However, fatigue margins rose significantly in the lifetime domain, which is very sensitive to changes in cyclic loading. Working with fatigue equivalent loads of 1237 Nm (PID) and 1045 Nm (AOA), the drop was 15.5%, with a cube root allowance of 4.9%.

After exploring the advantages and disadvantages of various approaches to blade extension, the project team chose to stretch the blade length 10% without scaling any other dimensions; ie: shift the blade spanwise locations outward while holding all other dimensions and section properties constant. Design limitations chosen early in the project still apply (hold rated power at 1.5 MW and maximum tip speed at 75 m/s). This approach allowed for fast and simple blade modification,

while retaining section properties suited to the original loads envelope. After checking the resulting loads against the baseline loads, the length can be readjusted accordingly. The main drawback with this approach is a deviation from the original aerodynamic concept: the lower solidity blade implies a new blade design, with higher optimum tip speed ratio and the risk of performance loss. Fortunately, the resulting performance was essentially identical.

Both baseline and final design parameters are presented in Table 5.

**Table 5: Baseline and Final Wind Turbine Specifications**

Parameter	Baseline Specification	Final Specification
Design wind regime	IEC class 2	IEC class 2
Rated tip speed (m/s)	75	75
Hub height (m)	84	84
Rotor diameter (m)	70	77
Rotor-swept area (m <sup>2</sup> )	3848	4657
Rating (kW)	1500	1500
Pitch Control System	Electromechanical, collective	Electromechanical, independent
Pitch Control Algorithm	PID	State-space with AOA input
Maximum Rotor Speed (rpm)	20.5	18.6
Rotor Tilt (deg)	5	5
Blade Coning (deg)	0	0
Hub Overhang (m)	3.3	3.3
Rotor Solidity	0.050	0.041
Max Blade Chord	8.0% of radius	7.3% of radius
Hub Radius	5.0% of radius	5.0% of radius
Airfoil, thickness, twist:		
5-7% span	cylinder, 1.0, 10.5	cylinder, 1.0, 10.5
25% span	scaled S818, 0.30, 10.5	scaled S818, 0.30, 10.5
50% span	scaled S825, 0.24, 2.5	scaled S825, 0.24, 2.5
75% span	scaled S825, 0.21, 0.0	scaled S825, 0.21, 0.0
100% span	scaled S826, 0.16, -0.6	scaled S826, 0.16, -0.6

#### 4.2.1 Structural Design Results

The approach to the contract objective of reducing COE was to improve energy production by increasing diameter while minimizing the impact on initial capital cost (ICC). The AOA controller, by mitigating aerodynamic forces, allowed the blade to be stretched by 10% without increasing the blade sections. This resulted in the blade weight increasing in a nearly linear fashion with length rather than following the standard cube law for scaling. The increase in blade weight was 452 kg per blade, or 10.5%.

The stretched rotor was then run on the ADAMS simulation to determine system loads using the AOA controller. It was found that the blade did not meet the fatigue margins set for the baseline rotor at two locations, the trailing edge at the 25% radial station and the high-pressure surface at the 75% radial position. The higher flexibility also resulted in the tip/tower clearance falling below the minimum specified. The WindPACT approach to simulate the weight necessary to stiffen the blade is to add a weight penalty equal to the percent deficit in tip clearance (4.8% in this case). This amounted to a blade weight increase of 228 kg. The weight was incorporated into

the blade as added skin thickness. The skin is a sandwich structure of fiberglass and balsa, and the fiberglass portion was increased from 2.16 mm to 2.94 mm. The new blade structural properties with the increased skin thickness are presented in Figures 9-13 and compared to those of the baseline rotor. It should be noted that the properties of the original 77-meter stretched rotor are identical to those of the 70-meter baseline rotor when presented versus non-dimensional radial position ( $r/R$ ) because the blade was simply stretched 10% in length without changing the section properties.

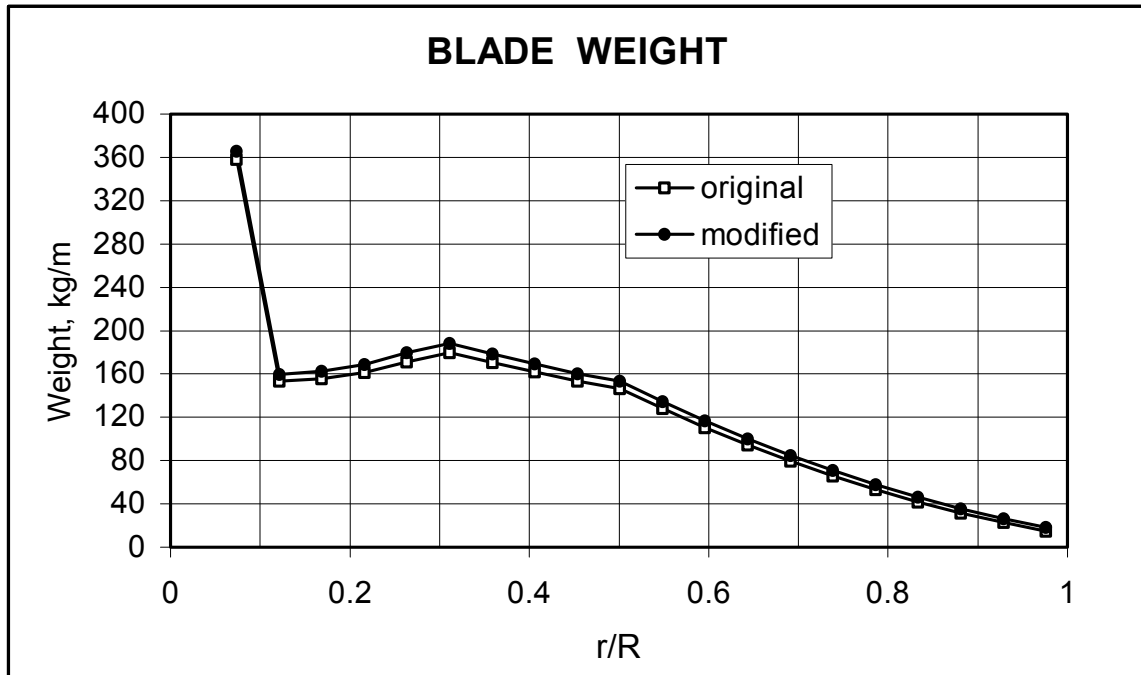


Figure 9: Blade weight

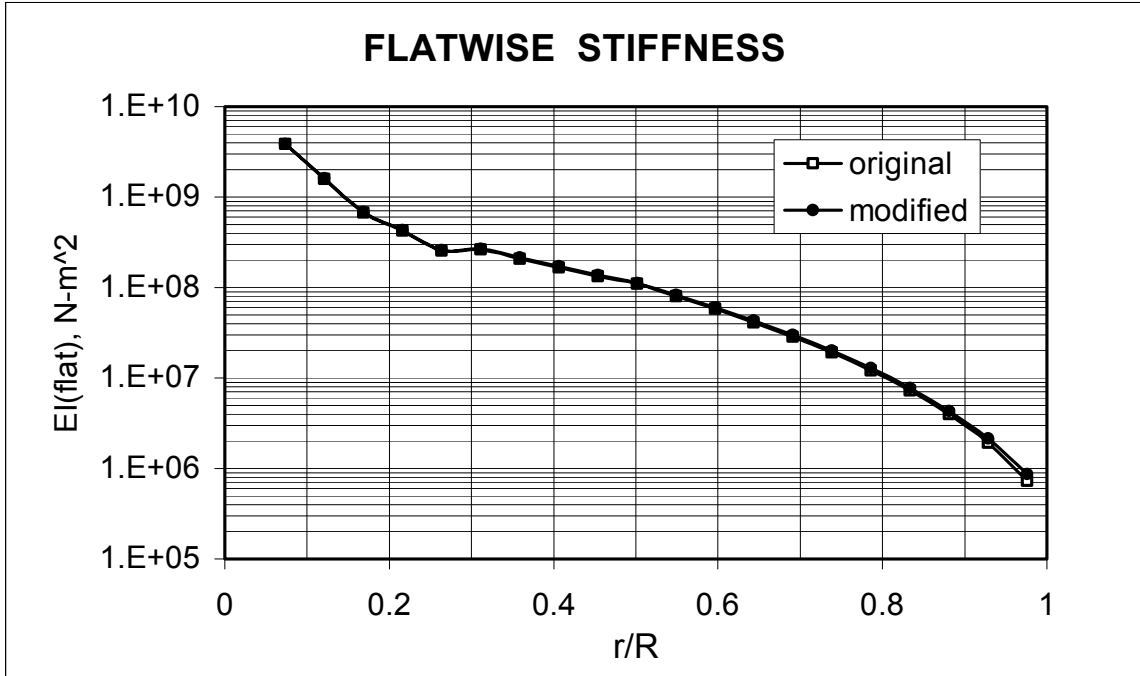


Figure 10: Flatwise stiffness

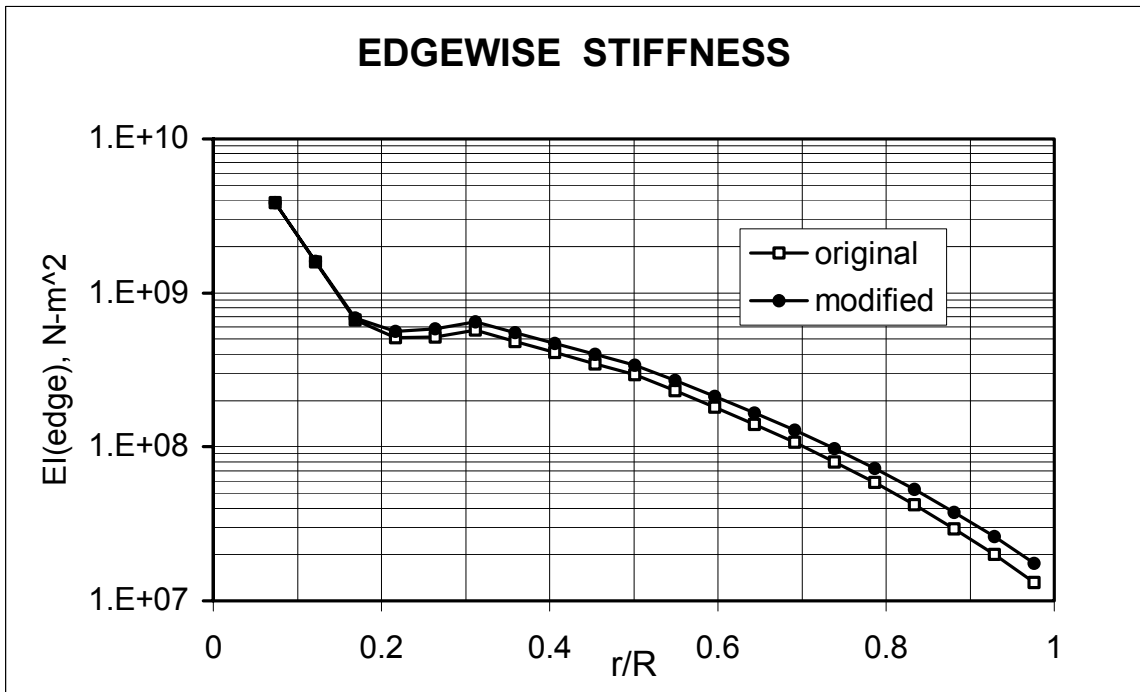
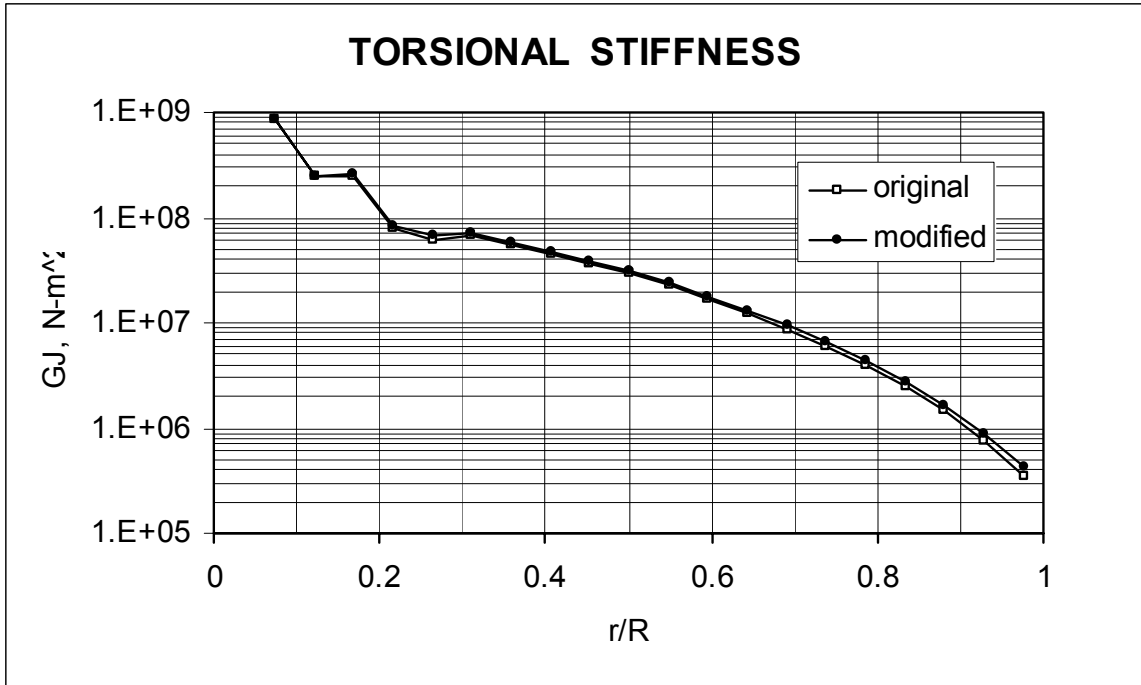
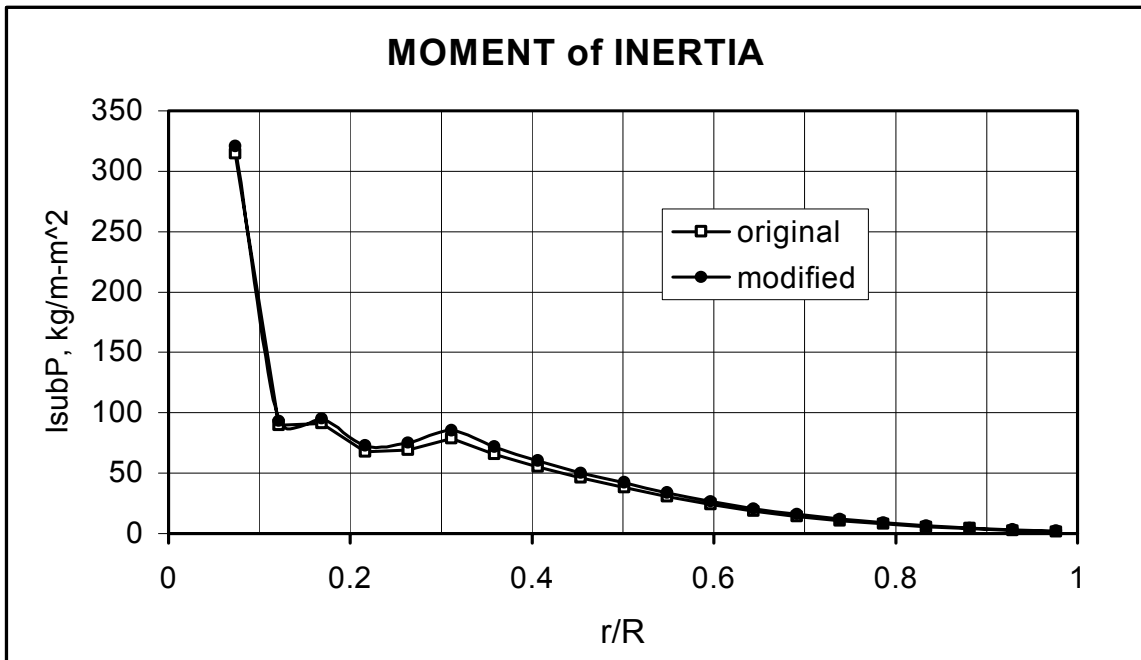


Figure 11: Edgewise stiffness



**Figure 12: Torsional stiffness**



**Figure 13: Moment of inertia**

Although the increased skin thickness had only a modest change in the blade properties, it was sufficient to reduce the moment-strain relationships enough to resolve the fatigue issues revealed in the ADAMS results of the unmodified stretched blade. A comparison of these relationships at

four radial stations is shown in Table 6. Values are shown for the low-pressure surface (lp), high-pressure surface (hp), and trailing edge (te).

**Table 6: Strain vs. Moment Relationships**

Microstrain/Moment, ustrain/N-m

Station	Original, Skin = 2.16mm			Modified, Skin = 2.94mm		
	ulp/My	uhp/My	ute/Mx	ulp/My	uhp/My	ute/Mx
Root (7%)	0.000815	0.000815	0.000815	0.000815	0.000815	0.000815
25%	0.001432	0.001538	0.002442	0.001354	0.001454	0.002102
50%	0.003197	0.003379	0.005213	0.002994	0.003165	0.004382
75%	0.012444	0.013128	0.013496	0.011204	0.011820	0.010511

#### 4.2.2 Performance Results

The increase in blade length without a corresponding increase in chord resulted in lower rotor solidity and higher blade aspect ratio. This can be favorable from an aerodynamic viewpoint because it reduces the strength of the tip vortex, thus lowering induced losses.

Lacking the PROPID model used by GEC for the WindPACT baseline, the baseline performance model was rebuilt using WTPerf. After verifying its ability to generate a similar Cp-TSR curve used in the Performance & Torque spreadsheet (Cp slightly reduced), a Final Design model was created for the extended rotor. The Final Design model increased Cpmax slightly (from 0.489 to 0.491) and increased optimum TSR by 0.5 (from 7.5 to 8.0). These changes were small enough to use the baseline Cp-TSR curve with the extended rotor for calculating AEP.

The AEP calculation is completed in the GEC Performance & Torque spreadsheet, which calculates a detailed power curve and then applies the selected wind distribution to derive annual energy production. The maximum tip speed was held constant at 75 m/s. The rotor was operated at constant tip speed ratio up until the maximum tip speed was reached, which occurred at 10.5 m/s wind speed, compared to 11.0 m/s for the baseline turbine. At this point, rotor speed and system power output were maintained at 18.6 rpm and 1500 kW, respectively. The Baseline Rotor system efficiency of 92.5%, peak power coefficient of 0.504 used by GEC, and wind shear exponent of 0.143 were also used in this analysis. The AEP at a surface wind speed of 5.8 m/s was 5,999,000 kWh, a 10.2% increase over the baseline performance.

### 4.3 Figure-of-Merit Impacts

#### 4.3.1 Reliability, Maintainability, System Lifetime

This concept tries to utilize existing subsystems and components as much as possible to minimize any adverse system impacts. All major structural components are adjusted as necessary for the new loads set to ensure that their lifetimes and safety margins match the baseline.

However, the advanced blade pitch control concept does require a few extra components that may need maintenance and risk failure. They include the angle of attack sensors and the backup power supply. For safety and system life, the controller can be instructed to execute a failsafe pitch to feather if the angle of attack sensors are lost. The backup power system is required during loss-of-grid fault conditions to ensure control power for parked rotor load mitigation. These extra components will cause a slight decrease in system reliability and increase in maintenance



requirements, and they deserve some engineering attention to minimize these effects. The pitch control algorithm should not impact reliability and maintainability because it is intended for use on existing controller hardware.

#### ***4.3.2 Cost of Energy Impact***

The Design and Cost spreadsheet developed under the WindPACT program was employed to determine fatigue damage of all load-bearing components. Also, using the same cost functions developed for the baseline rotor, the costs and COE of the stretched rotor were calculated. Details of the spreadsheet are presented in Appendix C.3. Except for electrical, all the major cost groups were affected by the addition of the AOA controller and the increased rotor diameter. Specific component costs that differ from the baseline rotor are summarized in Table 7.

**Table 7: Component Cost Impacts**

COST GROUP Sub group	BASELINE ROTOR	AOA ROTOR	COMMENT
ROTOR	D = 70m	D = 77m	
Blades	\$137,219	\$159,004	Blade weight was increased by 10.5% for the elongated blade and another 4.8% for the added skin thickness to increase stiffness to satisfy fatigue and tip clearance requirements
Pitch System	\$35,548	\$35,548	Although pitch loads dropped, duty cycle increased, so we assumed the standard pitch system with the AOA rotor. The DEC spreadsheet formula would have reduced this value to \$25,216 for AOA
DRIVE TRAIN			
Gearbox	\$160,688	\$162,951	The gearbox needed a slight weight increase to accommodate increased torque and gear ratio
NACELLE			
Bedplate	\$66,071	\$67,158	Bedplate cost rose slightly despite reduced thickness because of impact of increased radius in the WindPACT formula
Yaw System	\$12,092	\$7,928	Yaw system cost dropped according to the reduction in peak yaw moment
ELECTRICAL			
AOA Sensors	\$0	\$2,000	Specific sensor technology must still be developed; thus this cost is an educated estimate of the final sensor cost to production turbines
Backup Power	\$0	\$2,000	Based on a biodiesel generator for a 50-MW wind farm to provide blade pitch control power for the entire fleet
BAL of STATION			
Foundation	\$63,475	\$65,098	Foundation cost rose according to the increase in peak tower moment; active tower control should help

The impact of these cost changes increased the initial capital cost (ICC) from \$1,365,359 to \$1,392,022, an increase of 2.0%. The enhanced energy production created by the larger diameter rotor more than offset the modest cost rise, resulting in a reduction in COE from \$0.0426/kWh to \$0.0399/kWh (a drop of 6.3%).

## 5.0 MARKET INTEGRATION

### 5.1 Enabling Technology

The advanced independent pitch control system relies on various technologies. Some of the technology is familiar to the wind industry, some can be adapted from other industries, and some must be developed specifically for this application.

Because the intent of the advanced control is to modulate before damaging loads occur and before an opportunity for increased energy capture has passed, the Controller processor and components

must be able to perform quick real-time processing. This may involve an upgrade of the standard controller.

Sensing angle of attack on an operating wind turbine presents some unique challenges. No off-the-shelf sensor suitable for the advanced control currently exists, yet several technologies seem suitable, and sensor manufacturers are eager to develop an appropriate product.

Pitch drives will experience a higher duty cycle with the advanced control, but this is within the capabilities of current pitch systems. The advanced control has been designed to stay within the limitations of standard pitching systems. If at some point a faster or otherwise improved pitch system is required for better control, the components are readily available.

It may be that loads (while parked in a storm condition) are a limiting factor. If so, then operating the advanced control while parked, and potentially during a power outage, will become desirable. Backup power solutions are available and can be quite reliable (borrowing from the hospital industry) but will present an extra expense and need to be adapted to the wind farm environment.

All of these technological requirements are obtainable in the near future with a modest amount of development, and they do not hinder furtherance of the advanced control.

### ***5.1.1 Controller Hardware***

The advanced blade pitch control algorithm should be implemented on existing turbine controllers, including high-performance 32-bit microcontrollers and DOS-based industrial Pentium-style processors.

### ***5.1.2 Angle of Attack Sensors***

The advanced control being researched relies on accurate and timely measurement of AOA to minimize loads on a wind turbine and maximize energy capture. Such a sensor must be affordable (priced favorably against the COE decrease due to the advanced control), rugged (able to handle continuous harsh environment exposure with only periodic maintenance), innovative (Coriolis forces, aerodynamic disturbance over the blade, soiling, icing, etc. are all factors specific to use on a wind turbine rotor), and technically sufficient (accuracy, precision, resolution, delay, etc. must be sufficient to drive the advanced control algorithm). Available AOA-sensing technologies were surveyed and their suitability analyzed in the section presented here.

A specification was developed of required attributes of an AOA sensing system for use with a wind turbine and the advanced control system. The specification, based on the work of Gordon Rouse of Honeywell [6], is presented in Table 8. Technologies were identified and at least one supplier for each identified technology was contacted. The suitability of that technology was analyzed in comparison to the draft specification.

**Table 8: AOA Sensor Requirements**

<b>Parameter</b>	<b>Requirement</b>
Accuracy	+/- 0.1 Degrees
Precision	+/- 0.1 Degrees
Distance constant	< 3 meters
Range	-10 to 45 Degrees
Sample rate	5 Hz or faster
Delay	10 ms or less
Calibration	Individual sensor calibration not required
Life expectancy	20 years or 5 years with replacement strategy (i.e. how to access blade)
Temperature range	-20 to 150 °F operating
Humidity range	0 to 100% RH
Wind speed range	2 mph to 300 mph?
Cost	To be specified by vendor
Physical size	Does not cause significant drag, depending on mounting location
Bug/soiling survivability	Must operate under soiling/bug build up of 0.05 inches
Ice survivability	Must operate with ice accretion of 0.25 inches, or have heating capability
Rain survivability	Must continue to operate when wet
Power draw limitation	None
Maintenance requirements	Quarterly at most
Mounting	To be specified by vendor
<b>Software Requirements</b>	
Data output format	Digital preferred, analog okay
Sensor fault	Detection
Sensor drift	Compensation
Sensor calibration	Periodic self-calibration
Iced condition	Detection if operability is affected
Bug/dirt accretion	Detection if operability is affected

Suppliers usually did not have anything suitable available off the shelf, but they were very interested in this application and were willing to adapt or develop products to meet the system needs. They were also eager to participate in round two of the NREL LWST solicitation in partnership with this project to develop the idea and the technology. Table 9 lists AOA sensing technologies and the suppliers contacted with a brief summary of their suitability.

**Table 9: Angle-of-Attack Sensor Options**

Technology	Source 1	Source 2	Strength	Weakness	Suitability
Differential pressure	Honeywell		Internal components		Very good
Drag type	IDI	Inertia	Small	External	Good
Hot Wire	Aerotech	TSI		External, fragile?	Fair
Pressure probe	Custom		Small	External	Good
Laser Doppler	VioSense	TSI	Distance	Expensive	Good
Particle image	TSI		Distance	Impractical	Bad
Vane type	Goodrich	Space Age Control	Common	Reliability external	Poor

The technologies were listed in approximate order of suitability for application on the control of wind turbines.

Differential pressure technology is based on sensing pressure differences between the leading edge and the top and bottom surfaces of the blades. The only external alteration of the blade is a small orifice at each location where pressure and flow is to be sensed. Pressures at and airflows through these orifices are then sensed inside the blade; thus the blade has no external protrusions and is aerodynamically equivalent to a blade without the AOA sensor. This method provides near-zero aerodynamic disturbance and is tolerant to soiling. This technology seems at this stage of analysis to be the most suitable technology available.

The following three technologies also may be suitable to this application. Each involves a small sensor placed in front of the leading edge of the blade. In all cases, the sensor is small enough that aerodynamic disturbances may be insignificant. These sensors may be placed at the end of a boom to detect inflow at a point a small distance in front of the leading edge if such an arrangement is desirable (such as the Unsteady Aerodynamics Experiment conducted by NREL, shown in Figure 14).

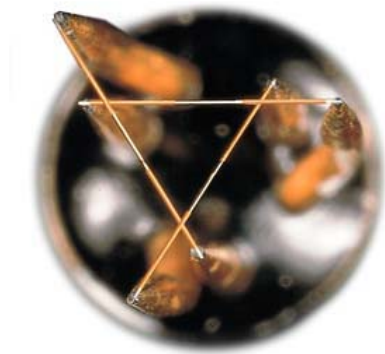


**Figure 14: Leading edge sensor booms (NREL)**

The drag type AOA sensor involves placing a drag object into the oncoming air stream and measuring the intensity and direction of force on the object. IDI has developed a sensor that employs an approximately 1" diameter spherical drag object. The surface of this object is pre-

roughened and is tolerant to a degree of soiling. IDI has also developed innovative ice detection and ice removal technology that may be used in conjunction with this sensor.

Hot-wire-based AOA sensors shown in Figure 15 can be fragile and temperamental, which is why TSI's hot-wire-based system is inappropriate for use with a wind turbine. The Hot Wire system provided by Aerotech has a much more rugged arrangement with stainless steel sheaths around the elements.

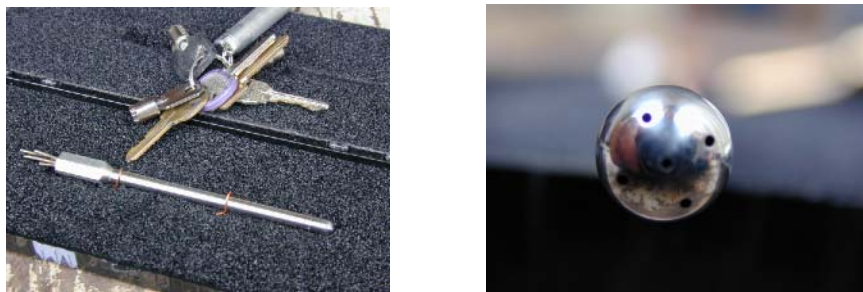


**Figure 15: Hot wire flow sensor**

A thermal anemometer measures fluid velocity by sensing changes in heat transfer from a small, electrically heated sensor (wire or thin film) exposed to the fluid under study. The heated sensor is held at a constant temperature using an electronic control circuit. Increasing the current flow to the sensor compensates for the cooling effect resulting from the fluid flowing past the sensor.

The magnitude of the current increase needed to keep the temperature constant is directly related to heat transfer, and thus flow velocity. This parameter is available as an output voltage. By using multiple hot wires, each with different orientation to the flow, the local flow direction can be measured.

The pressure-probe-based system in Figure 16 may also be employed similarly to that used on NREL's Unsteady Aerodynamics Experiment [7]. These pressure probes rely on a pressure transducer that measures the pressure and pressure difference at each of several orifices around a hemispherical tip of the external probe. This type of probe can give sideslip as well as AOA data.

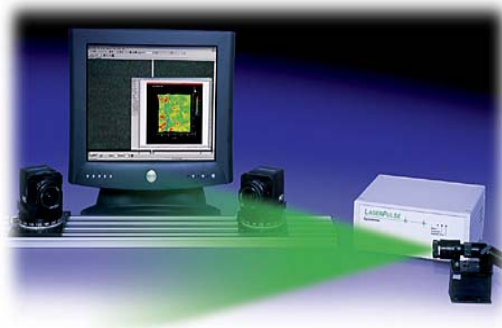


**Figure 16: Pressure probe sensor (NREL)**

Laser Doppler Velocimetry (LDV) and Particle Image Velocimetry (PIV) can detect air speed and direction at a distance from the hardware location. With sufficient warning of inflow

properties, an anticipative control algorithm could be devised that would be superior to the reactive algorithm necessary for use with a blade-mounted sensor requiring local airflow.

PIV systems (Figure 17) measure velocity by determining particle displacement over time using a double-pulsed laser technique. A laser light sheet illuminates a plane in the flow, and the positions of particles in that plane are recorded using a digital or film camera. A fraction of a second later, another laser pulse illuminates the same plane, creating a second particle image. From these two particle images, PIV analysis algorithms obtain the particle displacements for the entire flow region imaged and gives velocity information at hundreds or thousands of locations. Flow properties such as vorticity and strain rates may be obtained for the entire region. Other properties, such as mean, turbulence, and other higher order flow statistics, can also be obtained. A PIV system to meet the specification would need particles generated upstream to detect airflow properties. The PIV systems investigated are also prohibitively expensive. Thus PIV systems are not suitable for AOA detection for an operating wind turbine.



**Figure 17: Particle image velocimetry system**

LDV systems (Figure 18) are a proven technique that measures fluid velocity accurately and non-invasively. Laser light illuminates the flow, and light scattered from particles in the flow is collected and processed. In practice, a single laser beam is split into two equal-intensity beams, which are focused at a common point in the flow field. An interference pattern is formed at the point where the beams intersect, defining the measuring volume. Particles moving through the measuring volume scatter light of varying intensity, some of which is collected by a photo detector. The resulting frequency of the photo detector output is related directly to particle velocity. If additional laser beam pairs with different wavelengths (colors) are directed at the same measuring volume, two and even three velocity components can be determined simultaneously.



**Figure 18: Laser Doppler velocimetry system**

LDV systems do not require generated particles to operate; normal airborne particles are sufficient. The devices are, however, quite expensive. A system proposed by TSI would involve an LDV system on each blade and would be vulnerable to soiling as the “windows” through which the laser is projected becomes opaque. This system is most likely inappropriate to the wind turbine environment. An alternate system proposed by Viosense uses a single high-powered, long-range carbon dioxide laser to detect wind velocities in a full field in front of the rotor. This type of system would allow for an anticipative control algorithm, which could reduce loads and increase energy capture beyond that provided by a reactive control strategy. The production cost of the system is estimated to be on the order of \$80,000, which may be prohibitive depending on the degree of benefit that can be realized. If modeling and computer analysis show a dramatic benefit to having an anticipative control system, this technology should be pursued further. A laser-based anemometry system for use with wind turbines is disclosed in U.S. patent #6,320,272 by Ladling, et al., but this system is only used to determine wind speed at one point in front of the rotor to eliminate error in wind speed experienced by nacelle-mounted cup anemometers.

The most basic type of angle of attack sensor is the rotating vane type sensor shown in Figure 19, of which there are several manufacturers. The technology is based on the common wind vane with a potentiometer or other electrical signal device to detect the position of rotation. Inertia, damping, and aerodynamic shape are carefully selected to inhibit harmonic oscillation and create a steady true reading. This technology has several disadvantages, including disruption of airfoil aerodynamics, vulnerability to soiling, mechanical failure, and reactions to Coriolis forces. The two manufacturers contacted were not optimistic about the suitability of this type of AOA sensor for a wind turbine application.



**Figure 19: Rotating vane sensor**

Other technologies exist as well; some have been identified and some were undoubtedly missed. One promising technology, of which insufficient data could be obtained, involved a pressure-sensitive film or flexible pad which, when placed on the leading edge of the blade, could detect AOA via the variable pressure over its surface. One approach to this is to etch a grid of piezoelectric sensors out of a single silicon sheet, integrated with a small microprocessor and embedded in a flexible matrix. Potential advantages of this technology are negligible aerodynamic disturbance, solid-state reliability, and relatively low cost. A potential disadvantage is signal drift over time, as these sensors measure time-varying signals well but fall short for static and quasi-static signals. This technology may merit further attention in subsequent stages of AOA control development.

Considering the variability of technology available and the eagerness of the manufacturers to reach new markets, it is likely that a solicitation for a suitable sensor could make several technologies available in a short period.



### **5.1.3 Pitch Drives**

Pitch drives on 1.5-MW turbines built today can achieve between 7 and 12 degrees per second maximum pitch rate with good acceleration. The advanced control has been designed with this pitch rate in mind so that standard pitch drive systems may be used with the advanced control. If a faster pitch rate is desired, it is technically feasible to design such a system with available servo motors and gear systems.

The advanced control system also increases the pitch drive duty cycle substantially, as seen previously in Figures 2 and 3. This will have some impact on both the design and maintenance requirements for the drives. A short report on pitch drive systems is included in Appendix D.1.

### **5.1.4 Backup Power for Controller and Pitch System**

Backup power requirements were analyzed for running the Advanced Control pitch drive continuously through a high wind speed and power outage event. The pitch control investigation may reveal that storm survival loads are the limiting load case and the only way to reduce the weight or increase the size of the turbine via better control will be by reducing loads when the blades are in a feathered position during storm conditions. Part of the IEC safety standard [1] requires an assumption that utility power is not available during 50-year storm conditions.

The first step in estimating the amount of energy storage necessary is to quantify the power draw of the pitch drive system and set an amount of time that the system is to be self-sufficient. A 7-day maximum duration power outage is specified in the IEC standard [1], but it is unlikely that extreme winds would be present for the entire duration of this 7-day outage. No applicable duration is specified for extreme winds in conjunction with the power outage. Therefore, a 1-day or 24-hour period was chosen as the arbitrary period of self-sufficiency for this condition. Any other duration of energy storage can be scaled linearly to the results below.

Three independent methods were used to obtain an estimate of power draw by the pitch system, and details can be found in the full backup power report listed as Appendix D.2. Since no single method is considered authoritative, all three methods were averaged for a result of 1300 watts per blade while pitching in high wind. Thus, for a three-bladed wind turbine autonomous for 24 hours and a safety factor of 1.5, 138 kilowatt-hours of energy storage would be required. This could be in the form of a 10,000-pound battery bank at the base of the tower costing \$11,000. As an alternative to minimize the potential component failure chain, a much smaller battery bank in the hub was considered. Though it cost much less (~\$600), only 2 hours of autonomy could be provided under full-power draw. It is conceived that a power-saving mode of control could be employed during a power outage situation in which, instead of minimizing loads in general, the controller would only provide enough motion of blade to avoid damaging loads.

The most robust backup power option examined was to use the in-hub battery bank described above, plus a thermal generator central to the wind farm and UPS for charging the batteries giving longer autonomy, limited only by the size of the generator fuel tank. This option adds approximately \$2000 per turbine capital cost and some small increase in maintenance requirements for the generator.

The acceptability of the backup power options considered and the control strategy is directly tied to the consequences that result from failure. If the loss of a pitch actuator, slip ring, or inverter means that the wind turbine is destroyed (or in the case of a generator, that the whole wind farm is destroyed), then that risk, no matter how small, is unacceptable. However, if the consequence of failure is the loss of a blade or some other component, and if the risk is rather small, then it is

more likely that such a system will be acceptable because such risks are currently accepted for commercial wind turbines.

It is possible to provide backup power for long-term continuous operation of pitch drives to minimize structural loads during a utility outage and a 50-year storm condition. A generator with UPS and an in-hub battery bank is the best solution analyzed to provide a 24-hour period of coverage. However, such a backup system involves cost and structural commitments, which should be weighed against the benefits of reduced design margin resulting from relying on such a control strategy. The level of acceptance in the marketplace will be directly tied to the consequences of failure of the backup power and control system.

### ***5.2 Performance, Costs, and Figures of Merit***

As discussed in section 4.2.2, the AEP at a surface wind speed of 5.8 m/s was 5,999,000 kWh, a 10.2% increase over the baseline performance. As shown in Section 4.3.2, system costs were well managed, including expected small increases for blades, sensors, and backup power, plus slight increases for gearbox, bedplate, and foundation, and a reduction for the yaw system costs. A breakdown of baseline and final turbine costs and COE follows in Table 10.

**Table 10: Baseline and Final Turbine Cost Breakdown**

<b>Component</b>	<b>Baseline Cost (\$)</b>	<b>Final Cost (\$)</b>	<b>Unit Cost (\$/kWh)</b>	<b>% Total</b>
<b>Rotor</b>	<b>236,154</b>	<b>257,938</b>	<b>0.0051</b>	<b>12.9</b>
Blades	152,399	174,184		
Hub	48,206	48,206		
Pitch mechanism, bearings	35,548	35,548		
<b>Drivetrain, Nacelle</b>	<b>536,131</b>	<b>535,318</b>	<b>0.0107</b>	<b>26.7</b>
Low-speed shaft	19,857	19,857		
Bearings	12,317	12,317		
Gearbox	160,688	162,951		
Mech brake, HS coupling	2,985	2,985		
Generator	78,000	78,000		
Variable-speed electronics	81,000	81,000		
Yaw drive and bearing	12,092	7,928		
Mainframe	66,071	67,158		
Electrical connections	60,000	60,000		
Hydraulic system	6,750	6,750		
Nacelle cover	36,373	36,373		
<b>Control and Safety System</b>	<b>10,200</b>	<b>14,270</b>	<b>0.0003</b>	<b>0.7</b>
<b>Tower</b>	<b>179,500</b>	<b>179,500</b>	<b>0.0036</b>	<b>8.9</b>
<b>Balance of Station</b>	<b>403,374</b>	<b>404,996</b>	<b>0.0081</b>	<b>20.2</b>
Foundation	63,475	65,098		
Transportation	51,004	51,004		
Roads and civil works	78,931	78,931		
Assembly and installation	50,713	50,713		
Electrical interconnection	126,552	126,552		
Permits and engineering	32,698	32,698		
<b>Initial Capital Cost (ICC)</b>	<b>1,365,359</b>	<b>1,392,022</b>	<b>0.0277</b>	<b>69.4</b>
<b>Annual Energy Production</b>				
Ideal AEP (kWh)	5,446,000	5,999,000		
Availability	0.95	0.95		
Soiling losses, etc.	0.02	0.02		
Array losses	0.05	0.05		
<b>Net AEP (kWh)</b>	<b>4,817,000</b>	<b>5,306,000</b>		
Replacement Costs (LRC) (\$/yr/unit)	22,500	22,500	0.0042	10.6
Fixed Charge Rate (FCR)	0.106	0.106		
Operations and Maintenance (O&M) (\$/kWh)	0.008	0.008	0.0080	20.0
<b>COE (\$/kWh) = O&amp;M + ((FCR*ICC)+LRC)/AEP</b>	<b>0.0426</b>	<b>0.0399</b>	<b>0.0399</b>	<b>100.0</b>

### **5.3 Other Market Considerations**

Other market considerations include visual impact, noise, capacity factor, grid interaction, shipping, safety, and public perceptions. Visual impact will increase slightly with the longer blades. Acoustic noise essentially will be unchanged because maximum tip speed was not allowed to change. Capacity factor will increase as the larger rotor captures more energy at lower wind speeds, without changing the nameplate capacity. Grid interaction will be unchanged from the baseline. Shipping will be more problematic with the longer blades. Safety will be unchanged from the baseline, as the final design is required to meet the IEC safety standard. Public perceptions are rife with uncertainty, but there is no visible change with this concept over the baseline.

### **5.4 Turbine Development Plan**

This project demonstrated the theoretical advantage of the advanced independent blade pitch controller, along with a survey of the practical elements of implementing it. Taking the concept to market will require several further steps, each of which will be discussed further below:

1. Detailed Engineering Design
2. Proof-of-Concept Testing
3. Prototype and Demonstration
4. Market Introduction.

#### **5.4.1 Detailed Engineering Design**

While the concept has been demonstrated computationally, the new elements introduced here require further detailed engineering work to ensure a robust and reliable loads mitigation package. Such work is expected to take approximately 1 year and include the following tasks:

- Add tower flexibility to the dynamic model and control strategy
- Examine the potential benefit of including yaw rate in the control strategy
- Improve the control tools and simplify their overall design procedure
- Determine the most effective control parameters through an expanded sensitivity study
- Optimize the sensor types, locations, and influence strategy
- Engineer the sensor mounting and interconnection
- Develop a procedure for modifying existing control software
- Develop a procedure for acceptance and safety testing
- Develop a procedure for performance and loads testing
- Revise the design as necessary to ensure a robust and reliable system for advanced independent blade pitch control.

#### **5.4.2 Proof-of-Concept Testing**

Proof-of-concept testing can be performed on an existing wind turbine without extended blades because such a platform still offers the opportunity to perform functionality, performance, and loads testing. Such a platform should already have available a solid baseline of operational data; alternatively, two similar platforms could be tested side-by-side, with one each in the baseline and modified configurations. A proof-of-concept test should take about 1 year, with 3 months for component design, 3 months for component acquisition and installation, 3 months for the testing outlined above, and 3 months for reporting.

### ***5.4.3 Prototype and Demonstration***

After proving the concept and refining the design, the next step would be to test the system on a wind turbine with an extended rotor. This case does not offer substantial new learning opportunities over the proof-of-concept testing, so it probably makes more sense to move this phase into a system demonstration phase. A demonstration phase would include several turbines, possibly of different designs and preferably in different wind regimes. This phase would require 1 year or longer of testing, with further loads data collection and analysis on at least one of the demonstration units.

### ***5.4.4 Market Introduction***

Full disclosure of test results and issues from the previous three steps provides an essential backdrop to market introduction. Further effort will be required to identify potential applications and market opportunities and to develop relationships with wind turbine manufacturers and other interested parties. Although one might be inclined to imagine an effort to develop a wind turbine from the ground up that incorporates advanced independent blade pitch control, it is a daunting proposition in the face of a rapidly maturing industry with several well-capitalized manufacturers. A more likely approach will be to introduce the concept to existing wind turbine products through a combination of consulting and licensing. This approach also recognizes the potential utility of advanced independent blade pitch control to wind turbines of all designs and sizes.

## **6.0 CONCLUSIONS**

This study demonstrated that advanced independent blade pitch control based on blade AOA measurement is advantageous for loads reduction, allowing blade extension with minimum impact on cost, thus reducing COE relative to a similarly rated state-of-the-art wind turbine. Although relatively involved, the system is technically feasible to manufacture. The resulting load impacts were varied but favorable overall.

When applied to the WindPACT baseline 1.5-MW wind turbine, advanced AOA independent blade pitch control was able to reduce various structural loads between 5% and 30%, accommodate a 10% blade extension, and provide a net COE reduction of 6.3%, from \$0.0426/kWh to \$0.0399/kWh.

Additional hardware required includes blade-mounted inflow sensors and signal wiring, extended blades, and a backup power system for grid outage, using either an onboard battery bank or a ground-based biodiesel generator.

The WindPACT design methodology utilized in this project is highly sophisticated in its modeling and analysis tools, yet it is still constrained by a set of assumptions that limit its soundness for practical application. This presents, at times, an awkward precision imbalance between detailed, time-intensive tools and coarse approximations. It will be critical for future design efforts to complete certain improvements to the design methodology, including:

- Correct the blade property calculations and load application techniques
- Utilize full three-dimensional (or two at minimum) stress state analyses
- Revisit the selection of blade fatigue strain limits and S-N curve application
- Include 25%, 50%, and 75% spanwise station loads for all blades
- Extrapolate the fatigue curves and utilize the extrapolated curves in the lifetime and margin analyses

- Develop and apply a clear approach to stress concentrations and derating factors for each component location
- Develop a user-friendly interface for controller design and tuning.

Future work on the advanced blade pitch control study is necessary to understand its full potential, including:

- Extend the control system to include a flexible tower in the dynamic model and the control and estimation system
- Examine the potential benefit of including yaw rate in the control strategy
- Improve the control tools and simplify their overall design procedure
- Determine the most effective control parameters through an expanded sensitivity study
- Optimize the sensor types, locations, and influence strategy
- Field-test the concept on a wind turbine test platform with comparative loads and performance measurements.

## REFERENCES

1. International Electrotechnical Commission (1998). *IEC 61400-1e2: Wind Turbine Generator Systems – Part 1: Safety Requirements*, 2<sup>nd</sup> Edition, International Standard 61400-1.
2. Stol, K. (2001). *Dynamics Modeling and Periodic Control of Horizontal-Axis Wind Turbines*, University of Colorado Department of Aerospace Engineering Sciences, Doctoral Thesis.
3. Laino, D.; Hansen, C. (2002). *User's Guide to the Wind Turbine Aerodynamics Computer Software AeroDyn*, NREL Contract TCX-9-29209-01, <http://wind.nrel.gov/designcodes/aerodyn>.
4. Malcolm, DJ; Hansen, C. (2002). *WindPACT Turbine Rotor Design Study: Final Report*, NREL Contract YAT-0-30213-01.
5. Griffin, D. (2002). *Scaling Of Composite Wind Turbine Blades for the WindPACT Rotor Design Study*, Kamzin Technologies (now Global Energy Concepts).
6. Rouse, G. (2003). *Draft angle of Attack Sensor Requirements and Responses*, Honeywell.
7. Hand, M. M.; Simms, D. A.; Fingersh, L. J.; Jager, D. W.; Cotrell, J. R.; Schreck, S.; Larwood, S. M. (2001). *Unsteady Aerodynamics Experiment Phase VI: Wind Tunnel Test Configurations and Available Data Campaigns*. 310 pp.; NREL Report No. TP-500-29955.
8. Bossanyi, EA. (2002). *Individual Blade Pitch Control for Load Reduction*, Wind Energy 2003; 6:119-128, John Wiley & Sons, Ltd.

## **APPENDICES**

Appendix A.1: SymDyn Development and Lessons

Appendix A.2: Copy of File Inputprops.M Used to Create the SymDyn Model

Appendix A.3: Copy of AeroDyn Input File Used for the ADAMS and SymDyn Simulations

Appendix A.4: Copy of SNLWind-3d Summary File Used for the ADAMS and SymDyn Simulations

Appendix B.1: SymDyn-to-Adams Conversion of Independent Blade Pitch Control Using Angle of Attack Sensors and a Periodic Estimator and Controller

Appendix B.2: Fortran 90 Source Code for Pitch Control Algorithm (ADAMS Subroutine GSESUB)

Appendix B.3: Excerpt of the ADAMS Model Dataset Pertaining to the Pitch Control System

Appendix C.1: Bladestructuralcalc.Doc for WindPACT Rotor Design

Appendix C.2: WindPACT Design and Analysis Tools

Appendix C.3: DEC Cost of Energy Table, Final Design

Appendix D.1: Pitch Drive Systems

Appendix D.2: Backup Power Storage for Pitch Drive Systems



## **APPENDIX A.1: SYMDYN DEVELOPMENT AND LESSONS**

Karl Stol wrote the SymDyn code. He supplied a code and input file for the two-bladed Controls Advanced Research Turbine (CART) (downwind) that produced constant controller gains and a full-state estimation. The only wind disturbance implemented was hub-height wind speed. Karl supplied a rough draft of a SymDyn module that produced a periodic estimator and periodic control gains. This module required adjustments and extensions before integration with SymDyn.

A SymDyn wind turbine control model is assembled in the Simulink graphical overlay to MATLAB. The SymDyn model is built from a collection of software modules, which include MATLAB gateway modules for I/O (“mex files”), MATLAB callable routines (“m files”), and Simulink routines. The following improvements and changes were made:

### ***WindPACT***

Windward Engineering supplied a SymDyn input file describing the WindPACT turbine. E3-Design debugged SymDyn while implementing the WindPACT turbine model. Some of the parameters required adjustment.

### ***Sensors***

SymDyn was expanded to a broader range of sensor inputs, including independent sensors on every state (parameters that define the state of the wind turbine) and independent sensors on the disturbance states (parameters that define the state of the wind). It can also handle sensors whose output is coupled to multiple states and disturbance states.

### ***Two-Sided Derivative***

The original SymDyn code used a single-sided differencing scheme to calculate the linearized dynamic equations. More accurate derivatives can be found using a two-sided differencing scheme. The two-sided derivative takes more than twice as long to calculate and produces more precise values for the linearized equations of motion (A, B, Bd matrices). The time is reasonable though, and the precision appears to be important to the stability of the periodic state estimator. Therefore, we will use the two-sided derivative implementation.

### ***Numerical Methods***

The integration methods proved difficult. First, in the SimuLink domain of SymDyn, there are several options for integrating the dynamic equations through time. We found that the size of the time step is important to the accuracy and stability of the time integration. Also, some cases performed better with a stiff equation formulation of the integration, while others preferred the “standard” explicit Runge-Kutta. We have not established a clear guideline yet, but it is important to be aware of these possibilities. If the system is producing chaotic or divergent behavior, it can be caused by the integration method, rather than the system dynamics.

Second, the SymDyn code performs other integrations, namely to calculate the periodic estimator and controller gains. Sometimes these integrations were inaccurate or unstable, depending on the values of the weights chosen for the controller and estimator design. By increasing the number of steps in the integration, it was possible to ensure accuracy and stability. However, a useful improvement to the program might be to use a more stable, adaptive, or implicit matrix integration scheme, rather than a fixed and uniform step size.

## ***Weighting***

One of the important elements in controller design is the weighting selection required in linear quadratic estimator design and linear quadratic regulator design. Choosing these weights is a bit of an art. Many simulations were conducted in SymDyn to find values of these weights that produced good performance in the wind turbine. The various numerical instabilities discussed earlier sometimes made it difficult to determine the true source of anomalous results. For instance, was a chaotic or divergent response of the wind turbine caused by the integration error, the estimator implementation, accuracy of the dynamic equations, errors due to linearization of a nonlinear plant, or the weights used in the controller or estimator?

## ***Wind Input***

In the original version of SymDyn, the code was written in a general format to allow for expansion. However, several modules in the code required modifications to allow for more than just hub-height wind speed as a wind disturbance. It was critical to this project that we include more components of the wind disturbance. For this study, we implemented vertical and horizontal linear wind shear. Vertical linear wind shear was chosen over power-law vertical wind shear because the former was believed to be more suitable for a linearized disturbance matrix. Additional components of wind disturbance familiar to AeroDyn-based programs were added, including wind direction and vertical wind speed, but they were not useful.

## ***Periodicity***

Another difficulty discovered in the original version of SymDyn involved periodicity. As stated above, the initial periodic controller and estimator code were rough and not fully debugged. In the process of fully implementing the periodic controller and estimator capacity in SymDyn, some problems were discovered with the method and point about which linearization of the dynamic equations of motion took place. In finding the periodic equations of motion, the first step is to find a periodic solution to the motion of the wind turbine. Previously, this was taken as the motion that resulted with steady 18 m/s hub-height wind speed and 0.2 linear vertical wind shear at fixed pitch. The vertical wind shear (and other factors) caused a periodic response to the wind turbine that was not insignificant. When this operating point is chosen for the point of linearization, the controller must then adjust the pitch to create this periodic response. In other words, the controller does not seek to eliminate this periodic response through periodic control, and thus the response of the turbine cannot improve upon a fixed-pitch turbine in the same wind conditions.

The original code used a steady response (canceling the flapping velocity of the blades) as the reference point for the states and not the periodic blade motion that occurred when the linearized equations of motion were calculated. In this way, the original code attempted to force the controller to reduce the periodic motion due to wind shear. The problem with this method was that the estimator was unstable and had difficulty estimating the state of the wind turbine, even with a full array of independent state sensors in effect.

The solution was to use the periodic response as the reference point for the states, produce good state estimates, and then inject a periodic canceling command to the controller in addition to the command to minimize the differences between the estimated state and the periodic state used in the linearization. The result was a stable and accurate estimator and a controller capable of reducing the dynamics much below the periodic response of the linearization.

### ***Wind Shear Simulations***

Simulations in the SimuLink portion of SymDyn show very good performance. We instructed the linearization to take place at an operating point of 18 m/s wind speed with no wind shear. Even under these conditions, the fixed-pitch response of the wind turbine has periodic components (suspected to be caused by shaft tilt angle and gravity). After injecting post-estimator controller commands to cancel this periodic motion, the turbine was able to run with almost no periodic motion (other than the quasi-rigid body rotation of the rotor, of course). The response of the turbine was almost unchanged during a ramp in wind shear from 0.0 at time 0 seconds to 0.75 at time 15 seconds. This is a huge variation in wind shear over a relatively short period, and the perturbation of the rotor is negligible. A second case involved comparing the response of the turbine in a 0.4 vertical linear shear and steady hub-height wind, with and without vertical wind shear information to the controller. The results showed large vibrations when the vertical wind disturbance was ignored and very good performance with it included.

### ***Horizontal Wind Speed Simulations***

The controller showed a less impressive response to variations in the hub-height wind speed. For relatively small variations in hub-height wind speed, the turbine/controller model performed satisfactorily, but for larger changes in wind speed, the performance of the controller was degraded. Therefore a gain-scheduling method was implemented to accommodate hub height wind speed variations.

### ***Steady State Solution and Compensation***

The design of the control law is based on linear control theory. As such, it requires a periodic steady state or operating point solution to the motion of the wind turbine. Deviations from this operating point are incorporated into the control law using linear operators. The control law tries to make the wind turbine return to this operation point. The operating point in Symdyn is taken at a fixed blade pitch and a constant rotor average rotational speed. Even if the operating point is in steady uniform wind, the turbine displays a periodic dynamic response due to the gravity loads and shaft tilt.

Various efforts were made to find ways of canceling this motion by active control of the blade pitch. These efforts were complicated because they affect the accuracy and stability of the state estimator (the part of the control law that uses all known information to construct an estimate of all the important variables to the controller). This type of cancellation is outside standard linear control theory. At times it has helped the performance of the control system, but at other times it has been the source of problems (inaccurate estimates, unstable estimates). At this point, this active cancellation of the periodic operating point is an option that can be easily switched on or off in the SymDyn simulations.

### ***Implement Angle of Attack Subroutine Calls into SymDyn***

The SymDyn program makes subroutine calls to the Fortran AeroDyn subroutines. Originally, these subroutines returned only the forces and moments on the blades. Windward Engineering modified the original subroutines so that they would also return the Angle of Attack (AOA) at each blade station on each blade. SymDyn was modified to use these new subroutines. We verified the output to make sure we understood the changes and notation.

The new subroutines allowed new calculations sections to be added to SymDyn for calculating the matrices relating state variables, control inputs, and disturbance states to state variable rates of change and sensor outputs.

### ***Perturbation Sizes***

The linearized equations derived numerically from the nonlinear equations require perturbations to calculate the elements of the various matrices. The proper size of the perturbations was unknown. Accurate derivatives favor small perturbations; however, small perturbations favor accuracy when the turbine is operating close to the equilibrium operation point at the expense of accuracy when operating in states further from equilibrium. Larger perturbations (as long as they are not too big) favor accuracy further from the operating point, while sacrificing accuracy close to the operating point. Each state variable (i.e. blade flap angle, RPM, etc.), each control input (blade pitch), and wind disturbance (i.e., horizontal wind speed, wind direction, wind shear, etc.) must be perturbed in the process of calculating the required matrices. Several simulations were conducted to determine appropriate values for these perturbations.

### ***Actuator Dynamics***

Originally, SymDyn did not account for actuator dynamics. In other words, the assumption in SymDyn was that the commanded pitch angle of the blade is the actual pitch angle of the blade. In a practical implementation of the controller on a wind turbine, pitch actuator dynamics must be considered.

The WindPACT turbine used a transfer function to describe the relationship between pitch command and actual pitch angle. The relationship was that of a closed loop servo in which the natural frequency of the actuator was four times the rotational frequency of the rotor.

To test the effect of actuator dynamics on the performance and stability of the control system, the commanded pitch angle signal was routed through the transfer function before being applied to the dynamics simulation of the wind turbine. Initial tests revealed that the addition of the pitch actuator dynamics resulted in unstable operation. The natural frequency of the pitch actuator was increased, and yet the problem remained. This was confusing because intuition tells us that in the limit of a very fast actuator, pitch dynamics would be negligible.

Eventually, we concluded that pitch actuator dynamics needed to be included in the formulation of the fundamental dynamic equations used in SymDyn. We had been using SymDyn models with eight states (rotor rotational position, blade flap angle for each of three blades, and angular velocities for each). To include pitch dynamics, the control input was changed from pitch angle to pitching torque applied to each blade. This means that blade pitch angles and pitch rates become state variables, adding six states for a total of 14 state variable and three input variables (pitching torque for each blade).

This implementation of SymDyn has the advantage of being able to minimize the required blade torques and thus power and size of pitch actuator hardware. We found that through adjustment of the weights (see weighting section of this report), the peak torques could be reduced significantly. It is our view that torque, not pitch rate, will actually be most important to the cost and feasibility of a pitch actuator.

After running this new version of SymDyn, we discovered that this formulation does not allow for Disturbance Accommodating Control (DAC). When the control inputs were the pitch angles, there were linear controls that could be applied to the wind disturbance state estimates to counteract the effect of the disturbances on the turbine. Unfortunately, there were no useful linear controls that could be applied to the wind disturbance state estimates to give torque commands that would counteract the effect of the wind disturbances.

We tried to find another way of creating a DAC system. We tried using the same linear control law we used in the pitch-angle-based controller, but applying it to the state estimates of the pitch angle. Previously, the pitch angles were control inputs. Unfortunately, the controller was unstable. The reason for this instability was not determined. This is similar to the case of the compensating control that is discussed in the section on steady state compensation above. The failure to produce a stable controller that includes DAC is a serious limitation to using a controller formulated in terms of blade torques. The AOA sensors being considered in this study are thought to be useful in terms of producing disturbance state estimates and turbine state estimates. If the wind disturbance state estimates cannot be used in a torque control formulation of the controller, then a torque control formulation is not very useful to this study.

Thus, we returned to the pitch angle formulation of the controller. We found that the highest pitch actuator natural frequency that we had tried coincided with the natural frequency of the blade in flap. Also, we found some errors in the calculation of the linearized equations of motion. When we recalculated the linearized equations of motion and increased the natural frequency of the pitch actuator to 12 times (or greater) the rotation frequency (three times the WindPACT actuator or greater) of the rotor, the controller was stable and performed satisfactorily.

### ***Rate Limits and Saturation***

A practical implementation of the controller on a wind turbine will have limits on the maximum pitch rate and pitch angle. The effect of these nonlinear features was explored with several simulations by implementing bounds on the pitch rate. No problems were found.

### ***Controller and Estimator Weights***

Linear quadratic control design methods were used that had been extended for periodic systems. The linear refers to the equations used in the control law or the estimate update equations. Thus, the control signal is a linear function of the state estimates and the disturbance estimates. The quadratic refers to the function that is minimized in the optimization of the linear control law. For both the controller and the estimator, two matrices of weights (four total) with several elements in each matrix must be provided to the routines that calculate the control gains.

Choosing good values for the weights is not an easy process. We conducted hundreds of simulations in SymDyn to better understand how the issues of numerical accuracy, scaling, and performance of the controllers and estimators relate to the weights chosen for both the estimator design and the controller design.

### ***Numerical Accuracy and Stability of Integration***

As part of calculating the periodic estimator and controller gains, it is necessary to perform an integration of a matrix equation. Some sets of weights result in equations that are more difficult to integrate than others. Choosing weights that are easy to integrate occurs mostly through trial and error. If the equations are difficult to integrate, we can decrease the step size in the integration (increasing the number of steps). This slows the calculation and works only up to a point. We found that weights that gave better performance tended to be more difficult to integrate. We considered implementing an implicit and more robust and stable integration scheme into SymDyn. This may be a useful (though not trivial) future task. For now, we have managed other solutions (see below).

### ***Scaling***

It is important to properly scale the variables and thus equations to avoid numerical difficulties. We investigated the relative magnitude of the various variables and adjusted the weights and the

scales accordingly. Scaling was helpful for the AOA sensors and the torque variables because they were of quite different magnitudes than the blade flap angles and RPM.

### **Performance**

The weights are part of the measure of the performance of the estimator and the controller. Basically, they are penalties against the various states being different from the steady periodic operating conditions. Through trial and error (SymDyn simulations), we found weights that work well. The task of determining weights must be considered each time the number and type of sensors, state variable, and control inputs are changed. This significance of this will be more apparent in the section on sensors.

### **Sensors**

Given all the possible problems that can develop in writing and debugging a code and in choosing weights, we took a systematic approach to code development in which the simplest case was implemented first, followed by more complicated and difficult cases.

The first step was to get the open loop SymDyn simulation to work. Next, a full-state feedback controller was generated, and weights were adjusted until reasonable performance was achieved. Next, a state estimator was created in which all the turbine states were directly measurable. This is very close to a full-state feedback situation, but it does require the estimator. Next, the number of wind disturbance states was gradually increased. Each of the disturbance states was assumed to be directly and independently measurable. Next, the AOA sensors were included in the estimator, along with a full set of state-variable sensors.

After building a system with many sensors (some of which would be impossible to implement in a real wind turbine), it was time to gradually reduce the number of sensors to see how the performance of the control system was affected. If by eliminating sensors, the controller's knowledge of its state and the state of the wind was significantly reduced, the effectiveness of the control system would also be reduced.

The direct and independent measurement of the wind disturbance states was dropped while maintaining the AOA sensors. At one point, we observed that vertical wind and horizontal wind shear produced similar effects on the AOA sensors, while wind direction change and vertical wind shear produced similar effects on the AOA sensors (though different from vertical wind and horizontal wind shear). We eliminated the wind direction change and vertical wind from the controller formulation because the control system was producing poor results and was nearly unstable.

This change may have had important effects. However, we later found different sets of weights that improved the stability and performance of the control system. Thus, as future work one could reinstate the wind direction and vertical wind into the controller formulation. For the present study, we have used the three wind disturbance estimates.

It is expensive and complicated to measure every state variable defining the turbine. We believed that relatively few sensors, and especially AOA sensors, would provide sufficient information to the estimator and thus the controller that good performance would be achieved. Through numerous simulations and adjustments of weights, we created a system that uses one AOA sensor (blade element 11 of 15) on each blade, a rotor rotational positions sensor (psi), and an RPM sensor. The system is able to accurately determine the wind disturbance and performs nearly as well as the ideal full-state feedback system. This is good news for AOA sensors. Useful future

work would compare the performance of a controller based on blade-flap-position sensors (strain gages) with the AOA-based controller.

## APPENDIX A.2: COPY OF FILE INPUTPROPS.M USED TO CREATE THE SYMDYN MODEL

```

% inputprops.m: Contains the input turbine properties for derivation of SymDyn
parameters
%
%           via the SymDyn preprocessor (SymDynPP.m)
%   SymDyn v1.00
%   Assumes S.I. units
%

ftitle    = '1.5A08 properties (12/02)';           % title for reference

% Geometry and other constants

Nb        = 3;           % number of blades
rigid_hub = 1;           % 0 = free teeter, 1 = locked teeter (for use in
frequency matching)
gearratio = 87.965;      % gearbox gear ratio
precone   = 0;           % blade precone, pos. moves tip downwind [deg]
tilt0     = -5.0;        % nominal tilt [deg]
delta3    = 0;           % teeter axis angular offset (ignored for locked teeter
or Nb>2) [deg]
omega0    = 20.463;      % nominal rotor speed, pos. clockwise when looking
downwind [rpm]

dtheight  = 82.390;      % tower height [m]
dtilt     = 1.610;      % height from tower top to tilt axis, pos. up [m]
dshaft    = 0;           % dist. from tilt axis to shaft axis, normal to shaft,
pos. down [m]
dteeter   = -3.300;      % dist. from tilt axis to teeter axis, parallel to
shaft, pos. downwind [m]
dhradius  = 1.750;      % dist. from teeter axis to blade root, normal to hub
centerline [m]
dbroot    = 0;           % dist. from teeter axis to blade root, parallel to hub
centerline [m]
dblenght  = 33.25;      % blade length from root to tip [m]

% Center of mass locations

cyoke     = 0;           % c.g. of nacelle yoke, measured up from tower top
along yaw axis [m]
cnx       = 0.3;         % c.g. of nacelle, measured down from tilt axis [m]
cny       = -0.016;      % c.g. of nacelle, measured downwind from tilt axis [m]
cHSS      = -0.767;      % c.g. of HSS + generator from tilt axis along shaft,
pos. upwind [m]
cLSS      = -0.719;      % c.g. of LSS from tilt axis along shaft, pos. downwind
[m]
chub      = 0;           % c.g. of hub from teeter axis, measured upwind along
hub center [m]

% Masses

myoke     = 0;           % mass of yoke [kg]
mnac      = 38786;       % mass of nacelle + nonrotating parts of generator and
shaft bearings [kg]
mHSS      = 2105;        % mass of HSS + rotating generator parts [kg]
mLSS      = 10279;       % mass of LSS [kg]
mhub      = 15148;       % mass of hub [kg]

% Moments of inertia (MOI's)

Iyokex    = 0;           % MOI of yoke in {yoke} frame [kg.m^2]

```



```

Iyokey = 0; % "
Iyokez = 0; % "
Inacx = 3.33e4; % MOI of nacelle and all nonrotating gen. parts in
{nac} frame [kg.m^2]
Inacy = 2.54e4; % "
Inacz = 3.93e4; % "
IHSSlat = 5.03e2; % lateral MOI of HSS + generator [kg.m^2]
IHSSlong = 52.986; % longitudinal MOI of HSS + generator [kg.m^2]
ILSSlat = 8.72e3; % lateral MOI of LSS [kg.m^2]
ILSSlong = 7.19e2; % longitudinal MOI of LSS [kg.m^2]
Ihubx = 3.46e4; % MOI of hub in {hub} frame [kg.m^2]
Ihuby = 3.46e4; % "
Ihubz = 3.46e4; % "

% Joint and shaft stiffnesses

kyaw = 0; % yaw joint torsional stiffness [N.m/rad]
ktilt = 0; % tilt joint torsional stiffness [N.m/rad]
kteet = 0; % teeter torsional stiffness (ignored for rigid hub or
Nb>2) [N.m/rad]
kLSS = 5.6e9; % LSS torsional stiffness (value <= 0 interpreted as
rigid) [N.m/rad]
kHSS = -1; % HSS torsional stiffness (value <= 0 interpreted as
rigid) [N.m/rad]

% Tower distributed properties
% { x/height (m), mass-per-unit-length (kg/m), I/L (kg.m), GJ (N.m^2), EI
(N.m^2) }
%
% must contain at least two rows, one for x/height = 0.0 and one for x/height =
1.0

tdata = [
0.000 2549.7 9540.0 1.8697E+11 2.4306E+11
0.111 2275.8 7601.2 1.4897E+11 1.9366E+11
0.222 2017.5 5974.0 1.1708E+11 1.5220E+11
0.333 1774.7 4623.2 9.0608E+10 1.1779E+11
0.444 1547.4 3515.6 6.8900E+10 8.9570E+10
0.556 1335.8 2620.1 5.1349E+10 6.6753E+10
0.667 1139.6 1907.6 3.7385E+10 4.8601E+10
0.778 959.1 1351.4 2.6484E+10 3.4430E+10
0.889 794.1 926.7 1.8161E+10 2.3610E+10
1.000 644.7 611.0 1.1974E+10 1.5566E+10
];
mtop = 0.0; % lumped mass at tower top (part of tower not nacelle, e.g.
for yaw bearings)

% Blade distributed properties
% { x/length (m), mass-per-unit-length (kg/m), Iy/L (kg.m), Iz/L (kg.m),
ea_twist (deg),
% EIy (N.m^2), EIZ (N.m^2), chord (m), aero_twist (deg) }
%
% must contain at least two rows, one for x/length = 0.0 and one for x/length =
1.0

bdata = [
0.0000 1447.61 1292.087 1292.087 11.10 7.6815E+09 7.6815E+09
1.925 11.10
0.0211 180.33 160.959 160.959 11.10 1.1699E+09 1.1699E+09 1.890
11.10
0.0526 181.67 148.354 148.354 11.10 1.0923E+09 1.0206E+09 2.041
11.10

```

```

0.1053 183.90 127.344      127.344      11.10  9.6297E+08    7.7188E+08    2.294
11.10
0.1579 186.14 106.335      106.335      11.10  8.3366E+08    5.2314E+08    2.547
11.10
0.2105 188.37 85.326 85.326 11.10  7.0435E+08    2.7440E+08    2.800 11.10
0.2632 178.32 74.265 74.265 9.50    6.1465E+08    2.3457E+08    2.669 9.50
0.3158 168.27 63.204 63.204 7.90    5.2496E+08    1.9474E+08    2.539 7.90
0.3684 158.22 52.144 52.144 6.30    4.3526E+08    1.5490E+08    2.408 6.30
0.4211 148.17 41.083 41.083 4.70    3.4557E+08    1.1507E+08    2.277 4.70
0.4737 138.12 30.022 30.022 3.10    2.5587E+08    7.5235E+07    2.147 3.10
0.5263 122.90 25.417 25.417 2.60    2.1787E+08    6.2495E+07    2.016 2.60
0.5789 107.67 20.813 20.813 2.10    1.7986E+08    4.9755E+07    1.886 2.10
0.6316 92.44  16.209 16.209 1.60    1.4186E+08    3.7015E+07    1.755 1.60
0.6842 77.22  11.604 11.604 1.10    1.0385E+08    2.4275E+07    1.624 1.10
0.7368 61.99  7.000  7.000  0.60    6.5849E+07    1.1534E+07    1.494 0.60
0.7895 51.86  5.753  5.753  0.48    5.4254E+07    9.2738E+06    1.376 0.48
0.8421 41.73  4.507  4.507  0.36    4.2659E+07    7.0132E+06    1.259 0.36
0.8947 31.61  3.260  3.260  0.24    3.1064E+07    4.7525E+06    1.141 0.24
0.9474 21.48  2.013  2.013  0.12    1.9469E+07    2.4919E+06    1.024 0.12
1.0000 11.35  0.767  0.767  0.00    7.8741E+06    2.3129E+05    0.906 0.00
];
naelems = 15;          % number of equilength AeroDyn elements per blade

```

### APPENDIX A.3: COPY OF AERODYN INPUT FILE USED FOR THE ADAMS AND SYMDYN SIMULATIONS

```

WindPact 15A8V07 for Adams11 and AeroDyn12 May, 2002.
SI           Units for input and output [SI or ENGLISH]
BEDDOES     Dynamic stall model [BEDDOES or STEADY]
NO_CM       Aerodynamic pitching moment model [USE_CM or NO_CM]
DYNIN       Inflow model [DYNIN or EQUIL]
SWIRL       Induction factor model [NONE or WAKE or SWIRL]
1.0000E-06  Convergence tolerance for induction factor!! use 1.0e-6 for
linearization 5.0e-3
PRAND       Tip-loss model (EQUIL only) [PRANDt1, GTECH, or NONE]
"C:\DATA\Lange\from eric\WindPACT\wind\12ms"      Hub height wind file name (quoted
string)
84.0        Wind reference (hub) height.
0.0         Tower shadow centerline velocity deficit.
1.0         Tower shadow half width.
-3.5        Tower shadow reference point.
1.225       Air density.
1.625e-4    KinVisc - Kinematic air viscosity
5.0000E-03  Time interval for aerodynamic calculations. !!! 0.0 for
linearization, 5.0e-5 or -3 otherwise
4           Number of airfoil files used. Files listed below:
"C:\DATA\Lange\from eric\WindPACT\AeroData\cylinder.dat"
"C:\DATA\Lange\from eric\WindPACT\AeroData\s818_2703.dat"
"C:\DATA\Lange\from eric\WindPACT\AeroData\s825_2103.dat"
"C:\DATA\Lange\from eric\WindPACT\AeroData\s826_1603.dat"
15          Number of blade elements per blade
RELM       Twist      DR          Chord      File ID   Elem Data RELM and Twist
ignored by ADAMS (but must be present)
1.108      11.10      2.21667   1.949     1         NOPRINT
3.325      11.10      2.21667   2.269     2         NOPRINT
5.542      11.10      2.21667   2.589     2         NOPRINT
7.758      10.41      2.21667   2.743     2         NOPRINT
9.975      8.38       2.21667   2.578     2         NOPRINT
12.192     6.35       2.21667   2.412     2         NOPRINT
14.408     4.33       2.21667   2.247     2         NOPRINT
16.625     2.85       2.21667   2.082     3         NOPRINT
18.842     2.22       2.21667   1.916     3         NOPRINT
21.058     1.58       2.21667   1.751     3         NOPRINT
23.275     0.95       2.21667   1.585     3         PRINT
25.492     0.53       2.21667   1.427     3         NOPRINT
27.708     0.38       2.21667   1.278     3         NOPRINT
29.925     0.23       2.21667   1.129     4         NOPRINT
32.142     0.08       2.21667   0.980     4         NOPRINT

```

## APPENDIX A.4: COPY OF SNLWIND-3D SUMMARY FILE USED FOR THE ADAMS AND SYMDYN SIMULATIONS

```

<<<<< Summary file produced by Ver PC-3.0b Alpha, 07-May-1999 >>>>>

<<<<<< Simulation Created at 21:29 On 06-Nov-2002 >>>>>>

----- Contents of input file snlwind_3d.inp -----

---- | SNLWIND-3D INPUT PARAMETER FILE FOR VERSION PC-3.0 Alpha
---- | WindPACT input file for 70.0m rotor 15A07
---- | Runtime options:
---- | -----
12ms | | Root file name for output files (10 characters max)
3   | | random seed, ISEED
YES  | | Create time series files? YES-NO
YES  | | Store time series data in binary form? YES-NO
YES  | | Clockwise rotation looking downwind? YES-NO (binary files)
---- | -----
---- | Turbine/model specifications:
---- | -----
6   | | Square grid point matrix dimension (even values up to 8)
0.05 | | Time step (seconds)
10.25 | | Length of time series (minutes, 30.250 max) (1815 secs) 10.25
1   | | Number of blocks to be simulated, NAVR.
64.7 | | Hub height, ZHUB (m)
84.0 | | Flow field diameter, RDIAM (m) Enter value about 25% greater than actu
0.0  | | Vertical mean flow (uptilt) angle (degs), VFLOANGL
0.0  | | Horizontal mean flow (skew) angle (degs), HFLOANGL
---- | -----
---- | Meteorological boundary conditions:
---- | -----
IECKAI | | Turbulence model: SMOOTH,WF_UPW,WF_07D,WF_14D,IECKAI,IECVKM
A     | | IEC turbulence characteristic (A or B) (IEC models only)
64.7 | | Height of reference wind speed, UREF (m)
12.5 | | Mean wind speed at reference height, ZREF (m/s)
0.000 | | Gradient Richardson number, RI_NO
0.000 | | Surface roughness length, Z0 (m)
0.000 | | Power law exponent, PLEXP
0.000 | | Friction or shear velocity, USTAR (m/s)
0.000 | | Mixing layer depth, ZI (m)
0.000 | | U'W' cross-correlation coefficient, PC_UW
0.000 | | U'V' cross-correlation coefficient, PC_UV
0.000 | | V'W' cross-correlation coefficient, PC_VW
0.000 | | U component coherence decrement, COHD_U
0.000 | | V component coherence decrement, COHD_V
0.000 | | W component coherence decrement, COHD_W
0.000 | | Coherence exponent, COHEXP

----- Summary of responses to input values -----

12ms | | Root file name for output files (10 characters max)
3   | | ndom seed, ISEED
YES  | | Create time series files? YES-NO
YES  | | Store time series data in binary form? YES-NO
YES  | | Clockwise rotation looking downwind? YES-NO (binary files)
---- | -----
---- | -----
6   | | Square grid point matrix dimension (even values up to 8)

```

```

0.050 | Time step (seconds)
10.250 | Length of time series (minutes, 30.250 max) (1815 secs) 10
12300 | Length of time series (# of samples) [CALCULATED]
1 | Number of blocks to be simulated, NAVR.
64.700 | Hub height, ZHUB (m)
84.000 | Flow field diameter, RDIAM (m) Enter value about 25% grea
0.000 | Vertical mean flow (uptilt) angle (degs), VFLOANGL
0.000 | Horizontal mean flow (skew) angle (degs), HFLOANGL
-----
| IEC KAIMAL TURBULENCE MODEL SPECIFIED [IECKAI]
A | IEC turbulence characteristic [SPECIFIED]
64.700 | Height of reference wind speed, UREF (m) [SPECIFIED]
12.500 | Mean wind speed at reference height, ZREF (m/s) [SPECIFIED]
0.000 | Gradient Richardson number, RI_NO [IEC 61400-1]
| M-O z/L parameter [ IEC N/A ]
| M-O length scale, L (M) [ IEC N/A ]
| Non-dimensional wind shear [ IEC N/A ]
| Surface roughness length, Z0 (m) [ IEC N/A ]
0.200 | Power law exponent, PLEXP [IEC 61400-1]
| Friction or shear velocity, USTAR (m/s) [ IEC N/A ]
| Mixing layer depth, ZI (m) [ IEC N/A ]
| U'W' cross-correlation coefficient, PC_UW [ IEC N/A ]
| U'V' cross-correlation coefficient, PC_UV [ IEC N/A ]
| V'W' cross-correlation coefficient, PC_VW [ IEC N/A ]
8.800 | U component coherence decrement, COHD_U [IEC 61400-1]
8.800 | V component coherence decrement, COHD_V [IEC 61400-1]
8.800 | W component coherence decrement, COHD_W [IEC 61400-1]
0.000 | Coherence exponent, COHEXP [IEC 61400-1]

```

All input values were acceptable.

----- Summary of output data files -----

You have requested a binary time series file.  
The information file will be called "12ms.ddd".  
The data file will be called "12ms.wnd".

```

*****
* IEC TURBULENCE SIMULATION SCALING PARAMETER SUMMARY *
*****
TURBULENCE MODEL USED           = KAIMAL
TURBULENCE CHARACTERISTIC       = A
HUB MEAN U-COMP VELOCITY        = 12.5 m/s
HUB TURBULENCE INTENSITY [I(15)] = 0.18
STD DEVIATION SLOPE (a)         = 2.0
CHAR VALUE OF STANDARD DEVIATION = 2.40 m/s
TURBULENCE SCALE                 = 21.0 m
COHERENCY SCALE                 = 73.5 m
TARGET TURBULENCE INTENSITY     = 0.192
*****

```

----- Summary of transfer matrix warnings -----

There were 0 warnings during calculation of the transfer matrix H1.

Statistical summaries associated with time series block number 1

There were 0 warnings during calculation of the transfer matrix H2.

Statistical summaries associated with time series block number 1

There were 0 warnings during calculation of the transfer matrix H3.

Transfer matrix H was calculated successfully.

----- Summary of specified parameters -----

The mean wind profile is used in the time series calculations.

Mean Wind Speed Profile

Height (m)	U-Component (m/s)
106.7	13.82
89.9	13.35
73.1	12.81
64.7	12.50
56.3	12.16
39.5	11.33
22.7	10.14

Statistical summaries associated with time series block number 1

Statistical summaries associated with time series block number 1

----- CROSS-AXIS CORRELATION COEFFICIENTS USED IN SIMULATION-----

u'w' X-axis correlation coefficient (r^2) = 0.000  
u'v' X-axis correlation coefficient (r^2) = 0.000  
v'w' X-axis correlation coefficient (r^2) = 0.000

----- HUB-HEIGHT SIMULATED TURBULENCE STATISTICAL SUMMARY -----

Longitudinal wind component...

Mean U= 12.50 peak= 19.61 min= 3.69 Sigma= 2.355 m/s; TI= 18.838%

Horizontal wind speed...  
 Mean Uh= 12.65 peak= 20.46 min= 3.73 Sigma= 2.286 m/s; TI= 18.067%

Total wind speed...  
 Mean Ut= 12.71 peak= 20.65 min= 5.71 Sigma= 2.259 m/s; TI= 17.770%

Lateral wind component...  
 Mean V= 0.00 peak= 7.39 min= -6.38 Sigma= 1.887 m/s; TI= 14.845%

Vertical wind component...  
 Mean W= 0.00 peak= 4.81 min= -4.27 Sigma= 1.171 m/s; TI= 9.209%

Mean Flow Angles...

Vertical= 0.0 Horizontal= 0.0 degs

Turbulent velocity component extremes...

u' peak= 7.11 min= -8.81 m/s  
 v' peak= 7.39 min= -6.38 m/s  
 w' peak= 4.81 min= -4.27 m/s

Hub Friction Velocity (Ustar) = 0.296 m/s

Mean Reynolds stress components...

u'w'= 0.296 m/s  
 u'v'= 0.829 m/s  
 v'w'= 0.423 m/s

Instantaneous Reynolds stress component extremes...

u'w' peak= 28.50 min= -41.60 (m/s)\*\*2  
 u'v' peak= 46.69 min= -41.49 (m/s)\*\*2  
 v'w' peak= 25.43 min= -23.43 (m/s)\*\*2

Cross-component correlation coefficients...

u'w' coef= 0.032  
 u'v' coef= 0.155  
 v'w' coef= 0.081

----- GRID POINT VARIANCE SUMMARY -----

Y-coord	-42.00	-25.20	-8.40	8.40	25.20	42.00
Height	Standard deviation at grid points for U component					
106.7	2.357	2.291	2.174	2.305	2.248	2.341
89.9	2.387	2.293	2.176	2.231	2.271	2.343
73.1	2.329	2.115	2.182	2.229	2.329	2.427
56.3	2.286	2.262	2.217	2.338	2.372	2.271
39.5	2.385	2.400	2.348	2.310	2.487	2.374
22.7	2.309	2.239	2.320	2.184	2.236	2.247
Height	Standard deviation at grid points for V component					
106.7	1.893	1.816	1.760	1.861	1.826	1.878
89.9	1.894	1.825	1.758	1.805	1.843	1.866
73.1	1.877	1.739	1.793	1.798	1.873	1.928
56.3	1.869	1.847	1.832	1.862	1.883	1.810
39.5	1.893	1.923	1.889	1.873	1.954	1.933

22.7	1.911	1.909	1.914	1.869	1.894	1.906
Height	Standard deviation at grid points for W component					
106.7	1.180	1.159	1.141	1.171	1.171	1.181
89.9	1.173	1.154	1.134	1.155	1.170	1.177
73.1	1.170	1.139	1.148	1.145	1.169	1.191
56.3	1.173	1.162	1.167	1.159	1.168	1.138
39.5	1.176	1.188	1.174	1.172	1.178	1.197
22.7	1.200	1.215	1.207	1.204	1.203	1.223

\*\*\*\* U-COMPONENT MEAN & TURBULENCE INTENSITY \*\*\*  
 FROM 4 GRID POINTS SURROUNDING HUB (GRID CENTER)

MEAN = 12.483 m/s  
 TI = 18.150 %

---- Normalizing Parameters for Binary Data ----

Ubar = 12.500  
 TI(u) = 18.838%  
 TI(v) = 14.845%  
 TI(w) = 9.209%

The SNLWIND-3D simulation has completed normally.



## APPENDIX B.1: SYMDYN TO ADAMS CONVERSION OF INDEPENDENT BLADE PITCH CONTROL USING ANGLE OF ATTACK SENSORS AND A PERIODIC ESTIMATOR AND CONTROLLER

Here are some definitions to help understand the AOA control system and the various matrices needed to implement it.

There are several variables that make sense in the physical world (e.g., wind speed, angle of attack, blade flap angle) that must go through a change of variables so that they fit the form of a linear dynamical system. The change of variables is accomplished by subtracting the steady state solution from the variable.

The control gains are calculated at 12, 14, 16, 18, and 20 m/s wind speed, and at a large number (200-500) of azimuth points. The estimator maintains its best guess of the current wind speed (and wind shears). The control gains, estimator gains, and steady state operating point for the current operating point are found by using the estimated wind speed and estimated (measured) azimuth position and then interpolating the pre-calculated matrices and operating points at 12, 14, 16, 18, 20, 22, 24, and 26 m/s wind speeds.

### ***Turbine Parameters***

SymDyn represents a flexible wind turbine with a limited number of angles between rigid bodies. We begin with the definition of rotor azimuth.

$$\psi = \text{azimuth of rotor (radians)}$$

$$\psi_{ops} = \text{desired azimuth of rotor (radians)}$$

$$\begin{aligned}\psi_{ops} &= \Omega t \\ &= RPM * 2\pi / 60 \\ &= 20.46 * 2\pi / 60\end{aligned}$$

This sets a target position for the rotor for simulation purposes. In ADAMS simulations, and in real turbine operation, adjustments may be required to allow the control system to start up from any position.

The following blade flap angles are part of the way that SymDyn models the wind turbine. In ADAMS, the blades are flexible and are not represented by these flap angles and rates. The flap angles and rates are intermediate internal variables used in the estimator.

$$\beta_i = \text{flap angle of blade i (radians)}$$

$$\dot{\beta}_i = \text{angular flap rate of blade i (rad/sec)}$$

$$\mathbf{q} = \begin{bmatrix} \psi \\ \beta_1 \\ \beta_2 \\ \beta_3 \end{bmatrix} \quad (4 \times 1)$$

$$\dot{\mathbf{q}} = \begin{bmatrix} \dot{\psi} \\ \dot{\beta}_1 \\ \dot{\beta}_2 \\ \dot{\beta}_3 \end{bmatrix} \quad (4 \times 1)$$

$$\mathbf{q}_{ops} = \text{operating point values of } \mathbf{q} \quad (4 \times 1 \times \psi \times V_\infty)$$

$$\dot{\mathbf{q}}_{ops} = \text{operating point values of } \dot{\mathbf{q}} \quad (4 \times 1 \times \psi \times V_\infty)$$

The notation  $\times \psi$  means the values are tabulated as a function of azimuth. The notation  $\times V_\infty$  means the values are tabulated as a function of hub-height wind speed. Thus,  $\mathbf{q}_{ops}$  is a four-component vector of parameters defining the turbine state that is tabulated over azimuth and wind speed.

### State Variables

The turbine equation of motion written in terms of the turbine parameters is a second order system. To facilitate control law creation, the system is converted to a state-space system of first order equations. The vector of state variables  $\mathbf{x}$  used in SymDyn is

$$\mathbf{x} = \begin{bmatrix} \mathbf{q} \\ \dot{\mathbf{q}} \end{bmatrix} - \begin{bmatrix} \mathbf{q}_{ops} \\ \dot{\mathbf{q}}_{ops} \end{bmatrix} \quad (8 \times 1)$$

Note that the state variables are zero when the turbine is at the steady state operating point. Also note that  $\mathbf{q}_{ops}$  and  $\dot{\mathbf{q}}_{ops}$  are functions of  $\psi$  and  $V_\infty$ . Initially, we implemented the control system in ADAMS without interpolating with respect to  $V_\infty$ .

### Angle of Attack

The AOA at one position along each blade is used as inputs to the estimator. For now, the angle of attack sensor is located at the 11<sup>th</sup> of 15 blade elements.

$$AoA_i = \text{angle of attack of sensor on blade } i \quad (\text{degrees})$$

$$\mathbf{AoA} = \begin{bmatrix} AoA_1 \\ AoA_2 \\ AoA_3 \end{bmatrix} \quad (3 \times 1)$$

$$\mathbf{AoA}_{ops} = \text{operating point values of } \mathbf{AoA} \quad (3 \times 1 \times \psi \times V_\infty)$$

Note, that  $\mathbf{AoA}_{ops}$  is a function of  $\psi$  and  $V_\infty$ .

The deviation  $\boldsymbol{\alpha}$  of angle of attack from operating point angle of attack is

$$\boldsymbol{\alpha} = \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \end{bmatrix} \equiv \mathbf{AoA} - \mathbf{AoA}_{ops} \quad (3 \times 1)$$

Note: AOA sensor outputs are in degrees and not radians.

### **Sensor Outputs**

Next we define the sensor outputs.

$$\mathbf{y} = \begin{bmatrix} \psi \\ \dot{\psi} \\ \alpha_1 \\ \alpha_2 \\ \alpha_3 \end{bmatrix} \quad (5 \times 1)$$

Thus, the system has five sensors: one measuring azimuth, one measuring angular velocity of the rotor, and one measuring the deviation from operating AOA at the 11<sup>th</sup> station on each blade.

### **Blade Pitch Angles**

Next, we define the blade pitch angles and pitch control variables.

$$\boldsymbol{\theta} = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \end{bmatrix} \quad (3 \times 1)$$

$\theta_i$  = pitch angle of blade  $i$  (radians)

$\boldsymbol{\theta}_{ops}$  = operating point values of  $\boldsymbol{\theta}_{ops}$  (3 x 1 x  $V_\infty$ )

The  $\boldsymbol{\theta}_{ops}$  are a function of  $V_\infty$  (the uniform free stream wind speed) but **not** a function of  $\psi$  (azimuth position). We could make them a function of azimuth to reduce steady state dynamics. The  $\boldsymbol{\theta}_{ops}$  are given in the table below in degrees, but radians should be used in the control system. They are the same for each blade for a given  $V_\infty$ . These pitch angles give the proper torque to maintain the desired RPM in uniform wind.

Wind Speed (m/s)	12	14	16	18	20
$\theta_{ops}$ (radians)	6.7528	12.2358	16.0005	19.16	21.8885
Vert. Lin. Wind Shear	0.0	0.0	0.0	0.0	0.0
Horz. Lin. Wind Shear	0.0	0.0	0.0	0.0	0.0

The operating point wind shears are also given in the table. The operating point wind shears do not have to be zero. They could be the expected or average wind shear. If this were the case, the deviations from operating points would be minimized, and the deviations of the plant (wind turbine) from the linear model of the turbine would be minimized. However, when the turbine operates in wind shear without periodic control, there is more periodic motion than when the turbine operates in wind without wind shear. Because the controller is using the pitch angle input to return the turbine to periodic steady state condition of the operating point, and because we want to minimize the loads and motion of the wind turbine, it is better to choose an operating point without wind shear.

### **Blade Pitch Control Variables**

Now for defining the pitch control variables.

$$\mathbf{u} = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} \equiv \boldsymbol{\theta} - \boldsymbol{\theta}_{ops} \quad (3 \times 1)$$

### **Wind Disturbances**

First, we define the wind disturbance parameters. The wind inflow can be parameterized in many ways. For now, we have chosen to use a subset of the parameters used in the AeroDyn routines. The wind disturbance vector is

$$\mathbf{W} = \begin{bmatrix} W_1 \\ W_2 \\ W_3 \end{bmatrix} \equiv \begin{bmatrix} V_\infty \\ LshearH \\ LshearV \end{bmatrix} \quad (3 \times 1)$$

Thus, the first component is the hub-height wind speed in m/s, the second component is the linear horizontal shear, and the third component is the linear vertical shear. Additional components of vertical wind and wind direction can be added.

It should be noted that the actual wind is more complicated than can be represented by these three (or five) parameters.

We also define the wind disturbance operating point as

$$\mathbf{W}_{op} = \begin{bmatrix} W_{1,op} \\ W_{2,op} \\ W_{3,op} \end{bmatrix} \equiv \begin{bmatrix} V_{\infty,op} \\ LshearH_{op} \\ LshearV_{op} \end{bmatrix} \quad (3 \times 1)$$

The wind disturbance operating point defines the wind conditions in which the linear equations of motion were calculated from the nonlinear equations. An example would be 12 m/s, 0, and 0 for each component. As noted above, the wind shear operating points are not necessarily zero.

Next, we define the homogenized wind disturbance parameter  $\mathbf{u}_d$ .

$$\mathbf{u}_d = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} \equiv \mathbf{W} - \mathbf{W}_{op} \quad (3 \times 1)$$

### **State Equations of Motion**

The state variables  $\mathbf{x}$ , the control inputs  $\mathbf{u}$ , and the wind disturbances  $\mathbf{u}_d$  are related by the linear system of first order differential equations:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} + \mathbf{B}_d\mathbf{u}_d \quad (8 \times 1)$$

### **Sensor Output Equations**

The sensor outputs, the state variables, the control inputs, and the wind disturbances are related by:

$$\mathbf{y} = \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u} + \mathbf{D}_d\mathbf{u}_d \quad (5 \times 1)$$

where

$$\mathbf{C} = \begin{bmatrix} \mathbf{C}_x \\ \mathbf{C}_{AoA} \end{bmatrix} \quad (5 \times 8)$$

and

$$\mathbf{C}_x = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \quad (2 \times 8)$$

to produce azimuth and rotor angular velocity sensors, and

$$\mathbf{C}_{AoA} = \begin{bmatrix} c_{11}^{AoA} & c_{12}^{AoA} & c_{13}^{AoA} & c_{14}^{AoA} & c_{15}^{AoA} & c_{16}^{AoA} & c_{17}^{AoA} & c_{18}^{AoA} \\ c_{21}^{AoA} & c_{22}^{AoA} & c_{23}^{AoA} & c_{24}^{AoA} & c_{25}^{AoA} & c_{26}^{AoA} & c_{27}^{AoA} & c_{28}^{AoA} \\ c_{31}^{AoA} & c_{32}^{AoA} & c_{33}^{AoA} & c_{34}^{AoA} & c_{35}^{AoA} & c_{36}^{AoA} & c_{37}^{AoA} & c_{38}^{AoA} \end{bmatrix} \quad (3 \times 8 \times \psi \times V_\infty)$$

and

$$\mathbf{D} = \begin{bmatrix} \mathbf{0} \\ \mathbf{D}_{AoA} \end{bmatrix} \quad (5 \times 3 \times \psi \times V_\infty)$$

and

$$\mathbf{D}_{AoA} = \begin{bmatrix} d_{11} & d_{12} & d_{13} \\ d_{21} & d_{22} & d_{23} \\ d_{31} & d_{32} & d_{33} \end{bmatrix} (3 \times 3 \times \psi \times V_\infty)$$

The  $\mathbf{D}_{AoA}$  is highly diagonal as it relates the change in AOA to the change-in-pitch angle.

$$\mathbf{D}_d = \begin{bmatrix} \mathbf{0} \\ \mathbf{D}_{d,AoA} \end{bmatrix} (5 \times 3 \times \psi \times V_\infty)$$

$$\mathbf{D}_{d,AoA} = \begin{bmatrix} Dd_{11} & Dd_{12} & Dd_{13} \\ Dd_{21} & Dd_{22} & Dd_{23} \\ Dd_{31} & Dd_{32} & Dd_{33} \end{bmatrix} (3 \times 3 \times \psi \times V_\infty)$$

### **Augmented State Vector**

The estimator works with an augmented vector of state variable in which the disturbance states are added to the turbine states to form one vector of state variables  $\bar{\mathbf{x}}$ .

$$\bar{\mathbf{x}} = \begin{bmatrix} \mathbf{x} \\ \mathbf{u}_d \end{bmatrix} (11 \times 1)$$

### **Augmented Dynamic Equation**

Applying the definition of  $\bar{\mathbf{x}}$ , the equation of motion becomes

$$\dot{\bar{\mathbf{x}}} = \bar{\mathbf{A}}\bar{\mathbf{x}} + \bar{\mathbf{B}}\mathbf{u}$$

where

$$\bar{\mathbf{A}} = \begin{bmatrix} \mathbf{A} & \mathbf{B}_d \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \quad \bar{\mathbf{B}} = \begin{bmatrix} \mathbf{B} \\ \mathbf{0} \end{bmatrix}$$

and the sensor output equation becomes

$$\mathbf{y} = \bar{\mathbf{C}}\bar{\mathbf{x}} + \bar{\mathbf{D}}\mathbf{u} (5 \times 1)$$

where

$$\bar{\mathbf{C}} = [\mathbf{C} \quad \mathbf{D}_d] (5 \times 11 \times \psi \times V_\infty)$$

and

$$\bar{\mathbf{D}} = \mathbf{D}$$

(Note:  $\mathbf{D}_d$  is denoted  $\mathbf{C}_d$  in the SymDyn code.) The estimated augmented state vector is denoted by  $\hat{\mathbf{x}}$ , the estimated state vector is denoted by  $\hat{\mathbf{x}}$ , and the estimated wind disturbance is denoted by  $\hat{\mathbf{u}}_d$ , and they are related to each other by

$$\hat{\mathbf{x}} = \begin{bmatrix} \hat{\mathbf{x}} \\ \hat{\mathbf{u}}_d \end{bmatrix} \quad (11 \times 1)$$

The estimated augmented state vector is calculated by integrating

$$\begin{aligned} \dot{\hat{\mathbf{x}}} &= \overline{\mathbf{A}}\hat{\mathbf{x}} + \overline{\mathbf{B}}\mathbf{u} + \mathbf{K}(\mathbf{y} - \overline{\mathbf{C}}\hat{\mathbf{x}} - \overline{\mathbf{D}}\mathbf{u}) \\ &= (\overline{\mathbf{A}} - \mathbf{K}\overline{\mathbf{C}})\hat{\mathbf{x}} + (\overline{\mathbf{B}} - \mathbf{K}\overline{\mathbf{D}})\mathbf{u} + \mathbf{K}\mathbf{y} \\ &= \tilde{\mathbf{A}}\hat{\mathbf{x}} + \tilde{\mathbf{B}}\mathbf{u} + \mathbf{K}\mathbf{y} \end{aligned}$$

where the matrix of estimator gains  $\mathbf{K}$  is calculated in SymDyn, and the terms in the brackets can be pre-calculated to reduce simulation time.

### **Control Law**

Now for the linear control law, which is a linear function of the estimated state variables and the estimated wind disturbances:

$$\begin{aligned} \mathbf{u} &= \mathbf{G}\hat{\mathbf{x}} + \mathbf{G}_d\hat{\mathbf{u}}_d \\ &= [\mathbf{G} \quad \mathbf{G}_d]\hat{\mathbf{x}} \\ &= \overline{\mathbf{G}}\hat{\mathbf{x}} \end{aligned}$$

The matrices  $\mathbf{G}$  and  $\mathbf{G}_d$  are calculated in SymDyn.

The pitch command is given by:

$$\boldsymbol{\theta}_{cmd} = \mathbf{u} + \boldsymbol{\theta}_{ops} \quad (3 \times 1)$$

The actual pitch angle  $\boldsymbol{\theta}$  is slightly different due to actuator dynamics. It may also be different due to absolute limits and rate-of-change limits applied to the pitch command signal. The actual pitch angle should be used in the augmented state estimator calculations and not necessarily the commanded pitch angle from the linear control law.

### **Summary**

The data needed for implementation of the controller in ADAMS are:

$$\boldsymbol{\theta}_{ops} \quad (3 \times 1 \times V_\infty)$$

$$\mathbf{W}_{op} \quad (3 \times 1 \times V_\infty)$$

$$\mathbf{A} \mathbf{o} \mathbf{A}_{ops} \quad (3 \times 1 \times \psi \times V_\infty)$$

$$\mathbf{q}_{ops} = \text{operating point values of } \mathbf{q} \quad (4 \times 1 \times \psi \times V_\infty)$$

$$\dot{\mathbf{q}}_{ops} = \text{operating point values of } \dot{\mathbf{q}} \quad (4 \times 1 \times \psi \times V_\infty)$$

(There is some question as to whether the operating point data should come from the ADAMS results or the SymDyn simulations. Hopefully, there is not a big difference between the simulations results so it won't matter.)

Augmented state matrices

$$\bar{\mathbf{A}} \quad \text{from} \quad \mathbf{A} \quad \mathbf{B}_d$$

$$\bar{\mathbf{B}} \quad \text{from} \quad \mathbf{B}$$

$$\bar{\mathbf{C}} \quad \text{from} \quad \mathbf{C} \quad \mathbf{D}_d \quad \text{from} \quad \mathbf{C}_{AoA} \quad \mathbf{D}_{d,AoA}$$

$$\bar{\mathbf{D}} \quad \text{from} \quad \mathbf{D} \quad \text{from} \quad \mathbf{D}_{AoA}$$

Estimator and Control Gains

$$\mathbf{K} \quad \text{from} \quad \text{SymDyn}$$

$$\bar{\mathbf{G}} \quad \text{from} \quad \mathbf{G} \quad \mathbf{G}_d$$

Thus, matrices supplied to ADAMS are  $\tilde{\mathbf{A}}$ ,  $\tilde{\mathbf{B}}$ ,  $\mathbf{K}$ , and  $\bar{\mathbf{G}}$ , and the operating points.



## APPENDIX B.2: FORTRAN 90 SOURCE CODE FOR PITCH CONTROL ALGORITHM (ADAMS SUBROUTINE GSESUB)

```

! GSESub subroutine and supporting routines for LWT project
!   Windward Engineering 3/26/03
!
! -----
MODULE GSEMatrices

INTEGER::      NAzim   = 201  ! Number of azimuth steps in control matrices
INTEGER::      NAlphas = 3    ! Also used for number of outputs (pitch angles)
INTEGER::      Nwyes   = 5    ! Number of inputs
INTEGER::      NSpeed  = 6    ! Number of different wind speeds represented by matrices
INTEGER::      NMats   = 6    ! Number of matrices for each wind speed

INTEGER        index                !location in Azimuth tables

DOUBLE PRECISION, ALLOCATABLE :: AMat (:,:,,:)
DOUBLE PRECISION, ALLOCATABLE :: BMat (:,:,,:)
DOUBLE PRECISION, ALLOCATABLE :: GMat (:,:,,:)
DOUBLE PRECISION, ALLOCATABLE :: KMat (:,:,,:)
DOUBLE PRECISION, ALLOCATABLE :: AoAMat (:,:,,:)
DOUBLE PRECISION, ALLOCATABLE :: ThetaOps (:,:) !Target pitch angle

DOUBLE PRECISION, ALLOCATABLE :: AMatNow (:,:)
DOUBLE PRECISION, ALLOCATABLE :: BMatNow (:,:)
DOUBLE PRECISION, ALLOCATABLE :: GMatNow (:,:)
DOUBLE PRECISION, ALLOCATABLE :: KMatNow (:,:)
DOUBLE PRECISION, ALLOCATABLE :: AoAMatNow (:,:)
DOUBLE PRECISION                ThetaNow (3) !Target pitch angle

DOUBLE PRECISION      OmegaOps !Desired Omega, rad/s

DOUBLE PRECISION      WOps(3)  ! Target wind speed, h shear, v shear

REAL::                Speeds(6) = (/ 12.0, 14.0, 16.0, 18.0, 20.0, 22.0 /)! Array of wind speeds

END MODULE GSEMatrices

! -----
MODULE PitchValues

! Pitch Actuator limits
REAL::      MaxPitAng = 1.570796  !Maximum pitch angle (90 deg) - rad
REAL::      MinPitAng = 0.04537854 !Minimum pitch angle (2.6 deg) - rad
REAL::      MaxPitRat = 0.1745329  !Maximum pitch rate ( 10 deg/sec) - rad/sec
REAL::      MinPitRat = -0.1745329 !Minimum pitch rate (-10 deg/sec) - rad/sec

END MODULE PitchValues

! -----

SUBROUTINE GSESUB (ID, TIME, PAR, NPAR, DFLAG, IFLAG, &
  NSTATE, STATES, NINPUT, INPUT, NOUTPT, STATED, OUTPUT)

USE GSEMatrices
USE PitchValues
USE ElOutParams ! for angle of attack (ALFALL(iblade,jelem))
USE Constant    ! for Pi and any units conversions

IMPLICIT NONE
!
! === External variables ===
!
INTEGER      I
INTEGER      ID
INTEGER      NPAR
INTEGER      NINPUT
INTEGER      NSTATE
INTEGER      NARRAY

```

```

INTEGER          NOUTPT
DOUBLE PRECISION TIME
DOUBLE PRECISION PAR(NPAR)
DOUBLE PRECISION STATES(NSTATE)
DOUBLE PRECISION STATED(NSTATE)
DOUBLE PRECISION INPUT(NINPUT)
DOUBLE PRECISION OUTPUT(NOUTPT)
DOUBLE PRECISION UpDateTime
DOUBLE PRECISION OutTime
DOUBLE PRECISION OldTime
DOUBLE PRECISION PitDemand(NOUTPT)
LOGICAL          DFLAG
LOGICAL          IFLAG
!
! === External variable descriptions ===
!
! ID Identifier of the calling GSE
! TIME Current simulation time
! PAR Array of passed parameters from GSE statement
! NPAR Number of parameters in PAR
! DFLAG Not used
! IFLAG Initialization flag
! NSTATE Number of state variables in the GSE
! STATES Array of current state vector values
! NINPUT Number of inputs in the GSE
! INPUT Array of current input vector values
! NOUTPT Number of outputs in the GSE
! STATED Array of computed state derivative vector values
! OUTPUT Array of computed output vector values

DOUBLE PRECISION  CntrlStartTime      !Time at which this controller begins to track
DOUBLE PRECISION  CntrlTime           !Time at which this controller begins to control
DOUBLE PRECISION  Y(5)                !Y array in Eric's notation ( psi, psidot, dalpha1,
dalpha2, dalpha3)
DOUBLE PRECISION  U(3)                !U array in Eric's notation (Theta - ThetaNow)

DOUBLE PRECISION  Azimuth
DOUBLE PRECISION  AzimZero
DOUBLE PRECISION  AzimShift           !"error" in azimuth.  adams value shifted by this
amount (rad)
DOUBLE PRECISION  G_Gain

REAL              DelVave              !Average wind speed deviation from WOps(1) based on
pitch demand
REAL              Delt                !Used in step function on DelVave
REAL              HHWS                !Hub-height wind speed
REAL              HHWD                !Hub-height wind direction

REAL::            TimeAvePit = 0.0

INTEGER           AoAElem              !Element where we'll get angle of attack

LOGICAL::         SFLAG = .TRUE.      !Initialization flag
LOGICAL           ERRFLG

SAVE

! Return on first entry to get other variables initialized
IF (IFLAG) RETURN

! Initialization
IF (SFLAG) THEN

! Get W_Ops
CALL SYSARY ( 'ARRAY', 2001, 1, WOps, NARRAY, ERRFLG )

! Get OmegaOps
CALL SYSARY ( 'ARRAY', 2002, 1, OmegaOps, NARRAY, ERRFLG )

OldTime = TIME
PitDemand = Input(3:5)

```

```

AoAElem      = NINT( PAR(1) )
CntrlStartTime = PAR(2)
UpdateTime    = PAR(3)
OutTime      = PAR(4)
AzimShift    = PAR(5)
G_Gain       = PAR(6)

! Get the start time of the GSE control
CALL SYSFNC ( 'VARVAL', 1010, 1, CntrlTime, ERRFLG )

IF (CntrlTime .LT. CntrlStartTime) THEN
  WRITE(*,*) 'CntrlStartTime is less than CntrlTime in GSESub'
  WRITE(*,*) ' '
  CALL ERRMES (.TRUE., 'Aborting in GSESub', 101, 'STOP')
ENDIF

WRITE(*,*) 'Initializing GSE Subroutine'
WRITE(*,*) 'NSTATE = ', NSTATE
WRITE(*,*) 'NINPUT = ', NINPUT
WRITE(*,*) 'NOUTPT = ', NOUTPT
WRITE(*,*) 'W_Ops = ', SNGL(WOps(1))
WRITE(*,*) 'Omega_Ops = ', SNGL(OmegaOps)
WRITE(*,*) 'Using angle of attack from element ', AoAElem
WRITE(*,*) 'AoA controller integrator will begin at t = ', SNGL(CntrlStartTime)
WRITE(*,*) 'AoA pitch control will be fully active at t = ', SNGL(CntrlTime)
WRITE(*,*) 'Overall gain applied to G matrix = ', SNGL(G_Gain)
WRITE(*,*) '*****'

CALL ReadMatrices(NSTATE)
!CALL WriteMatrices(NSTATE) !For debug output only, to check values of
matrices

!Adjust GMat array by applying overall gain
GMat = G_Gain * GMat

SFLAG = .FALSE.

ENDIF !Initialization block

Y(1) = INPUT(1) !Azimuth angle, rad (changed to deviation later)
Y(2) = INPUT(2) - OmegaOps !Ang vel deviation, rad/s

! Select matrices for the current azimuth value (azimuth must be between 0, 2Pi for table
lookups)
Azimuth = MOD( SNGL(Y(1)+AzimShift), TwoPi ) ! based on "reality" not estimate
index = NINT( (NAzim-1) * Azimuth/TwoPi ) + 1
index = MIN(NAzim, index) !avoid overflowing arrays near end points

! Get the current matrices based on DelVave from last time step
CALL GetMatNow (NSTATE, DelVave)

U(1) = INPUT(3) - ThetaNow(1) !Input uses actual pitch instead of pitch demand
U(2) = INPUT(4) - ThetaNow(2)
U(3) = INPUT(5) - ThetaNow(3)

Y(3) = ALFALL( 1, AoAElem ) - AoAMatNow(1,1) !AoA deviation, blade 1, deg
Y(4) = ALFALL( 2, AoAElem ) - AoAMatNow(2,1) !AoA deviation, blade 2, deg
Y(5) = ALFALL( 3, AoAElem ) - AoAMatNow(3,1)

Y(3) = SIGN(MIN(5.0,ABS(Y(3))), Y(3))
Y(4) = SIGN(MIN(5.0,ABS(Y(4))), Y(4))
Y(5) = SIGN(MIN(5.0,ABS(Y(5))), Y(5))

! Calculate the raw pitch demand
PitDemand = MATMUL(GMatNow, STATES) + ThetaNow ! the pitch demand for each blade, rad

! Set values of state derivatives and output
IF( TIME < CntrlStartTime ) THEN
  STATED = 0.0D0 ! All derivatives and ICs, hence all states, are zero until
CntrlStartTime

```

```

AzimZero = Y(1)          ! Will be initial azimuth when AoA control begins
Y(1)      = 0.0D0        ! Azimuth error for debug output remains zero until AoA control
starts
ELSEIF( TIME < CntrlTime )THEN
  Y(1)    = Y(1) - OmegaOps * ( Time - CntrlStartTime ) - AzimZero          !Actual azimuth
deviation based upon desired speed
  Y(1) = 0.0          !for debug only
  STATED = MATMUL(AMatNow, STATES) + MATMUL(BMatNow, U) + MATMUL(KMatNow, Y)
  STATED(1) = 0.0D0      ! Hold estimated azimuth error zero until GSE takes over control
ELSE
  Y(1)    = Y(1) - OmegaOps * ( Time - CntrlStartTime ) - AzimZero          !Actual azimuth
deviation based upon desired speed
  Y(1) = 0.0          !for debug only
  STATED = MATMUL(AMatNow, STATES) + MATMUL(BMatNow, U) + MATMUL(KMatNow, Y)
ENDIF

! Get the average wind speed for the next time
CALL GetVave (TIME, UpdateTime, PitDemand, DelVave, TimeAvePit)

IF( TIME < CntrlStartTime ) THEN
  DelVave = 0.0
ELSEIF( TIME < CntrlTime )THEN
  Delt = SNGL((TIME - CntrlStartTime)/(CntrlTime - CntrlStartTime))
  DelVave = DelVave * Delt * Delt * (3 - 2 * Delt)
ENDIF

!Get the actual pitch demand based on pitch angle and rate limits
CALL GetPitDemand (TIME, UpdateTime, Input(3:5), PitDemand, OUTPUT)

! See if we need to write output
IF (OutTime < 0.00001) RETURN ! Skip the output

! See if it is time to write output
IF (Time - OldTime >= OutTime) THEN

! Update the azimuth to the last time pitch demand was updated

  CALL GSEDebug (PitDemand, OUTPUT, NOUTPT, Y, TIME, STATES, NSTATE, DelVave, TimeAvePit)
  OldTime = TIME

ENDIF

RETURN
END SUBROUTINE GSESUB

! -----
SUBROUTINE GetVave (TIME, UpdateTime, PitchDemand, DelVave, TimeAvePit)
! Determines the average wind speed based on pitch demand
! DJL - 06/26/03

USE          GSEMatrices

IMPLICIT    NONE

INTEGER     iSpeed

DOUBLE PRECISION TIME          ! Current simulation time
DOUBLE PRECISION UpDateTime    ! Time step for updating DelVave
DOUBLE PRECISION OldTime       ! Time at which DelVave was last updated
DOUBLE PRECISION DeltaT        ! Current TIME minus OldTime
DOUBLE PRECISION PitchDemand(3) ! Array of pitch demand

REAL        BladeAvePit        ! Current pitch demand averaged over 3 blades
REAL        TimeAvePit         ! Pitch demand averaged over a time window
REAL::      F = 0.0001         ! Weighting coefficient
REAL        Vave               ! Average wind speed
REAL        DelVave            ! Average wind speed deviation from WOps(1)
REAL        P                  ! interpolation factor

```

```

LOGICAL::          SFLAG = .TRUE.

SAVE

IF (SFLAG) THEN

    OldTime      = TIME
    DelVave      = 0.0
    SFLAG        = .FALSE.

ENDIF

DeltaT = Time - OldTime

! See if it is time to update
IF (DeltaT < UpdateTime) RETURN

! Calculate the blade-average pitch demand
BladeAvePit = SNGL(SUM(PitchDemand)/SIZE(PitchDemand))

! Calculate the time-average pitch demand
TimeAvePit = (1-F)*TimeAvePit + F*BladeAvePit

! Search for the current pitch angle and velocity (use blade one values since they are the
same)
IF (TimeAvePit <= ThetaOps(1,1)) THEN !We are below the lowest pitch angle
    Vave = Speeds(1)
ELSEIF (TimeAvePit >= ThetaOps(NSpeed,1)) THEN !We are above the highest pitch angle
    Vave = Speeds(NSpeed)
ELSE
    ! We need to interpolate between pitch angles
    DO iSpeed = 1, NSpeed - 1
        IF (TimeAvePit >= ThetaOps(iSpeed,1) .AND. TimeAvePit < ThetaOps(iSpeed + 1,1)) THEN
            P = ( TimeAvePit - ThetaOps(iSpeed,1) ) / ( ThetaOps(iSpeed + 1,1) -
ThetaOps(iSpeed,1) )
            Vave = Speeds (iSpeed) + P * (Speeds (iSpeed + 1) - Speeds (iSpeed))
        ENDIF
    END DO ! iSpeed
ENDIF

DelVave = Vave - WOps(1)

RETURN
END SUBROUTINE GetVave

! -----
SUBROUTINE GetMatNow (NSTATE, DeltaWindSpeed)
! Gets the AMatNow, BMatNow, ... matrices
! DJL - 06/23/03

USE          GSEMatrices
USE          PitchValues

IMPLICIT    NONE

INTEGER     I
INTEGER     iSpeed
INTEGER     NSTATE

REAL        DeltaWindSpeed      ! Wind speed disturbance from the operating point
REAL        WindSpeed
REAL        P                    ! interpolation factor

DOUBLE PRECISION AMatLo ( NSTATE, NSTATE )
DOUBLE PRECISION BMatLo ( NSTATE, NAlphas )
DOUBLE PRECISION KMatLo ( NSTATE, Nwyes )
DOUBLE PRECISION GMatLo ( NAlphas, NSTATE )
DOUBLE PRECISION AoAMatLo( 3,      1      )

```

```

DOUBLE PRECISION ThetaLo ( 3      )
DOUBLE PRECISION AMathi ( NSTATE, NSTATE )
DOUBLE PRECISION BMathi ( NSTATE, NAlphas )
DOUBLE PRECISION KMathi ( NSTATE, Nwyes )
DOUBLE PRECISION GMathi ( NAlphas, NSTATE )
DOUBLE PRECISION AoAMatHi( 3,      1      )
DOUBLE PRECISION ThetaHi ( 3      )

! The current operating point wind speed
WindSpeed = Wops(1) + DeltaWindSpeed

IF (WindSpeed <= Speeds(1)) THEN !We are below the lowest wind speed
  AMatNow = AMat (1, :, :, index)
  BMatNow = BMat (1, :, :, index)
  KMatNow = KMat (1, :, :, index)
  GMatNow = GMat (1, :, :, index)
  AoAMatNow = AoAMat(1, index, :, :)
  ThetaNow = ThetaOps(1, :)
ELSEIF (WindSpeed >= Speeds(NSpeed)) THEN !We are above the highest wind speed
  AMatNow = AMat (NSpeed, :, :, index)
  BMatNow = BMat (NSpeed, :, :, index)
  KMatNow = KMat (NSpeed, :, :, index)
  GMatNow = GMat (NSpeed, :, :, index)
  AoAMatNow = AoAMat(NSpeed, index, :, :)
  ThetaNow = ThetaOps(NSpeed, :)
ELSE
  ! We need to interpolate between wind speeds
  DO iSpeed = 1, NSpeed - 1
    AMatLo ( : , : ) = AMat (iSpeed, :, :, index)
    AMatHi ( : , : ) = AMat (iSpeed+1, :, :, index)
    BMatLo ( : , : ) = BMat (iSpeed, :, :, index)
    BMatHi ( : , : ) = BMat (iSpeed+1, :, :, index)
    KMatLo ( : , : ) = KMat (iSpeed, :, :, index)
    KMatHi ( : , : ) = KMat (iSpeed+1, :, :, index)
    GMatLo ( : , : ) = GMat (iSpeed, :, :, index)
    GMatHi ( : , : ) = GMat (iSpeed+1, :, :, index)
    AoAMatLo( : , : ) = AoAMat(iSpeed, index, :, :)
    AoAMatHi( : , : ) = AoAMat(iSpeed+1, index, :, :)
    ThetaLo ( : ) = ThetaOps(iSpeed, :)
    ThetaHi ( : ) = ThetaOps(iSpeed+1, :)

    IF (WindSpeed >= Speeds(iSpeed) .AND. WindSpeed < Speeds(iSpeed + 1)) THEN
      P = ( WindSpeed - Speeds(iSpeed) ) / ( Speeds(iSpeed + 1) - Speeds(iSpeed) )
      AMatNow = AMatLo + P * (AMatHi - AMatLo )
      BMatNow = BMatLo + P * (BMatHi - BMatLo )
      KMatNow = KMatLo + P * (KMatHi - KMatLo )
      GMatNow = GMatLo + P * (GMatHi - GMatLo )
      AoAMatNow = AoAMatLo + P * (AoAMatHi - AoAMatLo)
      ThetaNow = ThetaLo + P * (ThetaHi - ThetaLo )
    ENDIF
  END DO ! iSpeed
ENDIF

RETURN
END SUBROUTINE GetMatNow

! -----

SUBROUTINE GetPitDemand (TIME, UpdateTime, PitNow, PitDemIn, PitDemOut)
! Limits the pitch demand based on maximum and minimum pitch and
! maximum pitch rate
! DJL - 04/01/03

!USE          Constant
USE          PitchValues

IMPLICIT     NONE

INTEGER      I

```

```

DOUBLE PRECISION TIME          ! Current simulation time
DOUBLE PRECISION UpDateTime    ! Time step for updating pitch demand
DOUBLE PRECISION OldTime       ! Time at which pitch demand was last updated

DOUBLE PRECISION OldPitDem(3) ! Pitch demand at OldTime
DOUBLE PRECISION PitNow(3)    ! Current pitch angle
DOUBLE PRECISION PitDemIn(3)  ! Raw pitch demand
DOUBLE PRECISION PitDemOut(3) ! Pitch demand limited by constraints in this routine

DOUBLE PRECISION DeltaT       ! Current TIME minus OldTime

REAL          PitRate         ! Rate of pitch demand change

LOGICAL::     SFLAG = .TRUE.

SAVE

IF (SFLAG) THEN

    OldTime    = TIME
    OldPitDem  = PitNow
    PitDemOut  = PitNow
    SFLAG      = .FALSE.

ENDIF

DeltaT = Time - OldTime

! See if it is time to update the pitch demand
IF (DeltaT >= UpdateTime) THEN

    ! Limit the pitch angle
    DO I = 1, 3

        ! Limit the pitch rate
        PitRate = SNGL((PitDemIn(I)-OldPitDem(I))/DeltaT)
        IF (PitRate > MaxPitRat) THEN
            PitDemOut(I) = DBLE(MaxPitRat*DeltaT+OldPitDem(I))
        ELSEIF (PitRate < MinPitRat) THEN
            PitDemOut(I) = DBLE(MinPitRat*DeltaT+OldPitDem(I))
        ELSE
            PitDemOut(I) = PitDemIn(I)
        ENDIF

        ! Make sure the pitch demand is within limits
        PitDemOut(I) = MAX( MIN(DBLE(MaxPitAng),PitDemOut(I)), DBLE(MinPitAng) )

    ENDDO ! I

    ! Update the pitch demand values
    OldTime = TIME
    OldPitDem = PitDemOut

ENDIF

RETURN
END SUBROUTINE GetPitchDemand

! -----
SUBROUTINE GSEDebug (RawDem, OUTPUT, NOUTPT, Y, TIME, STATES, NSTATE, DelVave, TimeAvePit)
! Writes output to an tab delimited output file for debugging
! DJL - 04/01/03

USE          GSEMatrices
USE          Constant

IMPLICIT    NONE

INTEGER     I

```

```

INTEGER::          OutUnit = 47

INTEGER           NOUTPT
INTEGER           NSTATE
DOUBLE PRECISION STATES(NSTATE)
DOUBLE PRECISION OUTPUT(NOUTPT)

DOUBLE PRECISION TIME
DOUBLE PRECISION Y(5)
DOUBLE PRECISION RawDem(NOUTPT)

REAL             DelVave

REAL             TimeAvePit

LOGICAL::        FirstPass = .TRUE.

CHARACTER*50     Frmt

SAVE

IF (FirstPass) THEN

  WRITE(*,*) 'Creating the GSE debug output file'

  CALL OpenOutFile(OutUnit, 'GSECntrl.plt')

  Frmt = '( A4, 24(: A1, A ) )'

  WRITE(OutUnit, Frmt) 'Time',           &
    TAB, 'RawDem1',           &
    TAB, 'RawDem2',           &
    TAB, 'RawDem3',           &
    TAB, 'PitDem1',           &
    TAB, 'PitDem2',           &
    TAB, 'PitDem3',           &
    TAB, 'DelAz',             &
    TAB, 'DelFlap1',          &
    TAB, 'DelFlap2',          &
    TAB, 'DelFlap3',          &
    TAB, 'DelRPM',            &
    TAB, 'DelFlRate1',        &
    TAB, 'DelFlRate2',        &
    TAB, 'DelFlRate3',        &
    TAB, 'DelVwind',          &
    TAB, 'DelHShr',           &
    TAB, 'DelVShr',           &
    TAB, 'AzErr',             &
    TAB, 'RPMErr',            &
    TAB, 'DelAoA1',           &
    TAB, 'DelAoA2',           &
    TAB, 'DelAoA3',           &
    TAB, 'DelVave',           &
    TAB, 'TimeAvePit'

  Frmt = '( A5, 24(: A1, A ) )'

  WRITE(OutUnit, Frmt) '(sec)',         &
    TAB, '(deg)',                       &
    TAB, '(deg)',                       &
    TAB, '(deg)',                       &
    TAB, '(deg)',                       &
    TAB, '(deg)',                       &
    TAB, '(deg)',                       &
    TAB, '(deg)',                       &
    TAB, '(deg)',                       &
    TAB, '(deg)',                       &
    TAB, '(deg)',                       &
    TAB, '(deg)',                       &
    TAB, '(deg)',                       &
    TAB, '(rpm)',                       &
    TAB, '(deg/s)',                     &
    TAB, '(deg/s)',                     &

```



```

                TAB,      '(deg/s)',      &
                TAB,      '(m/s)',        &
                TAB,      '(-)',          &
                TAB,      '(-)',          &
                TAB,      '(deg)',         &
                TAB,      '(rpm)',         &
                TAB,      '(deg)',         &
                TAB,      '(deg)',         &
                TAB,      '(deg)',         &
                TAB,      '(m/s)',        &
                TAB,      '(deg)',         &

    FirstPass = .FALSE.

ENDIF !FirstPass

Frmt = '( F10.3, 24( : A1, ES12.5 ) )'

WRITE(OutUnit, Frmt) TIME,
    (TAB,      SNGL(RawDem(I)*RtoD), I = 1, NOUTPT), &
    (TAB,      SNGL(OUTPUT(I)*RtoD), I = 1, NOUTPT), &
    (TAB,      SNGL(STATES(I)*RtoD), I = 1, 4), &
    TAB,      SNGL(STATES(5)*30./pi), &
    (TAB,      SNGL(STATES(I)*RtoD), I = 6, NSTATE - 3), &
    (TAB,      SNGL(STATES(I)), I = NSTATE - 2, NSTATE), &
    TAB,      SNGL(Y(1))*RtoD, &
    TAB,      SNGL(Y(2))*30./Pi, &
    TAB,      SNGL(Y(3)), &
    TAB,      SNGL(Y(4)), &
    TAB,      SNGL(Y(5)), &
    TAB,      DelVave, &
    TAB,      TimeAveFit*RtoD

RETURN
END SUBROUTINE GSEDebug

! -----

SUBROUTINE ReadMatrices(NSTATE)
! Reads ascii text files containing the matrices from E Lang

USE GSEMatrices

IMPLICIT NONE

INTEGER      NSTATE
INTEGER      Sttus
INTEGER      iSpeed
INTEGER      iFile

CHARACTER(8), ALLOCATABLE :: MatFiles(:) ! Names of the matrix files
CHARACTER(72):: Path = 'Matrices/Elong/' !Path to the matrix files
CHARACTER(80)  FileName ! Path and name of the matrix file to read
CHARACTER(11)  Int2LStr

!Allocate matrix arrays
ALLOCATE ( AMat( NSpeed, NSTATE, NSTATE, NAzim ) , STAT=Sttus )
IF ( Sttus /= 0 ) CALL Abort ( ' Error allocating memory for AMat array.' )

ALLOCATE ( BMat( NSpeed, NSTATE, NAlphas, NAzim ) , STAT=Sttus )
IF ( Sttus /= 0 ) CALL Abort ( ' Error allocating memory for BMat array.' )

ALLOCATE ( KMat( NSpeed, NSTATE, Nwyes, NAzim ) , STAT=Sttus )
IF ( Sttus /= 0 ) CALL Abort ( ' Error allocating memory for KMat array.' )

ALLOCATE ( GMat( NSpeed, NAlphas, NSTATE, NAzim ) , STAT=Sttus )
IF ( Sttus /= 0 ) CALL Abort ( ' Error allocating memory for GMat array.' )

ALLOCATE ( AoAMat( NSpeed, NAzim, 3, 1 ) , STAT=Sttus )
IF ( Sttus /= 0 ) CALL Abort ( ' Error allocating memory for AoAMat array.' )

```

```

ALLOCATE ( ThetaOps( NSpeed, 3 ) , STAT=Sttus )
IF ( Sttus /= 0 ) CALL Abort ( ' Error allocating memory for AoAMat array.' )

!Allocate matrix arrays
ALLOCATE ( AMatNow( NSTATE, NSTATE ) , STAT=Sttus )
IF ( Sttus /= 0 ) CALL Abort ( ' Error allocating memory for AMatNow array.' )

ALLOCATE ( BMatNow( NSTATE, NAlphas ) , STAT=Sttus )
IF ( Sttus /= 0 ) CALL Abort ( ' Error allocating memory for BMatNow array.' )

ALLOCATE ( KMatNow( NSTATE, Nwyes ) , STAT=Sttus )
IF ( Sttus /= 0 ) CALL Abort ( ' Error allocating memory for KMatNow array.' )

ALLOCATE ( GMatNow( NAlphas, NSTATE ) , STAT=Sttus )
IF ( Sttus /= 0 ) CALL Abort ( ' Error allocating memory for GMatNow array.' )

ALLOCATE ( AoAMatNow( 3, 1 ) , STAT=Sttus )
IF ( Sttus /= 0 ) CALL Abort ( ' Error allocating memory for AoAMatNow array.' )

ALLOCATE ( MatFiles( NMats ) , STAT=Sttus )
IF ( Sttus /= 0 ) CALL Abort ( ' Error allocating memory for MatFiles array.' )

! Set the names of the matrix files
MatFiles(1) = 'Atilde'
MatFiles(2) = 'Btilde'
MatFiles(3) = 'K'
MatFiles(4) = 'Gbar'
MatFiles(5) = 'AoA_ops'
MatFiles(6) = 'Theta_op'

! Loop through the wind speeds
DO iSpeed = 1, NSpeed

  DO iFile = 1, NMats
    FileName = TRIM(Path)//TRIM(Int2LStr(NINT(Speeds(iSpeed))))//"/"//TRIM(MatFiles(iFile))

! Read the matrix files
    SELECT CASE (iFile)
      CASE (1)
        CALL ReadMatrixFile( FileName, AMat, iSpeed, NSpeed, NSTATE, NSTATE, NAzim )
      CASE (2)
        CALL ReadMatrixFile( FileName, BMat, iSpeed, NSpeed, NSTATE, NAlphas, NAzim )
      CASE (3)
        CALL ReadMatrixFile( FileName, KMat, iSpeed, NSpeed, NSTATE, Nwyes, NAzim )
      CASE (4)
        CALL ReadMatrixFile( FileName, GMat, iSpeed, NSpeed, NAlphas, NSTATE, NAzim )
      CASE (5)
        CALL ReadMatrixFile( FileName, AoAMat, iSpeed, NSpeed, NAzim, 3, 1 )
      CASE (6)
        CALL ReadOpsFile( FileName, ThetaOps, iSpeed, NSpeed )
    END SELECT

  END DO ! iFile

END DO ! iSpeed

WRITE(*,*) TRIM(Int2LStr(NMats))// " matrices read for "//TRIM(Int2LStr(NSpeed))// " different
wind speeds."

!       WRITE(*,*) 'A matrix last element = ', AMat(NSTATE, NSTATE, NAzim)

RETURN

END SUBROUTINE ReadMatrices

! -----

SUBROUTINE WriteMatrices(NSTATE)
! Writes selected columns from one matrix

```

```

USE GSEMatrices

IMPLICIT NONE

INTEGER    NSTATE
INTEGER    IState
INTEGER    i
INTEGER    j
INTEGER::  iUnit = 98

open(file = 'matrixcol.plt', unit=iUnit)

WRITE(*,*) 'Enter the State index of the matrix you want to write'
READ(*,*) IState

do i = 1, NAzim

    WRITE(iUnit,'(E11.4,4(A, E11.4))') Amat(1,IState, 2, i)    !, CHAR(9), &
    !           Kmat(IState, 2, i), CHAR(9), &
    !           Kmat(IState, 3, i), CHAR(9), &
    !           Kmat(IState, 4, i), CHAR(9), &
    !           Kmat(IState, 5, i)

enddo

CLOSE(iUnit)

RETURN

END SUBROUTINE WriteMatrices

! -----

SUBROUTINE ReadMatrixFile(FileName, Matrix, iSpeed, Ns, Ni, Nj, Nk)
! Reads one ascii text file and returns values in Matrix
! Also checks sizes of each dimension of the array

USE GSEMatrices

IMPLICIT NONE

INTEGER          IERR
INTEGER          iSpeed

INTEGER          Ni
INTEGER          Nj
INTEGER          Nk
INTEGER          Ns

INTEGER          NiRead
INTEGER          NjRead
INTEGER          NkRead

INTEGER          i
INTEGER          j
INTEGER          k

INTEGER::        Unit = 99

DOUBLE PRECISION Matrix(Ns,Ni,Nj,Nk)

CHARACTER(*)     FileName

! Open main input file
CALL OpenInputFile(Unit, FileName, 'FORMATTED')

READ(Unit, *) NiRead
IF( NiRead .NE. Ni ) THEN
    WRITE(*,*) 'Error reading file named ', FileName
    CALL ABORT( 'Error reading first index from this file' )
ENDIF

```

```

READ(Unit, *) NjRead
IF( NjRead .NE. Nj ) THEN
    WRITE(*,*) 'Error reading file named ', Filename
    CALL ABORT( 'Error reading second index from this file' )
ENDIF

READ(Unit, *) NkRead
IF( NkRead .NE. Nk ) THEN
    WRITE(*,*) 'Error reading file named ', Filename
    CALL ABORT( 'Error reading third index from this file' )
ENDIF

DO k = 1,Nk
    DO i = 1,Ni
        READ(Unit, *, END = 100) (Matrix(iSpeed,i,j,k), j=1,Nj)
    ENDDO
ENDDO

Close (Unit)

RETURN

100 WRITE(*,*) 'Error reading file named ', Filename
    CALL Abort(' Hit end-of-file while reading matrix file')

RETURN

END SUBROUTINE ReadMatrixFile

```

! -----

```

SUBROUTINE ReadOpsFile(Filename, Matrix, iSpeed, Ns)
! Reads one ascii text file and returns values in Matrix
! Also checks sizes of each dimension of the array

USE GSEMatrices

IMPLICIT NONE

INTEGER          IERR
INTEGER          iSpeed

INTEGER          Ns

INTEGER          i

INTEGER::        Unit = 99

DOUBLE PRECISION Matrix(Ns,3)

CHARACTER(*)     FileName

! Open main input file
CALL OpenInputFile(Unit, FileName, 'FORMATTED')

DO i = 1,3
    READ(Unit, *, END = 100) Matrix(iSpeed,i)
ENDDO !i

Close (Unit)

RETURN

100 WRITE(*,*) 'Error reading file named ', Filename
    CALL Abort(' Hit end-of-file while reading matrix file')

RETURN

```

```
END SUBROUTINE ReadOpsFile
```

```
! -----
```

```
SUBROUTINE GSEXX (ID, TIME, PAR, NPAR, IFLAG, NSTATE, &  
  STATES, NINPUT, INPUT, NOUTPT, PXXMAT)
```

```
USE GSEMatrices
```

```
IMPLICIT NONE
```

```
! === External variables ===
```

```
INTEGER ID  
INTEGER NPAR  
INTEGER NINPUT  
INTEGER NSTATE  
INTEGER NOUTPT  
DOUBLE PRECISION TIME  
DOUBLE PRECISION PAR(NPAR)  
DOUBLE PRECISION STATES(NSTATE)  
DOUBLE PRECISION INPUT(NINPUT)  
DOUBLE PRECISION PXXMAT(NSTATE,NSTATE)  
LOGICAL IFLAG
```

```
IF(IFLAG .OR. index .EQ. 0) RETURN
```

```
! Evaluate partial derivative matrix
```

```
PXXMAT = AMatNow
```

```
RETURN
```

```
END SUBROUTINE GSEXX
```

```
! -----
```

```
SUBROUTINE GSEXU (ID, TIME, PAR, NPAR, IFLAG, NSTATE, &  
  STATES, NINPUT, INPUT, NOUTPT, PXUMAT)
```

```
USE GSEMatrices
```

```
IMPLICIT NONE
```

```
! === External variables ===
```

```
INTEGER ID  
INTEGER NPAR  
INTEGER NINPUT  
INTEGER NSTATE  
INTEGER NOUTPT  
DOUBLE PRECISION TIME  
DOUBLE PRECISION PAR(NPAR)  
DOUBLE PRECISION STATES(NSTATE)  
DOUBLE PRECISION INPUT(NINPUT)  
DOUBLE PRECISION PXUMAT(NSTATE,NINPUT)  
LOGICAL IFLAG
```

```
IF(IFLAG .OR. index .EQ. 0) RETURN
```

```
! Evaluate partial derivative matrix
```

```
PXUMAT = 0.0      !Many are zero  
PXUMAT(1:NSTATE,1) = BMatNow(1:NSTATE,1) + KMatNow(1:NSTATE,1)  !Azimuth  
PXUMAT(1:NSTATE,2) = BMatNow(1:NSTATE,2) + KMatNow(1:NSTATE,2)  !Omega
```

```
RETURN
```

```
END SUBROUTINE GSEXU
```

```
! -----
```

```

SUBROUTINE GSEYX (ID, TIME, PAR, NPAR, IFLAG, NSTATE, &
    STATES, NINPUT, INPUT, NOUTPT, PYXMAT)

USE GSEMatrices

IMPLICIT NONE

! === External variables ===

INTEGER ID
INTEGER NPAR
INTEGER NINPUT
INTEGER NSTATE
INTEGER NOUTPT
DOUBLE PRECISION TIME
DOUBLE PRECISION PAR(NPAR)
DOUBLE PRECISION STATES(NSTATE)
DOUBLE PRECISION INPUT(NINPUT)
DOUBLE PRECISION PYXMAT(NOUTPT,NSTATE)
LOGICAL IFLAG

IF(IFLAG .OR. index .EQ. 0) RETURN

! Evaluate partial derivative matrix

PYXMAT = GMatNow

RETURN
END SUBROUTINE GSEYX

! -----

SUBROUTINE GSEYU (ID, TIME, PAR, NPAR, IFLAG, NSTATE, &
    STATES, NINPUT, INPUT, NOUTPT, PYUMAT)

USE GSEMatrices

IMPLICIT NONE

! === External variables ===

INTEGER ID
INTEGER NPAR
INTEGER NINPUT
INTEGER NSTATE
INTEGER NOUTPT
DOUBLE PRECISION TIME
DOUBLE PRECISION PAR(NPAR)
DOUBLE PRECISION STATES(NSTATE)
DOUBLE PRECISION INPUT(NINPUT)
DOUBLE PRECISION PYUMAT(NOUTPT,NINPUT)
LOGICAL IFLAG

IF(IFLAG .OR. index .EQ. 0) RETURN

! Evaluate partial derivative matrix

PYUMAT      = 0.0           !Most are zero
PYUMAT(1,3) = 1.0           !Partial theta wrt theta terms
PYUMAT(2,4) = 1.0           !Partial theta wrt theta terms
PYUMAT(3,5) = 1.0           !Partial theta wrt theta terms

RETURN
END SUBROUTINE GSEYU

! ----- End of File -----

```

## APPENDIX B.3: EXCERPT OF THE ADAMS MODEL DATASET PERTAINING TO THE PITCH CONTROL SYSTEM

These ADAMS dataset statements were copied from the model. They illustrate the communication between the model of the turbine and the controller GSESUB subroutine.

```

!
!----- DATA STRUCTURES -----
!
! X is gen shaft speed, rad/s. Y is generator torque, kN-m
!                               adams_view_name='torq_spd'
!                               adams_view_units='no_units'
SPLINE/1
, LINEAR_EXTRAPOLATE
, FILE = 'elong_speed_torque.dat'
!
!   PID pitch demand used during startup
!   User arguments are (ID number=10, control start time)
!                               adams_view_name='pitch_demand'
VARIABLE/10
, IC = 0
, FUNCTION = USER(10, 12)
!
! AoA controller variables:
!Azimuth in rad, 0 rad at 12 o'clock
!                               adams_view_name='GSE_Azim'
VARIABLE/1000
, IC = 0
, FUNCTION = AZ(3051,2050) + PI
!
!Rotor speed in rad/s
!                               adams_view_name='GSE_Omega'
VARIABLE/1001
, IC = 0
, FUNCTION = WZ(3051,2050,3051)
!
!Blade 1 current pitch
!                               adams_view_name='GSE_Theta1'
VARIABLE/1002
, IC = 0
, FUNCTION = AZ(4192,4161)
!
!Blade 2 pitch
!                               adams_view_name='GSE_Theta2'
VARIABLE/1003
, IC = 0
, FUNCTION = AZ(4292,4261)
!
!Blade 3 pitch
!                               adams_view_name='GSE_Theta3'
VARIABLE/1004
, IC = 0
, FUNCTION = AZ(4392,4361)
!
! Set AoA control start time here (used in three variables below)
! Must be >= AoA start time in GSE statement
! At this time the transition to total control by AoA controller is completed.
!   The transition begins 4s before this time.
!                               adams_view_name='AoA_start_time'
VARIABLE/1010
, FUNCTION = 26.
!
! Pitch error for each blade. VARVAL(10) is PID demand for all blade
!   ARYVAL(1002,n) is AoA controller demand for blade #n
! Transition from PID to AoA control uses STEP functions, starting at (AoA_start_time - 4.0)
!                               adams_view_name='PitchErr1'
VARIABLE/1011
, IC = 0
, FUNCTION =
, STEP(TIME, VARVAL(1010)-4., VARVAL(10)-VARVAL(1002), VARVAL(1010)-2.,

```

```

, STEP(TIME, VARVAL(1010)-1., VARVAL(10)+ARYVAL(1002,1)-ARYVAL(2000,1),
, VARVAL(1010), ARYVAL(1002,1)) - VARVAL(1002))
!
!
!           adams_view_name='PitchErr2'
VARIABLE/1012
, IC = 0
, FUNCTION =
, STEP(TIME, VARVAL(1010)-4., VARVAL(10)-VARVAL(1003), VARVAL(1010)-2.,
, STEP(TIME, VARVAL(1010)-1., VARVAL(10)+ARYVAL(1002,2)-ARYVAL(2000,2),
, VARVAL(1010), ARYVAL(1002,2)) - VARVAL(1003))
!
!
!           adams_view_name='PitchErr3'
VARIABLE/1013
, IC = 0
, FUNCTION =
, STEP(TIME, VARVAL(1010)-4., VARVAL(10)-VARVAL(1004), VARVAL(1010)-2.,
, STEP(TIME, VARVAL(1010)-1., VARVAL(10)+ARYVAL(1002,3)-ARYVAL(2000,3),
, VARVAL(1010), ARYVAL(1002,3)) - VARVAL(1004))
!
!
! ----- ARRAYS AND TRANSFER FUNCTIONS -----
!
!           Used to integrate pitch error
!           adams_view_name='PID_pitch1_x'
ARRAY/100
, X
!
!           adams_view_name='PID_pitch1_u'
ARRAY/101
, U
, SIZE = 1
, VARIABLES = 1011
!
!           adams_view_name='PID_pitch1_y'
ARRAY/102
, Y
!
!           adams_view_name='PID_pitch2_x'
ARRAY/200
, X
!
!           adams_view_name='PID_pitch2_u'
ARRAY/201
, U
, SIZE = 1
, VARIABLES = 1012
!
!           adams_view_name='PID_pitch2_y'
ARRAY/202
, Y
!
!           adams_view_name='PID_pitch3_x'
ARRAY/300
, X
!
!           adams_view_name='PID_pitch3_u'
ARRAY/301
, U
, SIZE = 1
, VARIABLES = 1013
!
!           adams_view_name='PID_pitch3_y'
ARRAY/302
, Y
!
!           adams_view_name='GSE_x'
ARRAY/1000
, X
!
!           adams_view_name='GSE_u'
ARRAY/1001
, U

```



```

, SIZE = 5
, VARIABLES = 1000, 1001, 1002, 1003, 1004
!
!                               adams_view_name='GSE_y'
ARRAY/1002
, Y

!
!                               adams_view_name='Theta_Ops'
!ARRAY/2000
!, IC
!, SIZE = 3
!, NUMBERS = 0.2794206, 0.2794206, 0.2794206
!
!                               adams_view_name='W_Ops'
ARRAY/2001
, IC
, SIZE = 3
, NUMBERS = 16, 0, 0
!
!                               adams_view_name='Omega_Ops'
ARRAY/2002
, IC
, SIZE = 1
, NUMBERS = 1.94807
!
!----- EQUATIONS -----
!
!----- GSE -----
!
!                               adams_view_name='GSE_pitch'
GSE/1000
, NS = 11
, NO = 3
, X = 1000
, XXFLAG = USER
, U = 1001
, Y = 1002
, XUFLAG = USER
, YUFLAG = USER
, YXFLAG = USER
, FUNCTION = USER(11, 20, 0.005, 0.05, 0, 1)
!
!args are 1: AoA el #. 2: Control start time 3:update Del t 4:output Del t(0 for none) 5:
Azimuth offset, rad 6: G matrix Gain
!
!                               1/s = integrator for pitch error
!                               adams_view_name='TFS100'
TFSISO/100
, X = 100
, U = 101
, Y = 102
, NUMERATOR = 1, 0
, DENOMINATOR = 0, 1
!
!                               adams_view_name='TFS200'
TFSISO/200
, X = 200
, U = 201
, Y = 202
, NUMERATOR = 1, 0
, DENOMINATOR = 0, 1
!
!                               adams_view_name='TFS300'
TFSISO/300
, X = 300
, U = 301
, Y = 302
, NUMERATOR = 1, 0
, DENOMINATOR = 0, 1

```

## **APPENDIX C.1: BLADESTRUCTURALCALC.DOC FOR WINDPACT ROTOR DESIGN**

### ***INTRODUCTION***

The spreadsheet-based structural design tool, co-developed by M. Zuteck and D. Griffin for use in the WindPACT Blade Scaling Study, has been reformatted and modified for use under the WindPACT Rotor Design Study. The following presents a summary of the current spreadsheet, including an explanation of the calculations, approximations, assumptions, hard-wired input, and user-driven options.

The current code revision is named “NR4-2 Blade Run 01.xls”. The data contained in this spreadsheet is intended to represent the initial blade used for the 750 kW baseline turbine model. Prior to discussing the details of the spreadsheet input/output, some general background will be provided on the structural modeling and analysis approach used.

### ***MODELING APPROACH***

#### ***Overview***

The scaling model was constructed based on direct aerodynamic and structural design calculations. A baseline set of design parameters (airfoil shapes, structural arrangement, materials, and manufacturing method) was selected to be consistent with current commercial designs. Using the baseline configuration, a matrix of aerodynamic and structural designs were developed and analyzed. The results were used to construct a model that can be used in an inverse fashion – solving for the required structure when given geometry, loads, and structural design criteria as input.

Often, a blade structural model may be based on a greatly simplified sectional representation so that a closed-form solution is obtained and/or computational complexity is minimized. A simplified sectional representation is particularly useful if structural calculations are desired at a large number of blade spanwise stations.

A distinctly different modeling approach has been taken in the present work. A series of detailed structural analyses were performed at relatively coarse spacing in the spanwise direction (5, 7, 25, 50, 75, and 100%  $r/R$ ), and the results of those analyses were used to infer the blade properties at intermediate stations. The following sections describe the elements of the model and how those elements are integrated in the computational modeling tool.

#### ***Aerodynamic Design***

The turbine configuration selected for this study is a three-bladed, upwind rotor with a rigid hub, full-span pitch control, and full variable-speed operation. Aerodynamic performance for each design is indicated by maximum rotor power coefficient,  $C_{pmax}$ . Rotor dimensions are described in terms of radius ( $R$ ), and sizing assumes a constant specific power rating of  $0.44 \text{ kW/m}^2$ . The resulting relationship between  $R$  and rotor rating is summarized in Table 1.

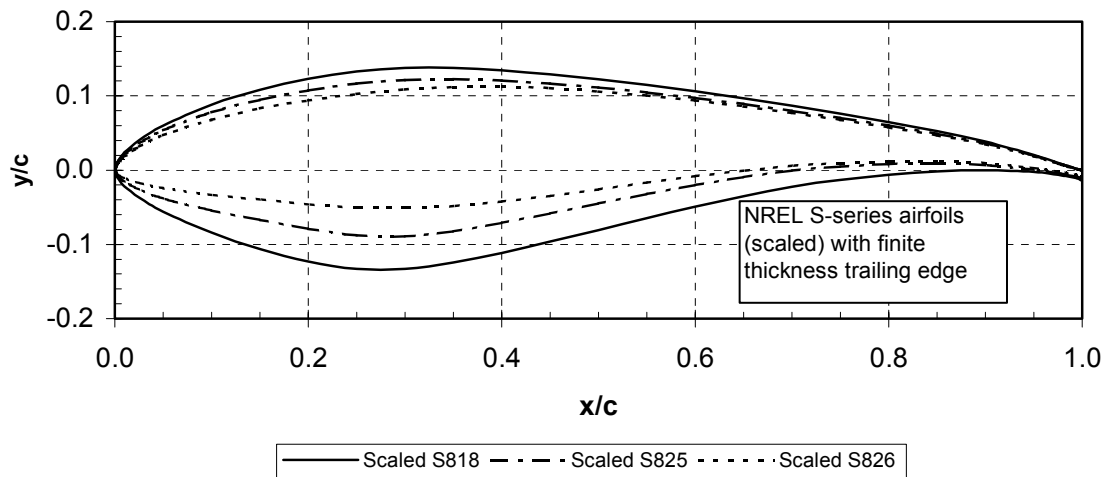
**Table 1. Rotor Sizing**

Radius (m)	Rating (MW)
23.3	0.75
32.9	1.5
42.5	2.5
60.2	5.0

Several of the NREL S-series airfoils were considered for use in this study.<sup>1</sup> The S818/S825/S826 family was identified as having desirable aerodynamic properties, but the airfoils were deemed to be too thin for efficient application to large blades (assuming current commercial materials are used). A more structurally suitable set of airfoil shapes was therefore derived by scaling the S818/S825/S826 foils and by adding a finite-thickness trailing edge. The shape modifications and locations of airfoils along the blade are summarized in Table 2, and the resulting shapes are shown in Figure 1. Aerodynamic properties for the modified shapes were calculated using the Eppler Design and Analysis code.<sup>2</sup>

**Table 2. Airfoil Shape Modifications**

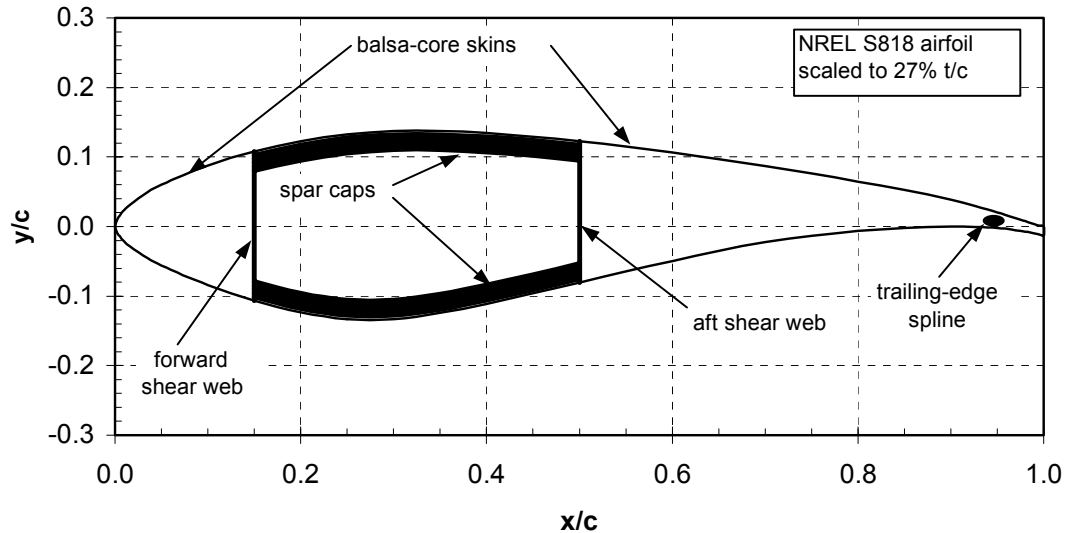
Airfoil	R (%)	Orig. t/c (%)	Scaled t/c (%)	Trailing edge (% c)
S818	25	24	27	1.3
S825	75	17	21	1.0
S826	95	14	16	0.75

**Figure 1. Airfoils used for baseline blade model**

The PROPID Design and Analysis code was used to develop aerodynamic designs for a wide range of rotor ratings and planforms.<sup>3</sup> In each case, the PROPID code was used to determine a near-optimal schedule of blade chord and twist. Note that aerodynamic design calculations did not include the optimization of blade stations inboard of 25% r/R. For analysis purposes, the blade was assumed to taper from the 25% r/R section to a circular root located at 5% r/R.

### Baseline Structural Model

A baseline structural arrangement was selected as being representative of current commercial blade designs. The primary structural member is a box-spar, with webs at 15% and 50% chord. The skins are sandwich-construction with balsa core. This arrangement is depicted in Figure 2, where the thickest airfoil section (25% span station) is shown.



**Figure 2. Arrangement of baseline structural model**

As seen in the figure, the assumed placement of the shear webs results in a relatively wide spar. As a result, the spar cap material provides a significant contribution to the edgewise-bending strength. Also note that due to the deep contour on the lower airfoil surface, the aft portion of the shear web will be less effective in flap bending than if it were concentrated in the deepest part of the foil (farther from the neutral bending axis). Also note that Figure 2 shows a “trailing-edge spline.” This represents material added for reinforcement in edge bending.

Table 3 lists the layers in the baseline structural shell and describes the material contained in each. The dimensions given in Table 3 were used in structural calculations for 750-kW rotor blades. The shear web cores were assumed to be 1% c thick for all rotor sizes, but the thickness of the triaxial shear-web skins was increased for larger blade sizes.

**Table 3. Structural Shell Definition (750 kW)**

Layer #	Material	Thickness
1	gel coat	0.51 mm
2	random mat	0.38 mm
3	triaxial fabric	0.89 mm
4		
0-15% c	balsa	0.5% c
15-50% c	spar cap mixture	specified % t/c
50-85% c	balsa	1.0% c
5	triaxial fabric	0.89 mm

The skins and spar cap are E-glass / epoxy laminate. The triaxial fabric is designated CDB340 and has a 25/25/50 percent distribution of +45°, -45° and 0° fibers. The spar cap is composed of alternating layers of triaxial and uniaxial (C260) fabric. This stacking sequence results in spar cap laminate with 70% uniaxial and 30% off-axis fibers by weight.

Characteristic material properties for these lamina were derived based on a combination of test data and laminate theory calculations.<sup>4,5</sup>

### **Calculation of Section Structural Properties**

Once the baseline model parameters were established, a series of direct point-designs were performed, each design providing a “base case” for developing the computational modeling tool.

Table 4 provides a summary of the preliminary structural base cases, where the ratings indicated are related to the physical dimensions used for corresponding section analyses. As indicated by the table, the section mechanical properties were computed for each spanwise station using spar caps that were 0%, 5%, and 10% of maximum section thickness. Second order curves were then generated for weight per unit length and “flapwise” and “edgewise” bending stiffness as a function of spar cap thickness.

**Table 4. Matrix of Structural Base Cases**

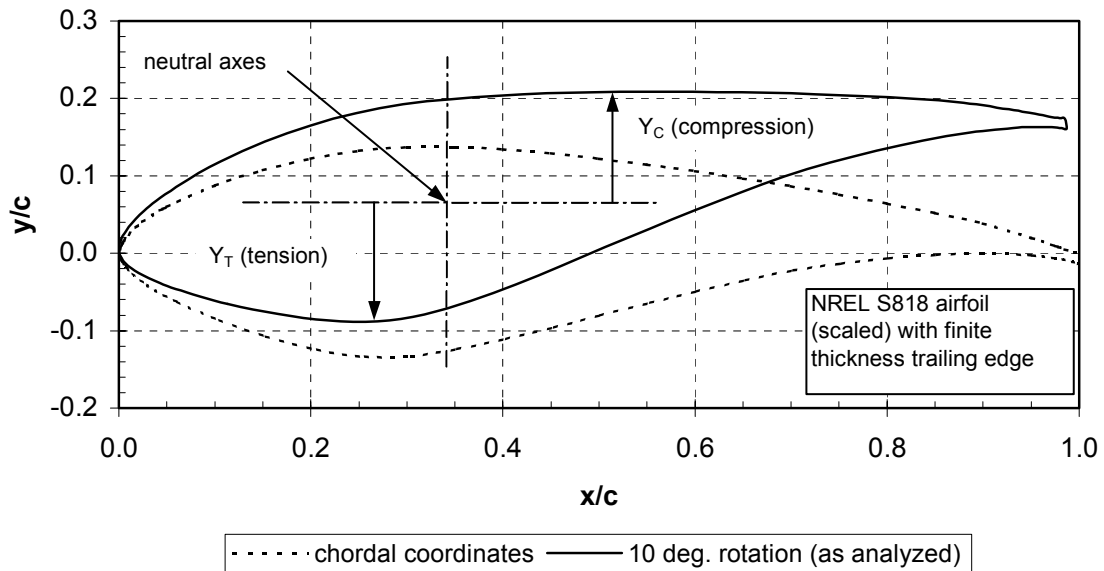
<b>Spanwise Stations</b>	<b>Airfoil t/c (%)</b>	<b>Spar Cap (% t)</b>	<b>Rating (MW)</b>
25%	27	0, 5, 10	0.75
	“	“	5.0
	30	“	0.75
	33	“	“
50%	36	“	“
	24	0, 5, 10	0.75
75%	“	“	5.0
	21	0, 5, 10	0.75
100%	“	“	5.0
	16	0, 5, 10	0.75
	“	“	5.0

An inverse curve-fit of the flapwise strength curves was then performed. With user-specified values of bending moment and design limits on material strain, the resulting curves allow calculation of the spar cap thickness required to support the load.

To allow investigation of the effect of t/c variation at the 25% r/R location (where the impact on weight and cost is most significant), additional structural base cases were computed for airfoil shapes scaled to 30% and 36% t/c. These results were similarly fit with second order curves as a function of spar cap thickness. A secondary curve fit was then developed to allow modeling of variations in both spar cap thickness and t/c parameters at the 25% r/R section.

To provide size scaling, structural base cases were also created for a blade size that represented a 5-MW turbine. By comparison to the 750-kW base cases, curve fits were developed to model the substantial differences that were shown in the underlying base cases. In developing the 5-MW section models, the spar caps, skins, and shear webs were treated independently and scaled as appropriate. This treatment of the 5-MW base cases resulted in the designs (correctly) departing from pure self-similarity as they scaled in size.

An important clarification must be made in the use of “flapwise” and “edgewise” structural properties. Although the terminology of flapwise and edgewise is used throughout this work, the associated properties (calculated and used in the structural scaling model) are more properly termed “out-of-plane” and “in-plane,” respectively. For each airfoil station, a twist angle was assumed and the structural analyses performed in the rotated orientation. Figure 3 illustrates this approach for the 25% span station, where the analyses were performed with a 10° rotation of the blade section. In determining the structural properties, moments were applied parallel to the original (untwisted) x-y axes. The structural analysis code determines the location of the neutral axis for the rotated section and calculated bending properties relative to those axes.



**Figure 3. Orientation of blade for "flapwise" and "edgewise" section properties (25% span station)**

As indicated in Figure 3, the critical fiber distances (tension and compression) are taken as the farthest material, in a perpendicular direction, from the out-of-plane neutral bending axis. Due to the rotated geometry, the critical tension and compression fibers will not occur at the same airfoil chord location, and they will be separated by a vertical distance greater than the maximum airfoil thickness.

Although the approach taken adds some complexity to the analysis and subsequent interpretation, it is intended to realistically portray the loading condition for the blade in service. Table 5 summarizes the twist angles used at each section for the base case calculations and also presents the near-optimal blade twist angles as determined by the PROPID aerodynamic code for two different values of design tip speed ratio. The data indicate that the twist angles assumed for the structural analysis are a close approximation to the near-optimal blade twist. Also note that the distinction between the flap-edge and in-out of plane properties is the greatest at the 25% span station, and it vanishes entirely at the 75% span station (where the twist is taken to be zero by definition).

**Table 5. Summary of Blade Section Twist Angles**

Station (%R)	Assumed Twist (Deg.)	PROPID Near-Optimal Twist (deg.)	
		Design TSR = 7	Design TSR = 8
25	10	10.5	9.3
50	2.5	2.5	2.1
75	0.0	0.0	0.0
100	-0.5	-0.6	-0.4

### **Load Cases**

The following paragraphs describe the loading conditions used in developing the initial blade structural designs. It should be noted, however, that the blade structural design spreadsheet allows user input of loads developed by alternate methods.

Two primary load cases were selected for the purposes of developing structural designs. Peak bending loads were derived using a 50-year extreme gust of 70 m/s (IEC Class 1).<sup>6</sup> The gust was assumed to occur with the blades in a fully feathered position, with a  $\pm 15^\circ$  variation in wind direction. To simplify the loads development, it was assumed that this load case resulted in each blade section simultaneously reaching its local maximum lift coefficient and that the bending loads were entirely in the flapwise direction. The resulting loads were summed over the blade to define characteristic peak bending moments at each blade station.

For edge bending, the designs were assumed to be governed by fatigue loading. A simplified loading spectrum was developed by combing the peak of gravity plus torque at rated power, assuming the loading is fully reversed. This load case was further simplified by assuming the rotor operated for 5000 hours per year at rated power, over a 20-year design life.

### **Partial Safety Factors**

The original intent for this project was to entirely use IEC partial safety factors. The IEC 61400-1 requires a “general” material factor of 1.1. The IEC standard further states that material factors shall be applied to account for “...scale effects, tolerances degradation due to external actions, i.e., ultraviolet radiation, humidity and defects that would not normally be detected.” However, the IEC document provides no specific guidance on appropriate values for these additional factors. Conversely, the Germanischer Lloyd (GL) standard provides an explicit list of partial safety factors for composite materials.<sup>7</sup> For an evaluation of static strength, the GL factors are:

- $\gamma_{M0}$  = 1.35 (general material factor)
- $C_{2a}$  = 1.50 (influence of aging)
- $C_{3a}$  = 1.10 (temperature effect)
- $C_{4a}$  = 1.20 (hand lay-up laminate)
- $C_{5a}$  = 1.10 (non post-cured laminate)

The GL standard further states that  $\gamma_{M0}$  is to be used in all cases but that the  $C_{ia}$  may be adjusted if demonstrated by experimental verification. Applying the GL factors as specified implies a combined material factor of 2.94.

For fatigue verification, the GL standard states that the same partial material factors shall be used, with the exception of the 1.5 factor for “aging.”

The GL partial load factor for an extreme 50-year event is 1.50, whereas the corresponding IEC load factor is 1.35. Although the majority of current commercial blades has been designed for certification

to the GL requirements, the IEC (extreme wind) load factor of 1.35 has been chosen for the present rotor design study calculations.

### ***Full-Blade Calculations***

A full blade calculation ties together all the model elements described above. User input to the spreadsheet-based calculation includes:

- 1) Rotor radius and rated power
- 2) Chord dimensions and flapwise design bending moments at 25%, 50%, and 75% span stations
- 3) Diameter and design bending loads for the circular root connection
- 4) Diameter of root studs and factors for scaling of laminate required for root/stud interface
- 5) Factor to account for “parasitic” weight due to excess material and bonds
- 6) Design values for peak laminate strain and  $\epsilon$ -N parameters for the edge bending fatigue calculation.

The following paragraphs provide a brief summary of the process used in a full-blade calculation. Further details of each calculation are given in the discussion on the spreadsheet input/output.

**Root Design** For the purposes of this study, a circular root bolt pattern was assumed, with diameter equal to 5.4% R (value inferred from commercial blade designs). Although root connection bolts are available in discreet size increments, the scaling calculations use continuous non-integer functions to model the root connection and associated weight. The laminate at the root is sized according to stud bonding requirements, with user-input scaling parameters. No strength verification calculation has been performed at the root, as preliminary calculations showed that typical commercial root designs greatly exceed the strength requirement implied by quasi-static analyses.

**“Inboard Blade”** The root connection build-up is assumed to taper away within a user-specified distance from the root (2% R used in present calculation). It is further assumed that the blade cross section at this station remains circular and is composed of a thick-laminate shell with the same composition as the spar cap material. Shell thickness at this station is determined by strength requirements, based on a direct calculation within the structural design spreadsheet.

**Remaining Blade Stations** At 25%, 50%, and 75% span, the blade section structure is determined using the design bending loads and maximum (design) strain limits. However, rather than using a direct calculation, the inputs are used in an inverse-process to determine the spar cap thickness to meet the required design criteria. Section structural properties ( $EI_{flap}$ ,  $EI_{edge}$  and weight per unit length) are then calculated at each section based on chord length and spar cap thickness. The design bending moments necessarily vanish at the tip section. Here the blade section properties are calculated based only on chord, with the assumption that the spar cap has entirely tapered away.

Following the initial structural sizing of the blade, additional verifications and/or adjustments are performed:

- 1) Fatigue life is verified for edgewise bending, assuming 5000 hours per year for a 20-year design life at an assumed constant-amplitude loading. If 20-year fatigue life is not achieved, then additional structure is added at 95% span (trailing-edge spline) until the edge-fatigue criteria are met.
- 2) Tip deflections are calculated for the 50-year extreme loads. For the purposes of this work, our guideline was that the maximum tip deflection should not exceed 10% of the rotor radius. This



criteria was not rigorously applied in determining blade structure but was rather used comparatively in assessing blade design variations. However, an optional “escalating factor” is included in the design spreadsheet so that blade structure may be increased by a (constant) multiplier on the design loads and the impact on tip deflections can be assessed.

- 3) An option exists for a (constant) multiplier on the blade section weight. This was included to account for “parasitic” weight due to bonding material and inevitable inefficiencies in material usage. In the majority of work to date, the parasitic weight adjustment was estimated as a constant 10% of the total blade weight. In practice, the amount of non-structural material in a blade will be determined by a trade-off between labor and material costs. Although the manufacturing economics are likely to favor additional labor expenditures to save material at large scales, no attempt was made to quantify that effect.

It should be noted that, in the absence of a parasitic weight adjustment, each calculated weight is a theoretical minimum for the configuration and materials modeled, and in practice other design considerations such as flap fatigue or load path requirements may result in higher weight values.

### **STRUCTURAL CALCULATION SPREADSHEET**

The following sections provide a detailed explanation for the input/output (I/O) cells of the structural design spreadsheet (current revision named “NR4-2 Blade Run 01.xls”) and the underlying calculations. The values as shown in the current spreadsheet are intended to represent the baseline 750 kW rotor blade, used for initial input in the WindPACT Rotor Design Study.

This discussion will focus on the I/O contained on the front page of the spreadsheet, titled “blade properties.” However, many supporting or supplementary calculations occur on the five additional sheets contained in the workbook. Due to the many complex links involved, an I/O format is used: green font denotes user-specified input, black font indicates that the values are either a calculated result or are hard-wired, and red font is an alert to the user.

#### ***Notes Block***

Located at the top of the sheet, this is a single-line full-page width block for user-specified notes.

#### ***Main Rotor Input Block***

The majority of input cells in this block are self-evident. The top of the first column includes rotor rating, radius, and rotational speed. The next three entries relate to the fatigue verification for edgewise bending and will be discussed in the following section. The chord and design-moment distribution is input at a pre-specified schedule of stations. Note that the chord at 5% R (i.e., the blade root diameter) is not directly input but calculated based on the rotor radius and input in the root calculation block. Also note that tip-bending moment is hard-wired as zero.

#### ***Material Design Strength***

For static loading, the material design strength is input in units of microstrain, for both tension and compression. For blade sections that include structural spars (25%, 50%, and 75% R), the spar cap sizing and associated structural properties are determined based on a combination of chord, moment, and design strain. Both tension (lower blade surface) and compression (upper blade surface) are used to evaluate the required spar structure, and the most demanding case is used to define the actual blade design. It should be noted that for the assumed baseline blade material, we found that all blade section designs were governed by compressive strain limits.

The edge fatigue design strength is based on a combination of the laminate strain-cycle curve and the simplified edge-bending fatigue spectrum. However, prior to describing the full calculation, a summary of the design material properties will be presented.

Characteristic strength properties were derived for the baseline E-glass/epoxy laminate by John Mandell, who based them on a combination of MFG and MSU test data and laminate theory. The strength values assume that the CDB340 triaxial lamina are the “weak link” in both tension and compression, for both the skins and the spar cap build-up.

Partial material factors were developed based on the values specified by GL. For static calculations, a combined material factor of 2.9 has typically been used. However, as the design strain values are direct user input, the value of partial material factor (for static analysis) may be varied accordingly. The fatigue calculations that are imbedded in the current code assume that the partial material factors for fatigue are equal to (2.9/1.5), and this value is hard-wired in the current version of the code. Table 6 summarizes the characteristic and design laminate strength values that have been used to develop the baseline blade design.

**Table 6. Design Values for Laminate Strength**

Loading	Strength ( $\mu\epsilon$ )		
	Characteristic	Design (static)	Design (fatigue, single cycle)
Tension	22,000	7586	11,379
Compression	10,500	3620	5431

Fatigue curves are available from two sources: the original  $\epsilon$ -N curves supplied by John Mandell and the subsequent “compromise” curves developed by David Malcolm.<sup>8</sup> Both equation sets are in the form of normalized strain. In the case of J. Mandell, all  $\epsilon$ -N curves are normalized to tensile strain limits, with the equations given below.

J. Mandell equation set:

$$\begin{aligned}
 R= 0.1 \quad \frac{\epsilon}{\epsilon_o} &= 1 - 0.147 \cdot \text{LOG}(N) && [N < 1000] \\
 \frac{\epsilon}{\epsilon_o} &= 1.6586 \cdot N^{-0.1574} && [N > 1000] \\
 R= 10 \quad \frac{\epsilon}{\epsilon_o} &= 0.48 - 0.432 \cdot \text{LOG}(N) \\
 R= -1 \quad \frac{\epsilon}{\epsilon_o} &= [0.48 - 0.432 \cdot \text{LOG}(N)] - 0.05
 \end{aligned}$$

In developing an alternate set of  $\epsilon$ -N curves, it was desired that the curves be in log-log form, with an intercept of unity at zero cycles. D. Malcolm accordingly developed a set of compromise curves of the form:

$$\frac{\epsilon}{\epsilon_o} = 1.0 \cdot N^{-1/m}$$

Table 7 summarizes the slope and associated (single cycle) normalization strain for the three loading ratios considered.

**Table 7. Summary Parameters for D. Malcolm e-N Curves**

<b>Loading Ratio, R</b>	<b>Slope, m</b>	<b>Single Cycle Limit</b>
0.1	8.04	Tension
10	16	Compression
-1	13	Compression

Returning to the blade design spreadsheet I/O, the first column of the “rotor input” block contains three entries concerning the edge fatigue calculations. The value shown for N is the calculated number of fully reversed edge-bending cycles, according to the rotor speed and assumed loading conditions (5000 hours per year at rated power over 20 years); m is the slope of the laminate strain-cycle curve (13 for fully reversed loading as indicated by Table 7); and  $\epsilon/\epsilon_0$  is the normalized design strain allowable for N cycles. The normalized edge fatigue design strain is converted to  $\mu\epsilon$  and output to the lower left of the rotor input block. For the example shown, the edge fatigue design strength is  $(0.234)(5431 \mu\epsilon) = 1270 \mu\epsilon$ . This value is used (further down in the spreadsheet) to determine whether a blade structure has sufficient edge bending strength to meet the design requirement and to size the reinforcement required if the edge fatigue criteria is not met.

### **Root Calculation Block**

Early in the WindPACT blade scaling project, it was determined that typical commercial turbine blades have root connections that substantially exceed the strength requirements implied by peak static loading. D. Griffin hypothesized that current commercial blade root designs are governed by GL requirements for fatigue assessment of the bolted connection. However, a full resolution of this issue was beyond the scope of work for the blade scaling study. As such, the available data for commercial blade root designs were analyzed, and the trends were used to develop empirical root-sizing relationships.

The first user input for root sizing is the bolt circle diameter, where a value near 5.4% R was observed for a wide range of commercial rotor sizes. The second input is bolt diameter. The value of 1.18 inch (30 mm) is typical for blades in the 750-kW size range. The next two input cells in the “root calculation” block involve the sizing of root studs and bonding laminate. The values shown indicate that the stud O.D. is assumed to be twice the bolt diameter, and the surrounding laminate shell (total thickness) is assumed to be 1.5 times the stud O.D. The final input cell is the laminate density, used in calculating the root weight.

Although no structural verification is done for the root section (based on formal moment-strain analysis), the section-bending properties ( $EI_{flap}$ ,  $EI_{edge}$ , and weight per unit length) are calculated within the spreadsheet. For the weight calculation, the stud weight is calculated and included (procedure described later). However, the present code calculates root section EI values based entirely on the laminate shell, and it does not account for the presence of the steel studs.

### **Inboard Blade Calculation Block**

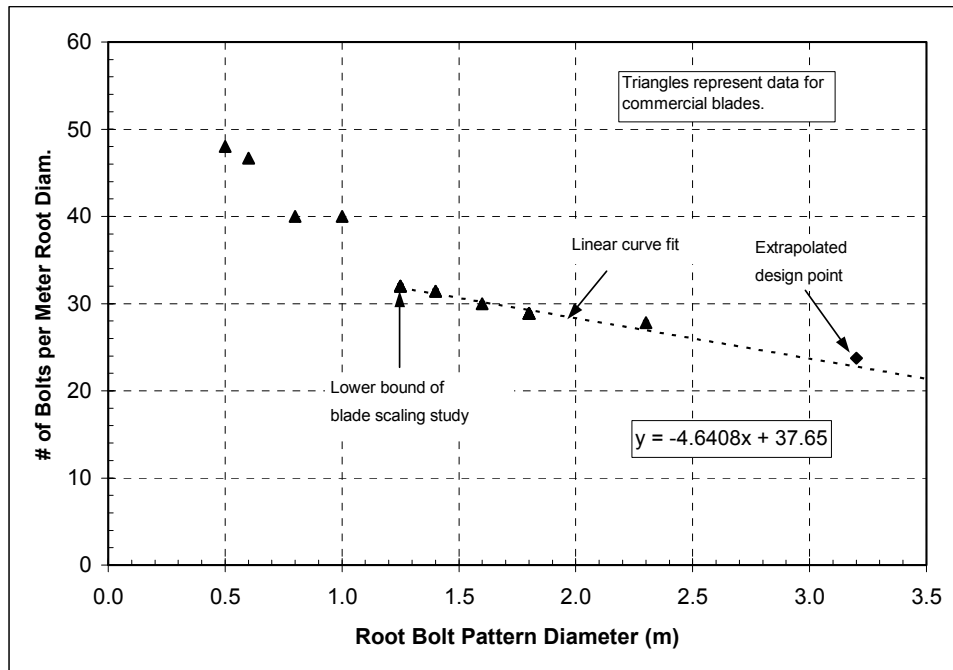
As noted above, the current model sizes the blade root based on trends inferred from current commercial blades and on requirements for bonding of the embedded studs. Although this approach faithfully models the root sizing trends for commercial blades, it was determined that without an intermediate station between the root and the 25% span location, the entire blade properties were too strongly dependent on assumptions made in the root sizing. To mitigate this effect, an additional calculation was made for a station near the root, but one at which it is assumed that root connection considerations are no longer dominant.

The “inboard blade connection” block has a single input: the distance in percent span from the root to the intermediate station. Hard-wired assumptions are that the blade at this station has remained circular, has a thick shell with the same laminate composition as the spar cap, and that the shell mid-section has the same diameter as the root bolt pattern.

The design bending moment at the intermediate inboard blade section is determined by an exponential curve fit between the 5% (root) and 25% span stations. With the design moment and laminate shell center fixed, the required thickness is determined such that the compressive design strain is not exceeded at the shell surface. Subsequently, the corresponding section EI and weight per unit span are calculated.

**Stud Calculation Block**

The “stud calculation” block is used to determine the net weight added to the blade root due to the embedded studs. No user input is required in this block. The root diameter and stud I.D. are directly linked to other cells and converted to SI units. The number of studs per blade is calculated based on a linear curve fit to data for current commercial blades, as indicated in Figure 4. Note that although in practice studs will necessarily occur in integer numbers, the curve-fit relationship used to model root connections allows for non-integer numbers of studs. Data for 7.5-inch long AWT-26/27 studs were used to derive a relationship between stud I.D., weight, and volume. These relationships were used to develop scaling relationships, with the assumption that the stud volume will scale as the cube of the I.D. The final output from the stud calculation is the net weight of the embedded studs (stud weight less the weight due to replaced laminate). These values are used in calculating the blade total weight and cost calculations.



**Figure 4. Bolt/root diameter relationship**

**TE Spline Calculation Block (and Edge Fatigue Flag)**

The trailing edge (TE) spline calculation requires no user input. The majority of the TE spline calculation occurs on the second page of the blade design spreadsheet, titled “underlying calcs,”

where the combined edge bending due to gravity and aerodynamic torque is calculated for each section. Also, at each section the  $EI_{\text{edge}}$  and design edge-bending strength (strain allowable for  $N$  cycles as calculated above) are used to determine the maximum allowable cyclic edge bending moment.

The actual edge bending moments are compared with the design values to determine whether the edge bending strength is sufficient. If the original structure is adequate in edge bending, then a “capacity deficit” value of zero will be returned to the “TE spline calc” block on the spreadsheet front page, and the “edge fatigue flag” block will remain empty.

If the actual edge bending moment exceeds the design value, then the difference (actual minus design moment, in inch-lbs) will be returned to the capacity deficit cell and the edge fatigue flag will display “EDGEWISE LIMIT EXCEEDED” in red font. The edge bending moment deficit is then used to calculate the amount of TE spline material that is required to recover the required edge bending strength. This calculation assumes that the TE spline material has a tensile modulus equal to the A260 unidirectional fabric and is centered at the 95% chord location. It is further assumed that the edgewise bending neutral axis (NA) is located at 35% chord and that the shift in edge NA due to the introduction of TE spline is negligible. Based on these assumptions, the TE spline cross-sectional area and weight per foot are calculated and added by linear superposition to the blade properties.

Note that although the “edge fatigue flag” is linked to a verification of edge bending capacity at all non-root stations, in the course of blade scaling calculations the 25% span section is the only station that has been identified as edge-critical. As such, the TE spline calculation has been hard-wired for the 25% span section. Also note that for the baseline materials and structural arrangement, no blades have been identified as edge-critical for ratings below about 2 MW.

### ***Deflection Calculation Block***

The deflection calculation occurs entirely on a spreadsheet page titled “stiffness calc.” Design moments and  $EI_{\text{flap}}$  are copied (via linked cells) to the stiffness calculation page. These values are interpolated via piecewise exponential curves to blade stations at 5% span increments. A double integration of  $M/EI_{\text{flap}}$  is then performed along the blade to calculate slope and deflection at each station. The tip deflection is normalized with respect to the undeflected rotor diameter and returned to the front page as a “deflection fraction” percentage.

As noted above, a general guideline established for the rotor scaling study is that tip deflections should not exceed 10% of the undeflected radius. However, this criteria was not rigorously enforced in deriving blade designs. For the current (baseline) blade example, the calculated deflection is 13%  $R$ . To investigate the implications (i.e. on  $EI_{\text{flap}}$  and weight) of requiring lower deflections, a “load escalating factor” input has been included. A non-unity value in this cell will scale the design moments at each station by the input value, which will in turn scale the blade structure as required to support the adjusted loads. In assessing the tip deflection, the load-escalating factor is scaled back so that the deflection is calculated based on the original design loads.

The load-escalating factor is only properly used to investigate the situation in which stiffness dominates the design. Adjustments to the design loads (i.e., through use of different load factors) should be implemented by direct modification of the design moments input to the “rotor input” block. It should also be noted that the use of a global escalating factor is only a first estimate of the implications of requiring additional stiffness, where the stiffness of each blade section is increased in direct proportion. In practice, an infinite number of design combinations may be used to increase the total blade stiffness. In addition to performing the deflection calculation, the “stiffness calc” page

converts the mass and stiffness data into a format suitable for input into the NREL-developed MODES code.

### **Summary Block**

This block pulls together all section properties that are calculated in the various spreadsheet blocks and formulas. The station and chord dimensions are self-evident. The moment values shown are the design values multiplied by the load escalation factor. The intermediate inboard station (7% R in current sheet) is calculated based on the value of “root drop-off region” that was input in the “inboard blade calculation” block. The design moment at the intermediate station is calculated by an exponential fit between the root and the 25% span station. Spar cap thicknesses are expressed as a percentage of the total blade thickness at each section. EI (flap and edge) and weight per unit span are tabulated, including adjustments for root studs (weight only), TE spline, and load escalation for increased stiffness.

The only effect not yet considered in the “summary” block is the adjustment due to “parasitic” weight, which is implemented in the following block. Also, due to the large number of calculations that are linked to the summary block, the original US system of units has been maintained in those cells.

### **Blade Weight Calculation Block**

This block allows a user-input scaling factor for “parasitic” weight, which is weight due to bonding material and the inefficiency in structural placement of material (i.e. material placed in mold to avoid the labor required for fabric cutting). The present baseline calculation assumes a 10% increase in total blade weight due to this effect. The adjusted weight per unit span is then summed over the blade length, and a conversion to SI units is made. To facilitate cost calculations, the blade structure and root connection weights are calculated separately, then summed to derive the total blade weight. In the present format, the root connection weight is calculated as the net weight added due to the root studs (steel less replaced laminate). This approach allows the two values to be added in deriving the total blade weight, but would require a supplementary calculation to correctly determine the weight contributions of the studs (steel) and blade root laminate.

### **Moment-Strain Relationship Block**

This block calculates the moment-strain relationships for each non-root station. These calculations are made in a manner consistent with the source structural calculations, where each blade section is rotated as indicated by Figure 3 and the supporting discussion. Distances to critical fibers, and design moments are all calculated based on this in-plane and out-of-plane orientation.

## **REFERENCES**

1. Tangler, J.L.; Somers, D.M. (1995). “NREL Airfoil Families for HAWTs.” American Wind Energy Association Windpower ‘95 Conference, 26-30 March 1995, Washington, DC. Washington, DC: American Wind Energy Association; pp. 117-128.
2. Eppler, R.; Somers, D.M. (1980). *A Computer Program for the Design and Analysis of Low-Speed Airfoils*. NASA TM-80210. National Aeronautics and Space Administration.
3. Selig, M.S.; Tangler, J.L. (1994). “A Multipoint Inverse Design Method for Horizontal Axis Wind Turbines.” American Wind Energy Association Windpower ‘94 Conference, 10-13 May 1994, Minneapolis, Minnesota. NREL/CP-500-16336. Golden, CO: National Renewable Energy Laboratory.
4. Mandell, J.F.; Samborsky, D.D. (1997). “DOE/MSU Composite Material Fatigue Database: Test Methods, Materials and Analysis.” SAND97-3002.

5. International Electrotechnical Commission. (1999). *IEC 61400-1: Wind turbine generator systems – Part 1: Safety Requirements*, 2<sup>nd</sup> Edition. International Standard 1400-1.
6. Germanischer Lloyd (1999) Rules and Regulations IV – Non-Marine Technology, Part 1 – Wind Energy, *Regulation for the Certification of Wind Energy Conversion Systems*.
7. D. Malcolm, from Excel spreadsheet “SNcurvesDM.xls,” June 2000.

## APPENDIX C.2: WINDPACT DESIGN AND ANALYSIS TOOLS

### 1.0 WindPACT Design and Analysis Tools

This work used the WindPACT preliminary design spreadsheets to produce the initial input files for ADAMS structural modeling and post processing of turbine models. These tools were obtained from Global Energy Concepts, Inc. The WindPACT 1.5-MW configuration (Excel file 1.5A08CO1V07) turbine model input was used as the baseline for this concept design study. The effort required that the team use the WindPACT blade design spreadsheets to create a known baseline that could be used as a starting point for the study. Developing a new baseline model was beyond the scope of the study.

Three primary spreadsheets were used in this work:

- BladeDesign1.5A08V07.xls
- InputData1.5A08V07adm.xls
- Loads\_1.5A08V07adm.xls
- DesignEval&cost1.5A08C01V07cAdm.xls.

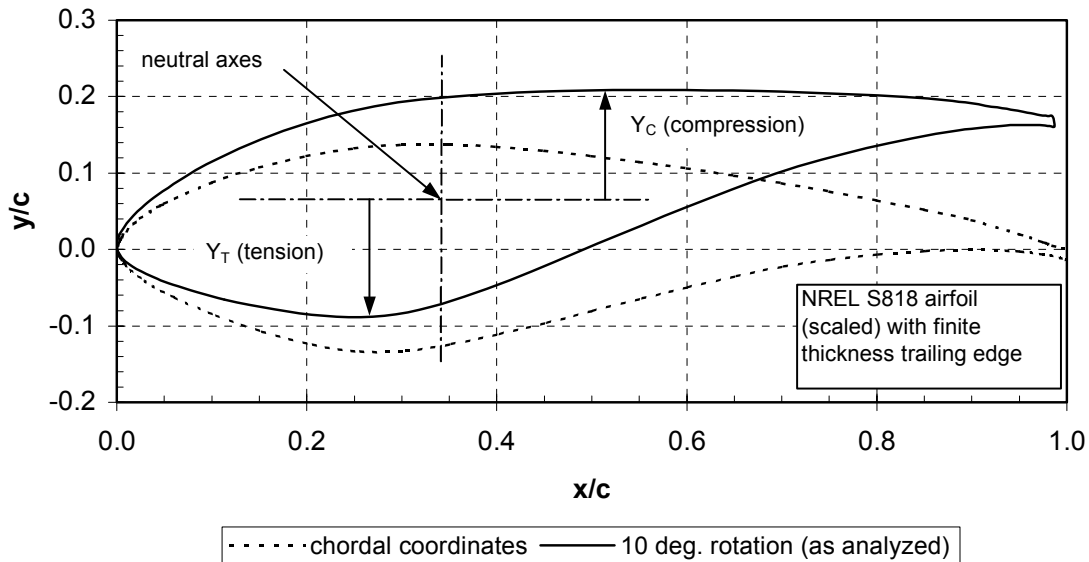
### 1.1 WindPACT Blade Design

The blade design spreadsheet (BladeDesign1.5A08V07.xls) was functional and well-organized, but the internal spreadsheet documentation was sparse, requiring additional information. An associated file, BladeStructuralCalc.doc (BSC), explains the WindPACT blade scaling study spreadsheet modeling approach and is included as Appendix C.1. Blades were designed with spars at 15% and 50% chord. The structural skins (spar caps) spanning between the spars provide most of the blade flap stiffness. The spreadsheets calculated the preliminary structural weight, stiffness, and overall material distribution based on rotor diameter, material properties, and a host of interface constraints. The stress response was of primary concern.

The structural response calculation for fatigue and limit state analysis used strength of materials decoupled ( $M_c/I$ ) for flap and edge loads. The flap load response calculations involved the axis system shown in Figure 1 as described in BSC [1, Figure 1]:

*“An important clarification must be made in the use of “flapwise” and “edgewise” structural properties. Although the terminology of flapwise and edgewise is used throughout this work, the associated properties (calculated and used in the structural scaling model) are more properly termed “out-of-plane” and “in-plane,” respectively. For each airfoil station, a twist angle was assumed, and the structural analyses performed in the rotated orientation. Figure 1 illustrates this approach for the 25% span station, where the analyses were performed with a 10° rotation of the blade section. In determining the structural properties, moments were applied parallel to the original (untwisted) x-y axes. The structural analysis code determines the location of the neutral axis for the rotated section, and calculated bending properties relative to those axes.”*





**Figure 1. Blade analysis methods used with the ADAMS 25% span data**

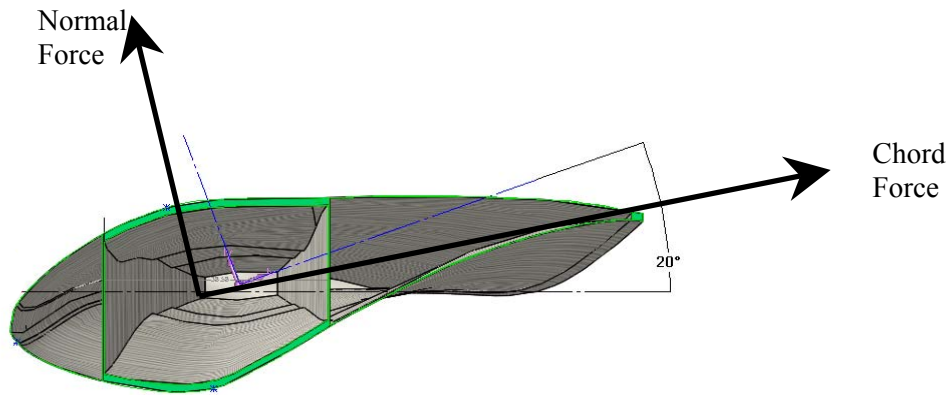
The presumed structural axis shown above and the  $Y_C$  and  $Y_T$  values are used to produce the stress response in the input and design evaluation spreadsheets for blade flap loads. The limitations of this flap load assumption likely were not important to the comparisons done in the WindPACT study. All blade configurations were evaluated with the same method. Since the axes are not aligned with the principle axes, the strength of materials ( $M_c/I$ ) approach is not appropriate. A more accurate approach would require transforming the loads to the principle axes via Equation 1 below [ref 2].

$$\sigma_{\text{bending}} = \frac{M_{\text{edge}} \cdot I_{\text{flat}} - M_{\text{flat}} \cdot I_{\text{flatedge}}}{I_{\text{flat}} \cdot I_{\text{edge}} - I_{\text{flatedge}}^2} \cdot c - \frac{M_{\text{flat}} \cdot I_{\text{edge}} - M_{\text{flatedge}} \cdot I_{\text{flatedge}}}{I_{\text{flat}} \cdot I_{\text{edge}} - I_{\text{flatedge}}^2} \cdot Y \quad \text{Equation 1}$$

The preliminary edgewise blade design was also found to have a conservative structural simplification. A trailing edge spline was added to stiffen the blade and provide a reduction in edgewise stress. The conservative assumption was that the stiffness was increased but the ( $c$ ) in the ( $M_c/I$ ) was not reduced. Thus, the edge stress response was not reduced as much as it should have been reduced. The addition of the spline should be considered in the calculation of the edge stresses. The principal axis will move approximately 1" aft for every 1.2 square inches of spline added at .95c to the 1.5-MW blades at 25%R. That amount of change in  $c$  in the  $M_c/I$  equation should be included in the calculations.

The above simplifications create theoretical analysis problems with the approach used in the follow-on spreadsheets. However, the time constraints on this study and the overall utility of the spreadsheets required the team to use them.

These simplifications allowed the ADAMS data to be used for stress calculations without requiring the transformation to the structural principle axes or additional calculated channels in ADAMS. If we had used Equation 1, it would have required significant testing and comparative studies to establish the baseline. ADAMS models produced the output loads in the geometrically twisted axes shown below in Figure 2. The principle axes were rotated further than the geometric axes. The orientation shown (approximately 20°) is the maximum expected principle axis twist. The angle between the geometric twist and the structural principle axes is less than ten degrees. The skin balsa cores and the blade detail assembly would affect the exact principle axes.



**Figure 2. Blade principle axes and ADAMS loads output axes at 25% span**

In summary, concerns were discovered in the WindPACT study's handling of blade structural properties, fatigue and limit state margins, uncoupled blade stress calculations, and local axis alignment. This LWST study ultimately chose to abide by these problematic procedures to satisfy constraints in time and consistency.

### **1.2 WindPACT Turbine Input Data**

The spreadsheet InputData1.5A08V07adm.xls is well documented and laid out for ease of use. Much of a turbine can be defined with this sheet's Main page tab. This spreadsheet allows the designer an efficient way to provide the ADAMS modeler turbine characteristics. The BladeDesign spreadsheet tabular data provide the blade inputs. The transfer is a simple cut and paste into this sheet. In addition to the blade data, tabs are provided for tower data and drive train data.

This LWST study concentrated on the controller and blade geometry and did not attempt modifications to the other turbine systems or components sections. However, ADAMS simulations were run with a rigid tower simplification.

Only one model was transmitted to the LWST team for ADAMS modeling based on this sheet: a modified baseline with changes noted in bold in Table 1. The planned changes to the  $EI_{flap}$  and  $EI_{edge}$  were small and were not included in the model. However, as noted above, the structural response issues are still unresolved.

**Table 1. Blade Input Data and Suggested Corrections, File InputData1.5A08V07adm.xls.**

	Chord	Section	Spar Cap	Gen. Axis Loc.	CG Loc.	EA Loc.	Weight	EI <sub>Flap</sub>
Station	(m)	(t/c)	(% t)	(y/c)	(y/c)	(y/c)	(kg/m)	(N-m <sup>2</sup> )
5%	1.925	1.00	NA	0.500	0.500	0.500	1447.61	7.68E+09
7%	1.890	1.00	NA	0.500	0.500	0.500	180.33	1.17E+09
25%	2.800	0.30	4.05	0.340	0.409	<b>0.360</b>	<b>184.01</b>	2.74E+08
50%	2.147	0.24	6.72	0.310	0.386	<b>0.348</b>	<b>135.98</b>	7.52E+07
75%	1.494	0.21	6.20	0.280	0.403	<b>0.365</b>	<b>61.03</b>	1.15E+07
100%	0.906	0.16	0.00	0.250	0.493	0.358	11.35	2.31E+05
	EI <sub>Edge</sub>	EA	GJ	I <sub>z</sub>	Twist	Airfoil Filename	EA Twist	
Station	(N-m <sup>2</sup> )	(N)	(N-m <sup>2</sup> )	([kg/m]-m <sup>2</sup> )	(deg)		(deg)	
5%	7.68E+09	1.72E+10	2.66E+09	1292.09	10.5	cylinder	0	
7%	1.17E+09	2.64E+09	4.09E+08	160.96	10.5	cylinder	0	
25%	7.04E+08	<b>2.37E+09</b>	<b>8.396E+07</b>	85.33	10.5	s818_2703.dat	20	
50%	2.56E+08	<b>1.81E+09</b>	<b>2.530E+07</b>	30.02	2.5	"	5	
75%	6.58E+07	<b>7.99E+08</b>	<b>4.344E+06</b>	7.00	0.0	s825_2103.dat	0.3	
100%	7.87E+06	1.18E+08	1.79E+05	0.77	-0.6	s826_2103.dat	-0.6	

### 1.3 WindPACT Loads Data from ADAMS

After performing the full set of ADAMS runs using the model information developed above, the resulting loads data are compiled in the spreadsheet Loads\_1.5A08V07C01adm.xls. This spreadsheet organizes all fatigue and extreme load results and generates statistical summaries. These results feed directly into the analysis and cost tool described below.

### 1.4 WindPACT Structural Analysis with Design Iteration

The DesignEval&Cost1.5A08C01V07cAdm.xls (DEC) spreadsheet is an extensive tool that incorporates and processes a large amount of turbine loads data. There are five main sections in the sheet: input parameters, fatigue data and analysis, limit state data and analysis, margin of safety factoring, and cost models.

- The input parameters are contained in the following tabs:
  - Turbine Input data: "Inputs"
  - Materials data: "Materials"
- The fatigue data are contained in the following tabs:
  - Torque PDFs binned by level at five wind speeds: "Trq8, Trq12, Trq16, Trq20, Trq24"
  - Other load rainflow cycles at five wind speeds: "8ms, 12ms, 16ms, 20ms, 24ms"
  - Summation of fatigue data and wind class selector hours at wind speed: "Rnfsum"
  - Analysis files: "Blade Eval, Hub Eval, Shaft Eval, Grbx Eval, Bedpl Eval, Tower Eval"
  - Summary of extreme and fatigue loads; load case: "LoadSummary"
- The limit state data are contained in the following tabs:
  - Turbine Limit State data IEC Gust cases and Critical loads from all runs: "Ult Lds"
  - Analysis files: "Blade Eval, Hub Eval, Shaft Eval, Grbx Eval, Bedpl Eval, Tower Eval"
  - Summary of extreme and fatigue loads; load case: "LoadSummary"
- The margins of safety factoring data are contained in the following tabs:

- Turbine loads factoring all runs and margins of safety: “EvaluationResults”
5. The cost data files are contained in the following tabs:
    - Turbine cost model components: “cost”
    - Turbine cost-of-energy results: “COE,” “COE2” (presentation)

### ***Procedure for Applying the DEC Tables***

This section outlines a procedure to check and utilize the DEC tool with a new turbine configuration.

1. Start by loading ADAMS data into the last five sheets (to the right). They are torque probability distribution functions (pdfs) for gearbox design at each wind speed bin copied from the loads file.
2. Load ADAMS data into the next five sheets (moving left). They are the rainflow counted load spectra for each wind speed bin copied from the loads file.
3. Check the next sheet to the left, "rnfsun," which integrates the fatigue loads for one year.
4. Load ADAMS data into "Ult Lds," which reproduces the statistics from the loads file.
5. Next check the component evaluation files (i.e., "blade eval"). They calculate fatigue lives, incorporating load factors, material factors, stress concentration factors, local geometry, and material properties. Update the “scale factors” and “partial load factors.”
6. Now jump to the "Inputs" sheet at the left end. Copy in the turbine description from the input data file used to build the ADAMS model.
7. Moving right, check the “Materials” properties and update as necessary.
8. Then check "Evaluation Results" for adequate fatigue and limit state margins in the boxes to the right. Fine-tune the input component geometries as needed by adjusting the input modifiers to the left (columns E and G).
9. Check the "Cost" sheet and adjust several input parameters at the top. This sheet combines geometry information with \$/lbm rates to derive component costs.
10. Check the “COE” sheet and update the AEP from Prop to run the final calculation; “COE2” is the same thing in a presentation format.

The "Evaluation Results" tab is an important design summary page, showing fatigue and limit state margins in the boxes to the right and allowing factoring or fine-tuning of the input parameters to the left. Zero margin means failure at the 20-year life; positive is good, negative is early failure. Because GEC did not extrapolate the fatigue spectra to high-load, low-cycle values, they required some extra margins, as noted in the boxes to the right of the sheet's margin results. The GEC system ignores the mainshaft margins, allowing them to fail in preference to matching known industry sizing.

The "Cost" tab combines turbine geometry info with cost per weight (\$/lbm) rates to derive the component costs, but the blades require some special attention to input a weight estimate based on the blade design spreadsheet. Finally, the “COE” tab includes the annual energy production (AEP) from the AEP spreadsheet and runs the final cost-of-energy calculation. The DOE results are duplicated in the “COE2” tab for plotting the presentation format.

## **2.0 Analysis Methodology**

The analysis methodology for this project involves establishing a baseline, developing a new design with controlled alterations to the baseline, and comparing the resulting figures of merit for the two cases. The analysis procedure follows:

1. Define baseline
2. Design and tune controller on baseline; optimize for loads reduction
3. Import controller into ADAMS
4. Test different baseline configurations (corrected GJ, flexible tower)

5. Quantify loads reductions
6. Design extended rotor
7. Re-tune controller for extended rotor
8. Run ADAMS and check new loads
9. Fine-tune components to meet design margins
10. Generate Cp-TSR curve and AEP for extended rotor
11. Compare resulting costs, AEP, and COE.

## **2.1 Baseline Turbine**

The baseline turbine analysis involved checking the WindPACT turbine input files (BladeDesign1.5A08V07.xls, InputData1.5A08V07adm.xls), duplicating the loads files from the ADAMS model (Loads\_15A08V07C01\_GEC\_FlexibleTower\_PID.xls), and checking the analysis and energy production in the DEC (DesignEval&cost1.5A08C01V07adm.xls) and AEP files (Perf&Torque1.5A08.xls). This task proved to be much more difficult than expected, as it took numerous iterations over 13 months to correctly identify and obtain a complete set of the correct baseline files.

**Versions:** The WindPACT blade studies completed by the GEC team in June 2002 included a fast-evolving baseline with at least seven main versions and more sub-versions, plus some discrepancy in the final report on the accepted version number. The project started with a version 03 baseline provided by GEC and referred to in the main body of their final report. Subsequent version 07 files provided by Windward Engineering caused some confusion before receiving assurances that 07 is the correct and final version. Later work on the analysis procedures revealed discrepancies between the design margins and the margin requirements, leading to the further discovery of a properly tuned version 07c.

**Improvements:** The team also identified several improvements in the baseline structural property and axis calculations as detailed in Section 3.4. This work raised a problematic question: Do the improvements impact baseline loads and design enough to justify shifting to a revised LWST baseline, or is the impact modest enough to justify holding to the GEC baseline? This dilemma created substantial unexpected work because we deemed it prudent to test the effect of using the corrections before making a choice. Although these changes allowed the blade weight to drop 15%, they were not utilized so that the project could stay consistent with the original WindPACT method.

Finally armed with a full understanding of the WindPACT baseline turbine and some desired improvements, the project moved ahead in the Preliminary Design phase, examining the impact of several baseline alternatives, as detailed below.

## **2.2 Preliminary Design**

The preliminary design involved developing an advanced AOA-based controller, augmenting the baseline turbine with the AOA controller, and then running a new set of structural loads and quantifying the difference with the baseline loads. The AOA controller was developed using a SymDyn model of the WindPACT Baseline turbine, confirming the model by comparison with the baseline ADAMS model results, and then Modifying SymDyn's interface with AeroDyn to make the blade AOA array available as a control input in the Simulink program.

Early efforts in the AOA controller design did not include modeling of the tower motion. As a result, runs with the tower elastic degrees of freedom activated were unstable. Subsequent work retuned the controller to handle tower motion with stability; however, full tower flexibility will require further development of the controller model.

The baseline turbine issues led us to evaluate several different configurations for load impacts before and after replacing PID control with AOA control. They included:

1. Original GEC baseline model with flexible tower
2. Original GEC baseline model with rigid tower
3. Revised baseline model with corrected blade properties and flexible tower
4. Revised baseline model with corrected blade properties and rigid tower.

Substantial review of the load data resulting from this matrix of eight design options revealed minimal differences between the original and revised baselines. However, the differences between rigid and flexible tower results were more subtle, requiring completion of the design/analysis process. Therefore, we chose to stick with the original baseline and focus on those two configurations for evaluating the AOA controller benefits over a traditional PID controller:

1. Original GEC baseline model with flexible tower
2. Original GEC baseline model with rigid tower.

Further tuning of the AOA controller brought the flexible tower results in line with those of the rigid tower, thereby validating the use of the original GEC baseline with flexible tower for the rest of the project:

1. Final baseline: Original GEC baseline model with flexible tower.

The selection of the GEC baseline over the modified version was not without difficulty, given the incorrect blade torsional stiffness values noted in previous reports [ie: refs 1, 2]. However, the impact of corrected values on the load results was minimal in the cases examined. This result, combined with a desire to minimize confusion when presenting results to the broader wind turbine design community, created a compelling case to adhere to the original baseline with its flaws. This choice may hold further risks, however, as longer blades are explored with lower blade torsional frequencies.

The WindPACT baseline followed these general design requirements:

- three blades
- upwind
- full-span variable-pitch control
- rigid hub
- blade first flapwise natural frequency between 1.5 and 3.5 per revolution
- blade first edgewise natural frequency greater than 1.5 times flapwise natural frequency
- rotor solidity between 2% and 5%
- variable-speed operation with maximum power coefficient = 0.50
- maximum tip speed  $\leq 85$  m/s
- air density =  $1.225 \text{ kg/m}^3$
- turbine hub height = 1.3 times rotor diameter
- annual mean wind speed at 10-m height = 5.8 m/s
- Rayleigh distribution of wind speed
- vertical wind shear power exponent = 0.143
- rated wind speed = 1.5 times annual average at hub height
- cut-out wind speed = 3.5 times annual average at hub height

- dynamically soft-soft tower (natural frequency between 0.5 and 0.75 per revolution)
- yaw rate less than 1 degree per second.

### **2.3 Final Design**

Rotating blade moments are proportional to radius cubed, so one can conversely take the reduction in moment (achieved through the AOA controller) and use its inverse cube to determine the increase in radius to produce the original moment. For the parked rotor, a square function would apply. Therefore the Final Design analysis procedure is as follows:

1. Select baseline
2. Run all the loads with the advanced controller applied to that baseline
3. Quantify the loads reductions
4. Determine rotor extension limits: inverse cubed root of operating loads reductions and inverse square root of parked loads reductions
5. Design a longer blade to these limits
6. Run the loads with the longer blade
7. Fine-tune the component sizing to meet the margin requirements
8. Estimate the new blade costs
9. Estimate the new blade performance
10. Tally the new COE figures of merit.

Step 7 is open to interpretation, as it tries to duplicate margins from a baseline in which many are much higher than needed, and the margin requirements themselves are very coarsely defined. Full 3-D blade scaling was not used because the baseline section strengths were considered appropriate for loads brought back up to the baseline envelope.

### **3.0 Assumptions and Limitations**

A list of assumptions and limitations is presented here, with more detailed discussions below:

- A reduced set of IEC load cases includes all that are readily modeled, but omits fault conditions.
- Fatigue distribution: the analysis procedure used large design margins instead of extrapolating the fatigue curves.
- A form of engineering judgment was introduced through “scale factors” hidden in the component stress tables; they can be used to apply stress concentration factors and other derating considerations.
- Full 3-D stress states were not used in the component stress analyses; simple beam theory is used instead for a single loading direction for each analysis.
- Loads were not included for 25%, 50%, and 75% span locations of Blades 2 and 3 in the WindPACT approach; yet these blades’ loads could dominate under certain conditions.
- Negative margins were accepted for the main shaft to maintain a design similar to known field dimensions.
- Several blade structural property calculation errors were discovered.
- Torque and gravity loads were summed for trailing edge compression, but torque minus gravity would be the correct application unless the rotor spins backwards.
- Inaccurate ‘c’ values were used in the strain-moment equations.
- Blade fatigue strain limits and S-N curves were extra conservative, as explained below.
- The handling of the trailing edge spline was awkward and probably unnecessary.

In reviewing the WindPACT baseline blade structural properties, we found several corrections that we chose not to implement to stay consistent with an established design set. However, any future work with this baseline should consider these corrections.

- stiffness values are more sensitive to chord-wise element size than expected
- elastic axis: corrected locations are 9.4% higher at span locations 25%, 50%, and 75%
- unit weight: corrected values are 2.3% lower at span locations 25%, 50%, and 75%
- stiffness EA: corrected values are 3.8% lower at span locations 25%, 50%, and 75%
- stiffness GJ: corrected values are 3.85 times (285%) higher at spans 25%, 50%, and 75%
- stiffness Ixx: minor change
- stiffness Iyy: minor change
- local coordinates: corrected coordinates should leave rotor plane and rotate with blade chord.

The baseline design for the 1.5-MW turbine developed under the WindPACT project was used as the baseline design for this contract. The structural properties of the blade were recalculated using the blade cross-section dimensions and material properties presented in the GEC Excel spreadsheets. The purpose of this exercise was to confirm the GEC blade properties using the codes that would also be used for the modified blade employing the proposed AOA pitch control concept. All blade mass, inertia, and stiffness properties, with the exception of torsion, were confirmed. The newly calculated torsional stiffness (GJ) is compared to the original GEC value in Figure 3. The new values for GJ are considerably greater than those of GEC over the central portion of the blade. The calculations were reviewed carefully, and the differences cannot be readily explained. It is not clear from the GEC spreadsheets and reports whether the contribution of the shear web to torsional stiffness was included. This may account for the discrepancy. The area moments of inertia ( $I_p$ ) showed much better agreement in Figure 4.

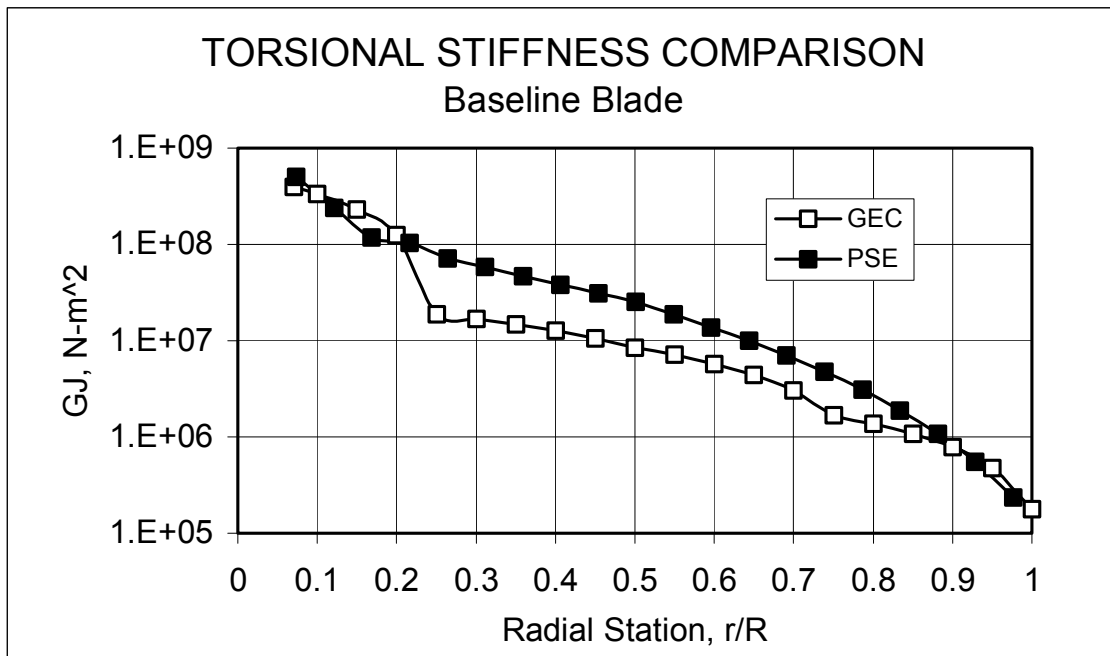
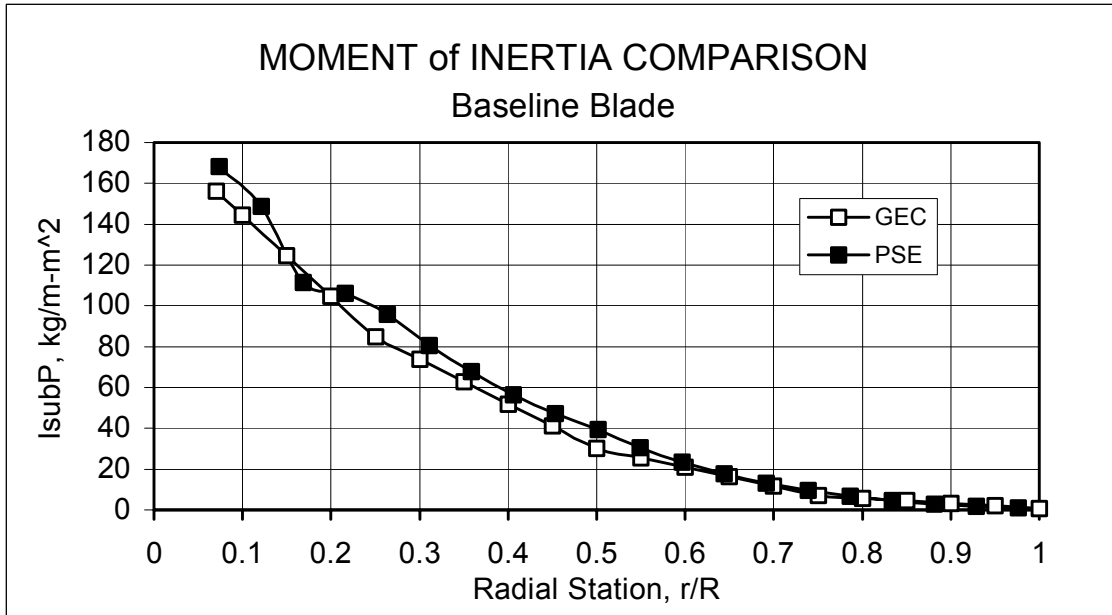


Figure 3. Torsional stiffness





**Figure 4. Area moment of inertia**

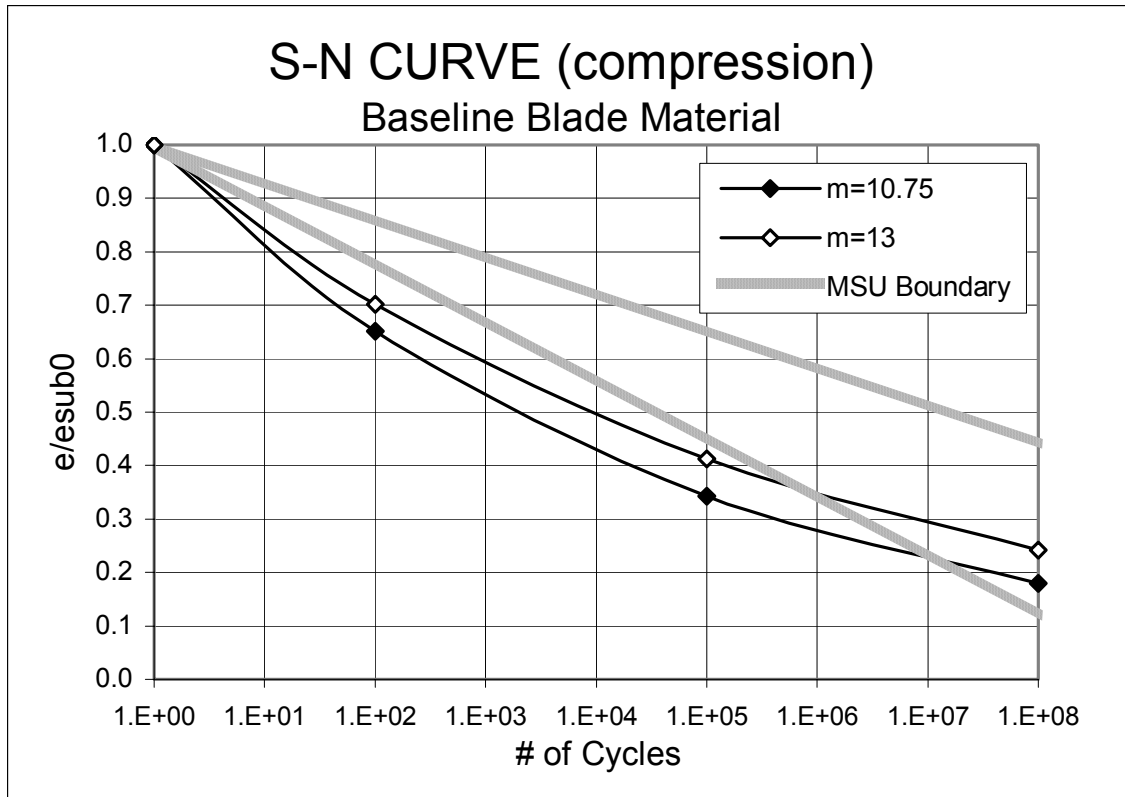
The natural frequencies of the first five modes at a rotor speed of 20.5 rpm are presented in Table 2.

**Table 2. First five blade modes at 20.5 rpm**

First flatwise	3.20p
First edgewise	6.48p
Second flatwise	8.80p
Third flatwise	22.77p
First torsion	40.13p

The GEC blade design spreadsheet indicated an insufficient edgewise margin at the 25% radial station. This would indicate that additional trailing edge spline material would be necessary to reduce the strain in this area. However, it appears that certain assumptions made in the modeling of the blade edgewise structure were overly conservative. Three specific areas are considered:

1. The edgewise limit fatigue strain level used by GEC was  $960 \mu\epsilon$  (blade properties sheet, cell C12). This was based on  $N$  cycles of  $1.23E+8$  (5000 hrs/yr for 20 yrs) and fatigue data from the Montana State University (MSU) report [ref 5]. The allowable static strain in compression is  $10,500 \mu\epsilon$ , and downgrading that by a materials factor of 1.933 gives a value of  $5431 \mu\epsilon$ . Using a simplified S-N formula [ $e/e_0 = N^{(1/m)}$ ] produces a value of 960 for an 'm' of 10.75. This value of m appears to be too conservative when examining the MSU data. Figure 5 shows the MSU range of data [ref 3, Fig 19] for compressive fatigue tests. GEC used the curve labeled "m=10.75," which is barely within the MSU boundaries. The 'm' value suggested by D. Griffin in his report [ref 1] was 13. This is closer to the average at the high cycle end. If a value of 13 were used, then the allowable strain would move up to  $1296 \mu\epsilon$ , a 35% increase. This would move the trailing edge strain from unacceptable to acceptable and void the need for a trailing edge spline.



**Figure 5. S-N diagram for blade material**

2. In determining the edgewise strain at the extreme fibers (i.e., at the leading and trailing edges), an accurate location of the neutral axis (NA) is vital. It appears that at the 0.25R station ( $c=2.79m$ ), GEC used an NA of 30% $c$ . This would be closer to the elastic axis (EA). New calculations place the NA at 42% $c$ , which would reduce the trailing edge fiber distance from 1.97m to 1.68m and would reduce strain by 15%. To assume the NA is as far forward as 30% would say that all the edge stiffness was from the spar, where in fact 39% of the total is from the outer skin.
3. GEC shows the critical design condition to be that where the sum of the torque at 1.5 MW plus the 1P gravity load produce a strain on the trailing edge that would not satisfy the 20-year life spec. The GEC spreadsheet shows this torque exceeding the blade capacity. It is an error to assume that the trailing edge would see the maximum compressive load under these conditions because the only time these two loads are additive is when the blade is moving down, which puts the leading edge in compression. Under these conditions, the trailing edge would be in tension, which has more than double the strain allowable. When the blade is moving up, the trailing edge compressive load would be the *difference* in the gravity and torque loads. The max strain on the leading edge would be lower than that on the t.e. because it is closer to the NA, for an estimated reduction of 17%.

In summary, three areas were identified where the GEC calculations produced conservative results: (1) the material fatigue properties (by 35%), (2) the location of the neutral axis (by 18%), and (3) the edgewise loading. If all of these changes are valid and were incorporated, there would be a significant opportunity to reduce the blade weight. This could be pursued up to the point at which some other operating condition drives the design. A previous concern was raised in which the WindPACT

baseline has a fatigue failure flag at the 25% span blade station for trailing edge compression. Further investigation showed that a compensating trailing edge spline was specified outside the design loop.

In reviewing the WindPACT design procedures, we noted that they do not extrapolate the fatigue distributions and they do not utilize full three-dimensional stress states in the component analyses. WindPACT instead applies substantial “scale factors” and design margins to compensate for these (and possibly other) coarse analysis assumptions. The LWT team would like to improve these analyses but acknowledges once again that this is out of the purview of the current contract, and it would cause further difficulty in presenting results to the wider wind community. The WindPACT methodology requires substantial fatigue life margins in lieu of extrapolating the fatigue spectra into the high-load/low-cycle region. After testing a few components, we estimated that damage accumulation could approximately double in these regions. Further analysis would be very helpful on this issue.

### **REFERENCES**

1. Griffin, D. (2002). *Scaling Of Composite Wind Turbine Blades for the WindPACT Rotor Design Study*, Kamzin Technologies (now Global Energy Concepts).
2. Malcolm, DJ; Hansen, C. (2002). *WindPACT Turbine Rotor Design Study: Final Report*, NREL Contract YAT-0-30213-01.
3. Mandell, J.F.; Samborsky, D.D. (1997). *DOE/MSU Composite Material Fatigue Database: Test Methods, Materials and Analysis*, SAND97-3002.



## **APPENDIX D.1: PITCH DRIVE SYSTEMS**



### **Capabilities of Industry Standard Pitch Drives for Independent Blade Pitch Systems**

**Client: Advanced Energy Systems**

**Date: March 4, 2003**

**Version: 01**

**Written by: Terrance Meyer**

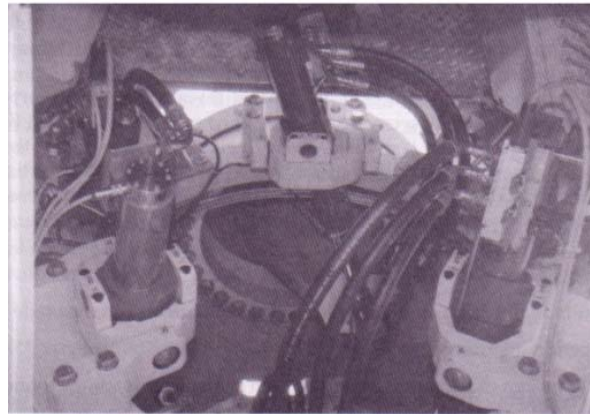
**Checked by: John Vanden Bosche**

## ***Introduction***

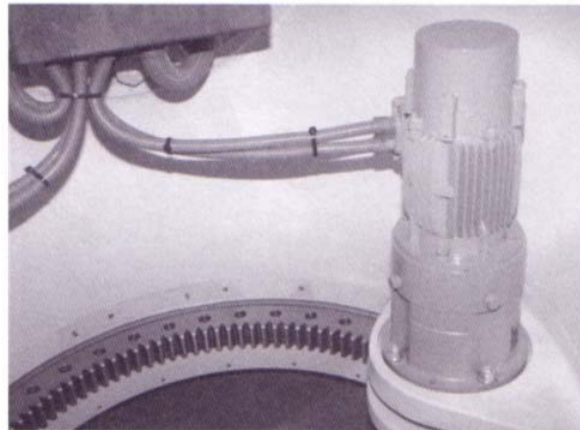
Chinook Wind was tasked with investigating pitch system capabilities and costs for commercial wind turbines in the 1.5-MW size range that utilize independent blade pitch control. The information gathered will be used to assess the feasibility of the final controller design developed in the NREL Low Wind Speed Turbine project. Key elements of the pitch drive capability to be assessed were the speed, acceleration, and torque capabilities, as well as the cost, weight, and physical size of pitch drives.

## ***Findings***

Specifications for pitch drives on commercial wind turbines are proprietary, so we have developed an independent specification from publicly available data on pitch system capabilities and topology. Some information was available through conversations with Lee Fingersh, personal experience, field observations, and publicly available information from the manufacturers. Some information about blade pitch drives was available in reference [1]. Photos of a hydraulic system and an electromechanical system from reference [1] are shown in Figures 1 and 2 respectively.



**Figure 1. Hydraulic blade pitch system**



**Figure 2. Electromechanical pitch system**

Pitch drives on 1.5-MW turbines built today can achieve between 7 and 12 degrees per second maximum pitch rate. The design driver for the maximum pitch rate is the shut down speed, or the speed it takes to get the blade to 90 degrees in an extreme gust situation with an accompanying loss of load to avoid damaging loads. It should be noted that the battery bank is designed not for continuous operation during a power outage but rather to move the blades out of the wind once the power goes out. As a margin of safety, the battery bank is typically sized to perform on the order of 10 full pitches of the blade.

The most common topology is a low-inertia version of either a DC servo motor or a variable speed AC motor with a continuous stall torque on the order of 25 Nm attached to a gear box with a gear ratio on the

order of 100:1. A pinion is driven by the gear box and mates with a bull gear around the circumference of the blade pitch bearing with a ratio on the order of 10:1 for a total gear reduction on the order of 1000:1. All of these ratios are approximate. Other systems exist, such as hydraulic systems or different electro-mechanical systems, but little information is available, and those systems seem to be less common. It is suggested that we assume the above described topology until an actual unit or better information is available.

Specifications for low inertia AC Servo Motors from two manufacturers (Baldor and Allen Bradley) are given in Tables 1 and 2. From this we can get an idea of cost and weight penalties for increasing pitch system capabilities and select a plausible motor for any pitch system we design into the control system.

**Table 1. Baldor Motor Specifications**

**Baldor**

<u>CONT. STALL TORQUE</u>	<u>MAX TORQUE</u>	<u>RPM</u>	<u>BUS VOLTS</u>	<u>CONT. STALL AMPS</u>	<u>INERTIA</u>	<u>CATALOG NUMBER</u>	<u>LIST PRICE</u>	<u>WEIGHT Pounds</u>
lb-in	lb-in				lb-in-s <sup>2</sup>			
117.71	4x	4000	300	20.3	(9.4836) 0.0082	<a href="#">BSM90A-</a>	\$ 2,450	48
117.71	4x	2000	300	8.92	(9.2578) 0.0082	<a href="#">BSM90N-</a>	\$ 1,540	43
117.71	4x	1200	300	5.77	(9.2578) 0.0082	<a href="#">BSM90N-</a>	\$ 1,890	25
117.71	4x	4000	300	19.6	(9.2578) 0.012	<a href="#">BSM90N-</a>	\$ 1,890	43
123.9	4x	2000	300	10.26	(13.548) 0.012	<a href="#">BSM100N-</a>	\$ 1,660	62
123.9	4x	1200	300	6.41	(13.548) 0.013	<a href="#">BSM100N-</a>	\$ 1,660	62
125.67	4x	2000	300	10.3	(14.677) 0.0196	<a href="#">BSM100A-</a>	\$ 2,350	47
203.55	4x	2000	300	16.87	(22.1284) 0.0196	<a href="#">BSM100N-</a>	\$ 1,980	62
203.55	4x	1200	300	10.85	(22.1284) 0.0295	<a href="#">BSM100N-</a>	\$ 1,950	62
300.02	4x	2000	300	22.9	(33.3055) 0.0273	<a href="#">BSM100A-</a>	\$ 3,125	74
300.9	4x	2000	300	22.91	(30.8217) 0.0273	<a href="#">BSM100N-</a>	\$ 2,280	62
300.9	4x	1200	300	16.03	(30.8217) 0.0349	<a href="#">BSM100N-</a>	\$ 2,349	62
354	4x	1200	300	18.09	(39.4021) 0.0349	<a href="#">BSM100N-</a>	\$ 2,805	62
366.39	4x	2000	300	30.3	(39.4021)	<a href="#">BSM100A-</a>	\$ 3,050	79



**Table 2. Allen Bradley Motor Specifications**

**Allan Bradley**

<u>CONT. STALL TORQUE</u> lb-in	<u>MAX STALL TORQUE</u> lb-in	<u>RPM</u>	<u>BUS VOLTS</u>	<u>CONT. STALL AMPS</u>	<u>INERTIA</u> lb-in-s <sup>2</sup>	<u>CATALOG NUMBER</u>	<u>LIST PRICE</u>	<u>WEIGHT</u> Pounds
172	430	4000	230	41.5	0.013	MPL-A540K	\$ 2,025	33
247	500	3000	230	41.5	0.016	MPL-A560F	\$ 2,300	44.5
172	430	4000	460	20.5	0.013	MPL-B540K	\$ 2,025	33
247	500	3000	460	20.5	0.016	MPL-B560F	\$ 2,300	44.5
325	640	3000	460	32.1	0.0354	MPL-B560F	\$ 2,600	59
425	895	3000	460	38.5	0.051	MPL-B560F	\$ 3,300	77

**References**

1. Burton, T., Sharpe, D., Jenkins, N., Bossanyi, E. Wind Energy Handbook, John Wiley & Sons, West Sussex England, 2001, pp. 351-356 and 505-506.

## **APPENDIX D.2: BACKUP POWER STORAGE FOR PITCH DRIVE SYSTEMS**



### **Feasibility and Market Acceptability of Backup Power System for High Wind, Power Outage Blade Survival**

**Client: Advanced Energy Systems**

**Date: September 17, 2003**

**Version: 02**

**Written by: Terrance Meyer**

**Checked by: John Vanden Bosche**

## **Introduction**

Chinook Wind has been tasked with investigating the backup power requirements for running the LWT pitch drive continuously through a high wind speed and power outage event. It may come as a result of the pitch control investigation that storm survival loads are the limiting load case and the only way to reduce the weight or increase the size of the turbine via better control, will be by reducing loads when the blades are in a feathered position during storm conditions. Part of the IEC 61400-1, Ed. 2 standard requires an assumption that utility power is not available during 50-year storm conditions.

## **Findings**

The first step in estimating the amount of energy storage necessary is to quantify the power draw of the pitch drive system and set an amount of time that the system is to be self-sufficient.

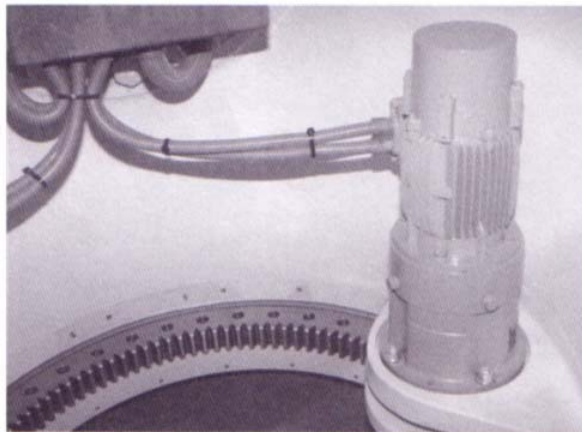
We decided on a 1-day or 24-hour period as the arbitrary period of self-sufficiency as there is no applicable standard, but any other duration of energy storage can be scaled linearly to the results. A 7-day maximum duration power outage is called for in the above mentioned standard, but it is unlikely that extreme winds would be present for the entire duration of this 7-day outage.

It should be noted that in a power outage situation, there will be no power to the yaw drive, and as such the blades must be able to pitch a full 360 degrees.

Three independent methods are used to obtain an estimate of power draw by the pitch system. Because no single method is considered authoritative, by reporting all three the confidence in the results is increased.

For the first method, a likely pitch drive motor is selected (Allan Bradley MPL-B560F with a continuous stall torque of 27.9 Nm, listed in the table below). This motor has a continuous stall current rating of 20.5 amps at 460 V. A duty cycle of 10% is selected, considering that the blade is not continuously pitching and that when it is pitching, the maximum stall current is not likely to be needed.

$$460V * 20.5A * 0.1 = 943 \text{ watts continuous power per blade}$$



**Figure 1. Electromechanical pitch system**

**Table 1. Allen Bradley Motor Specifications**

**Allan Bradley**

<u>CONT. STALL TORQUE</u>	<u>MAX STALL TORQUE</u>	<u>RPM</u>	<u>BUS VOLTS</u>	<u>CONT. STALL AMPS</u>	<u>INERTIA</u>	<u>CATALOG NUMBER</u>	<u>LIST PRICE</u>	<u>WEIGHT Pounds</u>
lb-in	lb-in				lb-in-s <sup>2</sup>			
172	430	4000	230	41.5	0.013	MPL-A540K	\$ 2,025	33
247	500	3000	230	41.5	0.016	MPL-A560F	\$ 2,300	44.5
172	430	4000	460	20.5	0.013	MPL-B540K	\$ 2,025	33
247	500	3000	460	20.5	0.016	MPL-B560F	\$ 2,300	44.5
325	640	3000	460	32.1	0.0354	MPL-B560F	\$ 2,600	59
425	895	3000	460	38.5	0.051	MPL-B560F	\$ 3,300	77

The second method employed the experimental experience of Lee Fingersh with the National Renewable Energy Laboratory’s (NREL’s) Controls Advanced Research Turbine (CART). The CART did not pitch during a feathered storm event, but during active pitching, 500 watts per blade were consumed continuously. Fingersh determined that this would be a reasonable estimate for the feathered condition used in this analysis. Because aerodynamic loads are the main driver for pitch system power draw, Fingersh recommended scaling with swept area of the blades. It is possible that the power consumption may increase with the blade’s mass moment of inertia about the pitch axis, which scales approximately with rotor diameter cubed. Although other scaling rules have been conceived for this analysis, we decided to use Fingersh’s recommendation to scale power consumption with rotor swept area. The CART is 600 kW and has a diameter of 43.28 meters; the study baseline turbine has a rating of 1500 kW and is 70 meters in diameter. In a final design, other than our baseline, if the rotor diameter significantly increases, it may be appropriate (considering these assumptions) to increase the backup power reserve.

$$500W * \frac{70^2}{43.28^2} = 1308 \text{ watts continuous power per blade}$$

The third method employs results from Reference 2 and observation of a battery bank sited in a 1.5-MW wind turbine rotor designed for 10 pitches of the blade 90 degrees out of the wind. On page 64 of Reference 2, a table outlines the assumed equivalent loads and motion of a blade. The random and variable motion of a pitching blade was reduced to a single equivalent rate of oscillation and a single equivalent amplitude of those oscillations. For a wind speed of 50 MPH, an oscillation rate of 28.5 oscillations per minute with an amplitude of 1.076 degrees was used, giving a blade travel of **61.4 degrees per minute**, which is used to approximate the storm condition. The battery used for all three blades in the observed turbine is a standard-size marine battery, which is nominally 100 amp hours at 12 volts or **1.2 kWh**. Ten pitches at 90 degrees each gives **900 degrees**.

$$\frac{1.2kWh}{3blades} * 61.4 \text{ deg./min} * 60 \text{ min/hr} = 1637 \text{ watts continuous per blade}$$

Of the three methods, the third has the greatest power draw, but it may include a margin of safety on the battery, which the other two methods do not.

To size the battery bank, the three methods for determining battery draw are averaged and rounded to **1.3kW**. A buss voltage of **460V** is assumed (this will drive the Allen Bradley motor mentioned above).

$$\frac{1300W / blade * 3blades}{460V} = 8.5 Amps / blade * 24h * 1.5safety = 300 Amphours @ 460V$$

### **Backup Power Options**

A 460V-battery bank at 300 Amp-hours of industrial 2-volt cells was quoted by Dyno Battery [3] at a cost of \$7,000; an additional \$4000 should be added for non-scaling power electronics and structure costs, for a total of **\$11,000**. The weight of the battery pack is around **10,000 pounds**, indicating a tower base installation. This number should give a ballpark of the economic trade-off of having the blade able to pitch continuously during a storm condition with loss of utility power. If the battery bank is located at the base of the turbine, slip rings are needed to bring power into the rotor to the pitching motors. Slip rings can be unreliable; thus redundant slip rings or some parked positive contact device is indicated. It is Chinook Wind's opinion that the added cost and complexity of this solution would have difficulty finding acceptance in the marketplace.

An alternative would be a much smaller battery bank in the hub; perhaps 10 marine-type batteries providing a 2-hour period of independence, costing ~\$600, and weighing 600 pounds. This is unsatisfactory if the survival of the blades or other turbine equipment is dependent on the reduced loads achieved by the advanced control. It is conceivable that a power saving mode of control could be employed during a power outage situation in which, instead of minimizing loads in general, the controller would only provide enough action to avoid damaging loads. If the new design resulting from this study relies on blade pitching to avoid damage during a loss of utility power combined with a 50-year storm wind, the alternative of a small battery pack in the hub is likely to be the only one that could be easily accepted in the market because it has the least added cost and added vulnerability. This option requires a pitching algorithm for controlling the blades during storm survival that utilizes minimal pitch motion to significantly reduce power consumption by the pitch system.

Another alternative that would allow for a smaller battery bank at the tower base or in-hub (slip rings would still be required) is providing auxiliary generation for battery charging. This generation could take the form of a central fossil fuel generator, which would be activated in a power outage, similar to hospital backup power generators. Such a generator for a 50-MW wind farm (1.3 kW x 3 blades X 33 1.5-MW turbines = 130 kW) has been specified at \$26,000 for a Generac SD 150, 150-kW generator with transfer switch and 24-hour fuel tank. An Uninterruptible Power System (UPS) costing \$8,000 should be added to provide stability during transfer and for added redundancy. At approximately \$1000 per turbine for 33 turbines, the total is \$34,000; if an in-hub battery bank is added at a cost of \$600, the estimated total cost would be around **\$2000 per turbine**, including cabling, redundant slip rings, or any power electronics not included explicitly in this analysis. A fossil fuel generator may add a significant obstacle to permitting of the wind farm, as well as being an added system requiring maintenance and affecting reliability. In spite of these factors, an extra \$2000 is relatively inexpensive compared to \$11,000 for a battery system with other maintenance and reliability issues. Thus, if it is necessary to power the pitch mechanisms for the wind turbine to survive a high-wind-speed power outage event, a wind farm generator and UPS with a redundant in-hub battery bank appears to be the preferred option.

If the consequence of a generator malfunction is that the entire wind farm is destroyed, then it is Chinook Wind's opinion that such an option may not be well accepted in the market. It should be noted that a battery bank in hub will provide 2 hours of operation (more with a power saving algorithm), during which an emergency repair of the generator could be performed.

It is conceived that an additional generation source to the fossil fuel generator mentioned above could be a ~200 kW wind turbine designed to perform at high wind speeds, having shorter blades and more rugged components, with the UPS and or generator providing excitation current in the absence of the grid. This wind turbine at an added cost of approximately \$200,000 or approximately \$6,000 per turbine is more expensive than a central fossil fuel backup generator, although it would add to the capacity of the wind farm, thereby offsetting its cost.

The acceptability of the backup power options considered, as well as the control strategy, are directly tied to the consequences that result from failure. If the loss of a pitch actuator, slip ring, or inverter means that the wind turbine is destroyed, or in the case of a generator that the whole wind farm is destroyed, then that risk, no matter how small, will be unacceptable. However, if the consequence of failure is the loss of a blade or some other component, and if the risk is rather small, then it is more likely that such a system will be acceptable because such risks are currently accepted for commercial wind turbines.

### **Conclusions**

We conclude that it is possible to provide backup power for long-term continuous operation of pitch drives to minimize structural loads during a utility outage and a 50-year storm condition. A generator with UPS and an in-hub battery bank are the best solutions analyzed to provide a 24-hour period of coverage. However, such a backup system involves cost and structural commitments, which should be weighed against the benefits of reduced design margin resulting from relying on such a control strategy. We further conclude that the level of acceptance in the marketplace will be directly tied to the consequences of failure of the backup power and control system.

### **References**

1. Burton, T., Sharpe, D., Jenkins, N., Bossanyi, E. Wind Energy Handbook, John Wiley & Sons, West Sussex England, 2001, pp 351-356 and 505-506.
2. NREL guideline DG03 *Wind Turbine Design Yaw & Pitch Roller Bearing Life*, Rev 12-12-00.
3. John Williams, Personal Conversation, Dyno Battery, Seattle Washington.

# REPORT DOCUMENTATION PAGE

*Form Approved*  
OMB No. 0704-0188

The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing the burden, to Department of Defense, Executive Services and Communications Directorate (0704-0188). Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

**PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ORGANIZATION.**

<b>1. REPORT DATE (DD-MM-YYYY)</b> December 2004		<b>2. REPORT TYPE</b> Subcontract Report		<b>3. DATES COVERED (From - To)</b> 7/30/02 - 7/31/04		
<b>4. TITLE AND SUBTITLE</b> Low Wind Speed Turbine Design Project Conceptual Design Study: Advanced Independent Pitch Control				<b>5a. CONTRACT NUMBER</b> DE-AC36-99-GO10337		
				<b>5b. GRANT NUMBER</b>		
				<b>5c. PROGRAM ELEMENT NUMBER</b>		
<b>6. AUTHOR(S)</b>  T. Olsen, E. Lang, A.C. Hansen, M.C. Cheney, G. Quandt, J. VandenBosche, and T. Meyer				<b>5d. PROJECT NUMBER</b> NREL/SR-500-36755		
				<b>5e. TASK NUMBER</b> WER4.3301		
				<b>5f. WORK UNIT NUMBER</b>		
<b>7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)</b> Advanced Energy Systems, Inc. 1428 South Humboldt Street Denver, CO 80210				<b>8. PERFORMING ORGANIZATION REPORT NUMBER</b> LAM-2-31235-02		
<b>9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)</b> National Renewable Energy Laboratory 1617 Cole Blvd. Golden, CO 80401-3393				<b>10. SPONSOR/MONITOR'S ACRONYM(S)</b> NREL		
				<b>11. SPONSORING/MONITORING AGENCY REPORT NUMBER</b> NREL/SR-500-36755		
<b>12. DISTRIBUTION AVAILABILITY STATEMENT</b> National Technical Information Service U.S. Department of Commerce 5285 Port Royal Road Springfield, VA 22161						
<b>13. SUPPLEMENTARY NOTES</b> NREL Technical Monitor: A.D. Wright						
<b>14. ABSTRACT (Maximum 200 Words)</b> AES conducted a conceptual study of independent pitch control using inflow angle sensors. The control strategy combined input from turbine states (rotor speed, rotor azimuth, each blade pitch), with inflow angle measurements (each blade angle of attack at station 11 of 15) to derive blade pitch demand signals. The controller reduced loads sufficiently to allow a 10% rotor extension and reduce COE by 6.3%.						
<b>15. SUBJECT TERMS</b> wind energy; pitch control; wind turbines						
<b>16. SECURITY CLASSIFICATION OF:</b>			<b>17. LIMITATION OF ABSTRACT</b>  UL	<b>18. NUMBER OF PAGES</b>	<b>19a. NAME OF RESPONSIBLE PERSON</b>	
<b>a. REPORT</b> Unclassified	<b>b. ABSTRACT</b> Unclassified	<b>c. THIS PAGE</b> Unclassified			<b>19b. TELEPHONE NUMBER (Include area code)</b>	

Standard Form 298 (Rev. 8/98)  
Prescribed by ANSI Std. Z39.18