All Theses and Dissertations

2018-08-01

Efficient Incorporation of Fatigue Damage Constraints in Wind Turbine Blade Optimization

Bryce Taylor Ingersoll
Brigham Young University

Efficient Incorporation of Fatigue Damage Constraints

in Wind Turbine Blade Optimization

Bryce Taylor Ingersoll

A thesis submitted to the faculty of
Brigham Young University
in partial fulfillment of the requirements for the degree of

Master of Science

Andrew Ning, Chair
Steven Gorrell
John Salmon

Department of Mechanical Engineering

Brigham Young University

ABSTRACT

Efficient Incorporation of Fatigue Damage Constraints
in Wind Turbine Blade Optimization

Bryce Taylor Ingersoll
Department of Mechanical Engineering, BYU
Master of Science

Improving the wind turbine blade design has a significant effect on the efficiency of the wind turbine. This is a challenging multi-disciplinary optimization problem. During the blade design process, the aerodynamic shapes, sizing of the structural members, and material composition must all be determined and optimized. Some previous blade design methods incorporate the wind turbine's static response with an added safety factor to account for neglected dynamic effects. Others incorporate the dynamic response, but in general is limited to a few design cases. By not fully incorporating the dynamic response of the wind turbine, the final turbine blade design is either too conservative by overemphasizing the dynamic effects or infeasible by failing to adequately account for these effects. In this work, we propose two methods which efficiently incorporate the dynamic response into the optimization routine. The first method involves iteratively calculating damage equivalent fatigue that are fixed during the optimization process. We also demonstrate the training and use of a surrogate model to efficiently estimate the fatigue damage and extreme events in the design process. This surrogate model has been generalized to be used for different rated turbines, and can predict the fatigue damage of a wind turbine with less than 5% error. In general, these alternative, more efficient methods have been shown to be an adequate replacement of the more computationally expensive method of calculating the dynamic response of the turbine within the optimization routine.

ACKNOWLEDGMENTS

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1. BACKGROUND

1.1 Brief History of Wind Turbines

Since ancient times, people have exploited wind resources for their gain [1,2]. Early examples include sailors using wind for ship propulsion and farmers using windmills to grind grain and corn. In the late 19th century, Poul la Cour developed one of the first systems to generate electrical power from a rotating wind mill. Thus, the modern wind turbine was born. The wind turbine was improved by Johannes Jensen and Poul Vinding in 1919, when they designed the Agricco. The blades of this wind turbine could be turned into different positions, and the wind turbine also turned against the wind automatically. In the 1950's, Johannes Jull erected a new and more modern turbine which utilized nearly 60% of the possible effect of the wind.

Despite these advancements, the wind turbine was in direct competition with steam turbines and diesel engines, which were the prominent producers of electricity in the first half of the 20th century. It was not until the oil crisis of the early 1970's created a renewed spark of interest in wind turbines. This crisis encouraged a number of countries such as Denmark, Germany, France, the United Kingdom and the United States to become less dependent on oil imports. Since then, commercial wind turbines have become an increasingly important industry. The wind power capacity of the United States has increased significantly over the last two decades (see Figure 1.1) and has emerged as a sustainable, renewable, and clean source of energy[1].

1.2 Advantages & Disadvantages of Wind Energy

There are several advantages and disadvantages of using wind as an energy resource through wind turbine use [3]. Wind power plants have relatively little impact on the environment compared to traditional power sources such as coal plants. However, wind turbines create a certain amount of

---

[1]http://www.awea.org/Resources/Content
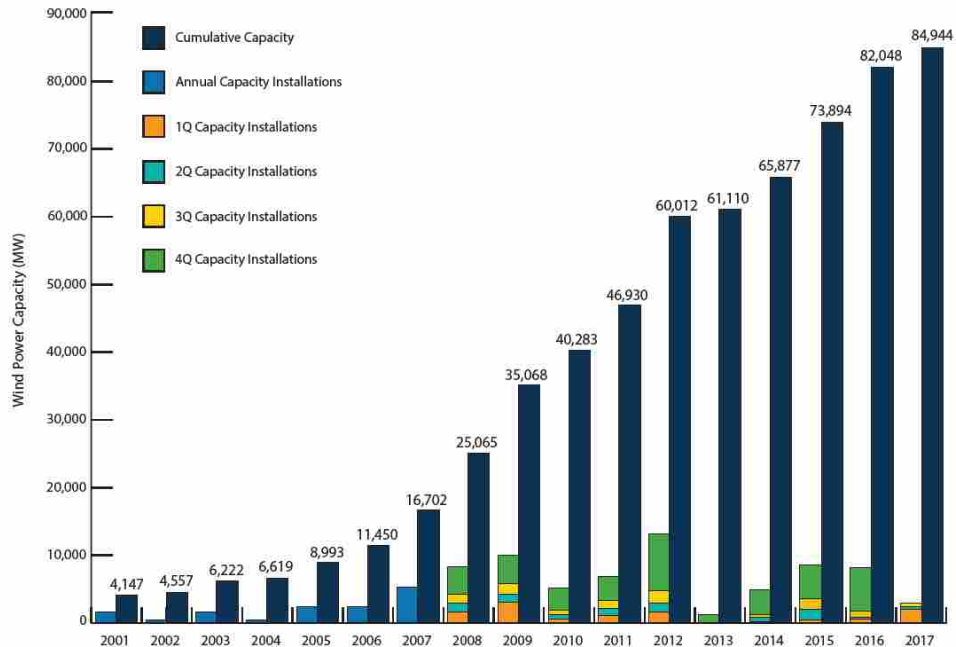
Figure 1.1: The new installation and cumulative wind power capacity in the United States from 2001 to 2015. From the American Wind Energy Association.

noise when they produce electricity. The level of noise is an important factor, especially in densely populated areas. Aesthetically, two-blade turbines create a flickering effect, which can be annoying for those that live nearby. Environmental concerns for wildlife, such as birds and bats being killed by flying into the rotors, are also important. But, for the most part these problems have been resolved or greatly reduced through proper site placement and technological development.

Wind energy, similar to solar and hydroelectric energy, is a renewable resource [4]. In contrast to fossil fuels such as coal and natural gas, wind energy is clean and non-polluting. In addition, when power is generated from wind turbines, no CO_2 emissions are produced.

The cost of wind power continue to decrease. However, wind energy requires a higher initial investment than fossil-fueled generators due to the cost of turbine construction and land. A large focus to decrease wind farm costs is to reduce the price of the turbines themselves. Additional factors, such as land cost and available interest rates, as well as the amount of wind available at a given site, also influence the price of the energy generated. However, wind costs are much more competitive compared with fossil-fueled systems on a "life-cycle" cost basis (counting fuel and operating expenses for the life of the generator) because there is no fuel to purchase and minimal

operating expenses. In addition, wind energy is relatively labor intensive and thus creates many jobs, though this does contribute to an increase in wind's cost of energy.

An issue with wind energy is that it can only be produced when there is sufficient wind. Wind itself cannot be stored and battery storage of wind produced electricity is often inefficient [5]. In addition, sites with sufficient wind capacity are often located in remote locations. Wind resource development also must compete with other uses for the land, which may be more lucrative, such as city and housing development.

1.3 IEC Standards for Wind Turbines

Design requirements for wind turbines have been specified by the International Electrotechnical Commission (IEC) [6]. This worldwide organization seeks standardization in all national electrotechnical committees (IEC National Committees). The IEC's goal is to promote international cooperation in standardization in the electrical and electronic fields.

The publication "International Standard: IEC 61400-1" specifies requirements for onshore wind turbines to ensure the engineering integrity of turbines, as well as to provide an protection against damage from various hazards during the turbine's planned lifetime. These standards cover all major subsystems of wind turbines. Such subsystems include mechanical systems and support structures, control and protection mechanisms, and internal electrical systems. These standards require the use of a structural dynamics model to predict experienced design loads. This dynamic model is used to determine the wind turbine's response over a range of wind speeds and conditions in specified design load cases (DLCs).

The design standards specified by the IEC for onshore wind turbines have been updated and improved over time. The most recent version was the third edition published in 2005, which replaced the second edition published in 1999. Examples of changes include an expanded turbulence model to include an extreme turbulence model and amending certain design load cases. In addition to "IEC 61400-1", the IEC has published "IEC 61400-2" which includes considerations for small turbines, and "IEC 61400-3", which details specifications for offshore wind turbines.

In this work, several specific requirements from these standards are used. The wind conditions specified in a subset of the design load cases suggested in the IEC standards are implemented and used in determining the dynamic loading of the wind turbine designs that are investigated. The

turbulence and turbine classes defined in the IEC standards are also used in the generation of the wind conditions. The suggested simulation time as well as the probability of an extreme event are also used in this study.

1.4 Improving the Efficiency and Power Output of Wind Turbines

Wind energy production has steadily increased worldwide over the last decade. This is due to many factors, one of which is wind's decreasing cost of energy (COE), a metric used to compare the cost of different energy sources. As the cost of wind energy has decreased, wind farms have become a more viable source of renewable energy. There are a number of ways to decrease a wind farm's COE. These include analyzing the topography and wind resources at possible sites, optimizing the layout of wind turbines to reduce wake interference [7], and choosing appropriate wind turbines. Extensive work has been done to improve the power production and efficiency of wind farms by modeling wind farm layout as an optimization problem. This optimization problem seeks to optimize the locations (and in some cases, the orientations) of a group of wind turbines within specific boundaries. A wind rose, a tool that shows how the direction and wind speed are usually distributed at a specific site, is incorporated in the design process. A crucial part of wind farm design is predicting the turbine wakes downstream from a wind turbine [8, 9]. The turbine wake represents an area of slower wind, so wind farm optimization generally tries to minimize the wake interaction between wind turbines.

Numerous works have been done to highlight potential improvements to the wind turbine design. Some research includes determining the optimal number of turbine blades [10], optimization of the wind turbine tower [11], and predicting the most effective generator to use at a specific site [12]. Significant research focused on the wind turbine's blade design and increasing the ability of wind turbines to extract energy from the incoming wind. This is mainly influenced by the design of the wind turbines' blades because the blades' aerodynamic shapes govern the interaction between the incoming wind and the turbine.

An effective blade design optimization method can significantly increase the performance of a wind turbine. During the design process, the material composition, aerodynamic shapes, and sizing of the structural members must all be determined and optimized. This set of design variables define the aerodynamic and structural performance of the wind turbine, which factor

4

into the turbine's annual energy production (AEP) or wind's COE. In addition, the blade design must also satisfy a number of constraints which ensure safe operation in a series of design load cases [6, 13, 14], which cover the various conditions in which the wind turbine may operate.

To ensure that the wind turbine is able to safely operate in conditions specified by design load cases, structural constraints are specified during the design optimization process. These constraints include bounds on the strains experienced by the blade, fatigue, and clearance between the tower and blades. In particular, many works note that the design of many wind turbine components, including the blade, are driven by constraints on the fatigue damage [10, 15–17]. Therefore, it is important to include fatigue damage constraints in the blade design process.

To determine the fatigue damage of a wind turbine, the dynamic loading experienced by the wind turbine should be computed or predicted. As noted by Bottasso et al. [18, 19], determining the dynamic loading for a number of design load cases is computationally expensive. When an extensive set is used, their computational cost dominates and leads to a very computationally costly optimization routine. In their work, they show that even when using a subset of all design load cases, this routine is expensive, where the computational time is on the order of one to two days. The computational expense limits the number of design load cases that can be used.

In contrast to evaluating the dynamic loading of the wind turbine, a number of works constrain the blade turbine design by calculating the experienced stresses and strains at defined maximum static loading conditions [20–23]. Fatigue effects are not directly calculated, but are often accounted for by including a safety factor in the computed stresses and strains to estimate fatigue effects. While this is a cheap alternative to actually determining the fatigue damage, it is difficult to determine whether the safety factor is adequate or too conservative without actually calculating the wind turbine's dynamic loading.

It would potentially be very beneficial to develop a method that determines the fatigue damage and other dynamic effects by evaluating the turbine's dynamic loading within the optimization loop, and is not prohibitively computationally expensive if a large suite of design load cases are used. Such a method would allow for the efficient optimization of a more realistic blade design. The purpose of this work is to present two different approaches that accomplish these goals. As proposed but not demonstrated in our previous work [22, 24], the first approach involves calculating and updating equivalent fatigue damage along the blade by evaluating the dynamic loading of

the wind turbine outside of the optimization loop. This fatigue damage constrains the turbine blade design during its optimization routine. Once the optimization is complete, the fatigue damage of the final design is calculated, and the design is again optimized using the updated equivalent damage. We show that this iterative method converges within four to five iterations, and is much faster than including the evaluation of the dynamic loading within the loop.

The method described in the previous paragraph is an improvement over evaluating the dynamic loading of the wind turbine in the loop. However, there is a steep learning curve to implementing a dynamic evaluation code in a blade optimization framework. The second approach replaces the evaluation of the dynamic loading of the turbine with a surrogate model. Surrogate models have been used previously in wind turbine optimization. One surrogate optimization methodology implements a high-fidelity viscous-inviscid interaction code to evaluate the aerodynamic design [25]. Another approach using a Kriging fit to optimize the site-specific aerodynamic design [26]. Our approach is unique in that the surrogate model is used to predict the fatigue damage and extreme loads for a variety of design load cases and is implemented in a blade optimization routine.

This surrogate model is trained using a variety of differently rated horizontal axis wind turbines. This surrogate model, once trained, is cheap to evaluate and estimates the equivalent fatigue damage. Once the initial work to create the surrogate model is done, there is no need to evaluate the dynamic loading within the optimization routine. Using a surrogate model will allow for quicker implementation of fatigue damage constraints in various conceptual studies in the blade design process.

The goal of this research is to efficiently incorporate the wind turbine's dynamic response during the blade optimization process. Doing this will result in a more accurate representation of the wind's effects on the blades and more precise constraints on the blades' structural and aerodynamic parameters. This will lead to better final blade designs, meaning an increase in wind turbine aerodynamic and structural performance.

CHAPTER 2.   FORMULATION OF THE TURBINE BLADE DESIGN PROBLEM

To demonstrate our proposed optimization methods and routines, we define the blade optimization problem. We then describe the integration of the dynamic response into the optimization routine. Finally, we describe necessary calculations used to convert the dynamic response of the turbine into optimization constraints.

The blade optimization is done in RotorSE, an engineering model for analysis and optimization of horizontal axis wind turbines [27]. RotorSE is written within the OpenMDAO framework [28], an open-source platform that supports multidisciplinary analysis and optimization. The blade aerodynamic analysis, CCBlade[1] used in RotorSE is based on blade element momentum (BEM) theory [2]. The BEM equations were solved using a method that guarantees convergence and efficiently converges to the solution [29]. Within the RotorSE framework, a beam finite element, pBEAM[2], was used for the blade structural analysis. The inertial and stiffness properties were determined in PreComp[3], a modified classic laminate theory.

2.1   Formulation of the Turbine Blade Design Problem

The objective of the blade optimization routine is minimize the wind turbine's cost of energy (COE), which is defined as [30]:

$$COE = \frac{OPEX(1-T) + FR(TCC+BOS)}{AEP} \qquad (2.1)$$

where $OPEX$ is the overall project expenditures, T is the tax deduction rate on $OPEX$, FR is the financing rate, TCC is the turbine capital costs, and BOS is total balance of station costs. Specifically, a value of 0.4 is used for T and 0.095 for FR. Improving the blade design can influence the

---

[1]https://nwtc.nrel.gov/CCBlade
[2]https://nwtc.nrel.gov/pBEAM
[3]https://nwtc.nrel.gov/PreComp
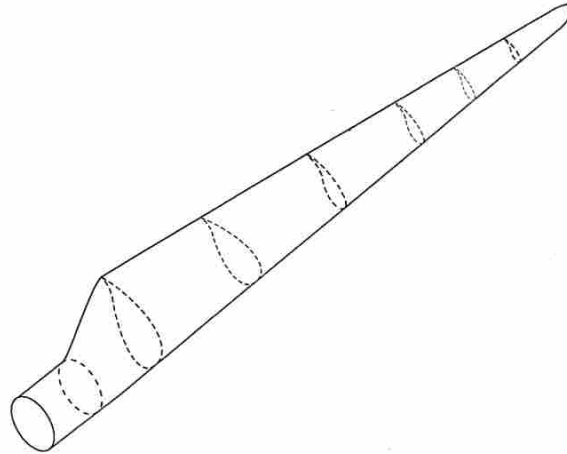
Figure 2.1: An illustration of the airfoil cross sections of a typical wind turbine blade.

COE by both raising the AEP by increasing the aerodynamic performance, as well as decreasing the needed material and lowering the turbine capital costs.

A wind turbine blade design is characterized by a number of parameters and values. These parameters define the design and can be manipulated to potentially improve the power production and efficiency of a given wind turbine. A number of works have focused on improving the blade design using a variety of different blade parameters, all with the goal of improving the wind turbine's performance.

An important component of the blade design is the airfoil selection. As seen in Figure 2.1, a given blade's cross sections are defined by selected airfoils at certain points along the length of the blade. Between these points, the cross section is determined by blending the intermediary airfoils. The chosen airfoils have a direct effect on the energy produced, and a number of studies have focused on how to improve the airfoil selection [31, 32].

A number of works have focused on the optimal material choice for blade design. Due to the material having a direct effect on the blade's mass and fatigue limits, work has been done to accurately calculate the fatigue of various materials used in the blade design [33, 34]. In addition, when composite materials are used, research has been done to determine the best composite layup structure [35, 36].

Aeroelastic and structural parameters are also often manipulated when improving the performance of a particular blade design. Examples of common aeroelastic parameters used in blade

design optimization are the blade chord (the distance between the leading and trailing edge of the blade) and the blade twist [37, 38]. Structural parameters may include the spar cap thickness and trailing edge thickness. These parameters vary non-discretely along the length of the blade. In general, parameters values are chosen at distinct points along the length the blade, and a fit or spline is used to define a parameter's value at any point along the length of the blade.

As done by Ning et al. [22, 24], we optimize the coupled aero-structural design of the wind turbine blade. Four groups of design variables are included: aeroelastic design variables of chord and twist distribution control points, and structural design variables of spar cap and trailing edge thickness. An Akima spline was fit to the control points of each design variable, so that the design variables can be defined at any position along the length of the blade. As described by Ning et al., the rotor geometry should be defined with as few terms as possible to allow for efficiency in the optimization routine, while still allowing for flexibility in describing the rotor geometry. Thus, four control points are used for each aeroelastic distribution and five control points for each structural distribution. An example of the control points used to define the chord distribution is shown in Figure 2.2.

A number of constraints must be met to ensure the usability of a given blade design. Structural constraints limit the forces and bending moments experienced by the wind turbine to prevent buckling, fatigue, or maximum load failure. The blades' maximum tip deflection is limited to prevent a collision between the tower and blades. Natural frequencies and thrust are also constrained to prevent failure.

In this work, we use the same constraint groups as used in Ning et al. [22, 24]. We constrain the strain and buckling of the spar cap and trailing edge, as well as constrain the fatigue damage effects at the same locations. The flapwise and edgewise frequencies are also constrained, as well as the rotor thrust at rated power. Finally, we constrain the maximum tip deflection to avoid collision between a rotor blade and the tower.

2.2   Problem Framework

As mentioned before, the optimization routine is done in RotorSE, an engineering model for horizontal axis wind turbines [27]. Brief descriptions of the sub-packages RotorSE incorporates are included below. CCBlade, a blade element momentum method, calculates the aerodynamic
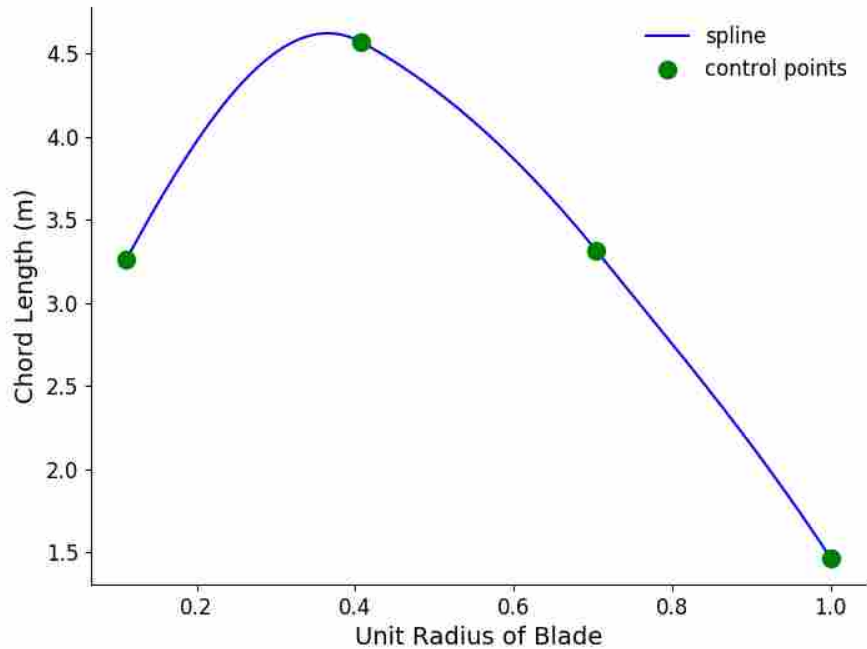
Figure 2.2: An example of the chord distribution. The distribution is defined by an Akima spline, which is determined by four control points.

performance. Structural analysis is performed using the following three sub-packages: PreComp, a classical laminate cross-section method, pBEAM, a beam finite element method, and CurveFEM [39], a beam finite element method for curved blades.

CCBlade is a blade element momentum method. It is used to calculate the aerodynamic performance of a wind turbine. Its implemented method has a guaranteed convergence rate and is therefore very robust [29]. Analytic gradients can be provided for the distributed loads, thrust, torque, and power with respect to design variables of interest.

PreComp (**pre**-processor used for computing **comp**osite blade properties), calculates structural data for composite blades such as stiffness properties, inertial properties, and offsets of the blade shear center, tension center, and center of mass with respect to the blade pitch axis. Access to these properties allow for rapid evaluation of alternate composite layouts and their effects on blade properties and material strains.

pBeam (**p**olynomial **b**eam **e**lement **a**nalysis **m**odule) is a finite element code for beam-like structures [40]. Section properties defined in pBeam can be described as polynomials of any

order between nodes. Thus, high fidelity can be achieved in describing variation in structural properties with accuracy while using fewer elements. In addition, the use of polynomials also allows for higher accuracy because integrals and derivatives are evaluated analytically rather than through numerical methods, such as finite difference. pBEAM can estimate a number of structural properties, such as axial stress and strain, axial buckling loads, structural mass, deflections in all degrees of freedom, and coupled natural frequencies.

CurveFEM is a beam finite element method for curved blades. It was developed to evaluate the aeroelastic behavior of swept blades, since they offer the potential to increase energy capture and lower fatigue limits. Compared to the Adams processor used in NREL's FAST code, this methodology would allow for lower cost analysis and faster computation times.

CHAPTER 3.   DETERMINATION OF THE TURBINE'S DYNAMIC RESPONSE

To determine the dynamic response of the wind turbine, we use FAST [41], an aeroelastic computer-aided-engineering tool, to perform the dynamic simulation. To connect the wind turbine design defined in RotorSE, we used AeroelasticSE[1], a python-wrapper for the FAST executable. AeroelasticSE formats the wind turbine design specifications into the proper input files for use in FAST.

Within FAST, aerodynamics models, hydrodynamics models for offshore structures, control and electrical system (servo) dynamics models, and structural (elastic) dynamics models are joined to perform a aero-hydro-servo-elastic simulation in the time domain. FAST is based on advanced engineering models–derived from fundamental laws, but with appropriate simplifications and assumptions, and supplemented where applicable with computational solutions and test data. Sub-packages incorporated in FAST are briefly described below. To compile and run FAST, several subroutines must be included, and they are described briefly below.

InflowWind is a module for processing wind-inflow that has been coupled into the FAST engineering tool to enable aero-elastic simulation of horizontal-axis wind turbines[2].

AeroDyn is a tool that calculates the aerodynamic loads on wind turbine blade elements based on velocities and positions provided by dynamics analysis routines and simulated wind inputs[3]. This module is an element-level wind-turbine aerodynamics analysis routine. It requires information on the status of a wind turbine from the dynamics analysis routine and returns the aerodynamic loads for each blade element to the dynamics routines.

---

[1]https://nwtc.nrel.gov/AeroelasticSE
[2]https://nwtc.nrel.gov/InflowWind
[3]https://nwtc.nrel.gov/AeroDyn

3.1 Incorporation of FAST routine

We used the design specifications of the NREL 5MW reference wind turbine [42] for blade optimization. We also used the WindPACT baseline turbine models [43] which have been designed to help with studies involving how aerodynamic loading scales with size. For each baseline reference turbine, a FAST executable was built from source. Each executable was unique in that it specified the turbine's torque and pitch control routines. The NREL 5MW control routines were developed by Jonkman et al. [44], while the WindPACT reference turbines used a simple pitch control routine.

In the RotorSE framework, structural constraints are determined at specific radial positions, with an additional position at the root. For this study, we specified seventeen positions to calculate structural constraints along the length of the turbine blade. The point locations are shown in table 3.1. Within the AeroelasticSE routine, these positions are defined as the FAST parameter RNodes. FAST requires the user to specify at which locations loading data will be recorded. This is done by specifying the FAST parameter BldGagNd, which assigns the locations of the virtual strain-gages in FAST. For example, if we set BldGagNd as [1,2,3], loading data will available at the first, second and third RNodes values.

The FAST input NBlGages is the number of strain gage locations specified by BldGagNd. A limitation of FAST v7.0 is that the maximum value of NBlGages is seven. Thus, bending and loading information can only be recorded at seven distinct points along the length of the blade, not including the blade root. If we desire to calculate the dynamic loading at each radial position specified in RotorSE, FAST must be run three times.

To reduce the computational cost, we specified BldGagNd to be a subset of radial positions that would result in a small error between the interpolated and actual values at the seventeen radial positions used in RotorSE. We set BldGagNd to [1,3,5,7,9,12,17], and the corresponding blade fractions are given in Table 3.1. An example of the interpolation can be seen in Figure 3.1. The root mean square difference between the calculated and interpolated data was ~ 0.44 %.

To evaluate the dynamic loading using FAST, we must specify the wind conditions during the simulation. These conditions were determined by a number of design load cases (DLCs), which have been specified by the IEC Design Standard for onshore wind turbines [6]. The purpose of these DLCs is to evaluate the turbine's response in a variety of wind conditions with respect

Table 3.1: Locations of aeroelastic data
in RotorSE framework.

Blade Fraction	Point
0.0222	1
0.067	2
0.111	3
0.167 - 0.833, linearly spaced	4-14
0.889	15
0.933	16
0.978	17
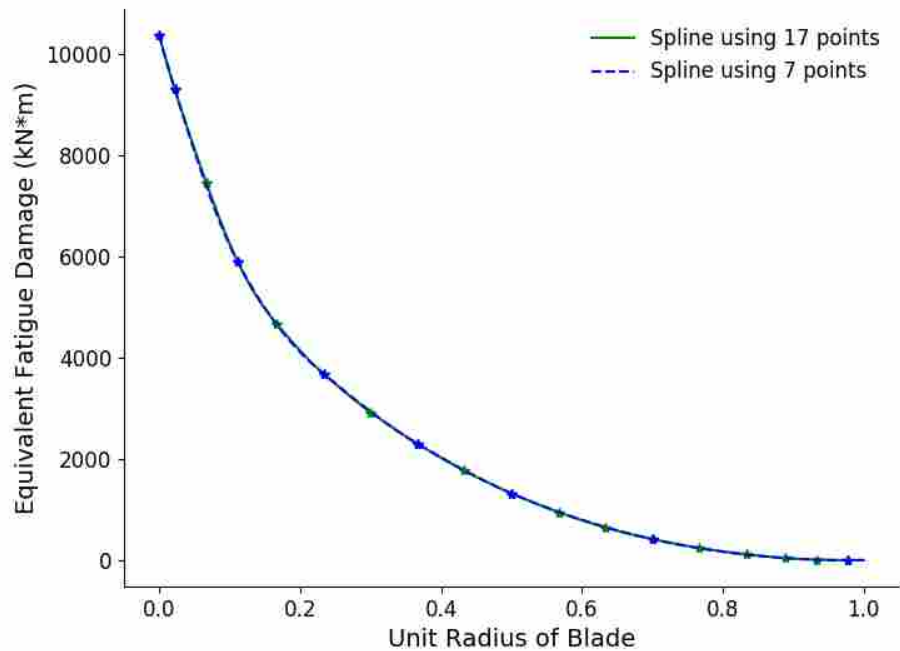
Figure 3.1: A comparison of calculated flapwise damage equivalent moments where virtual strain gages are placed at every radial position used in RotorSE (green) compared to being placed at a subset of all radial positions (blue).

to fatigue and ultimate strength analysis. To demonstrate this methodology, a selection of design load cases shown in tables 3.2 and 3.4 were used in this study. These design load cases includes power production while the wind turbine experiences normal and extreme turbulence, extreme gusts and wind shear, as well as parked configurations with extreme yaw misalignment. These

Table 3.2: Design load cases using non-turbulent wind files.

DLC	Description	Num. of Wind Files
DLC 1.4 (ECD)	Evaluation of ultimate loads during regular power production with an extreme coherence gust with change in wind direction.	6
DLC 1.5 (EWS)	Evaluation of ultimate loads evaluation during regular power production with the presence of extreme wind shear.	4
DLC 6.1 (EWM50)	Evaluation of ultimate loads while in a parked configuration during a 50-year extreme steady wind event.	1
DLC 6.3 (EWM01)	Evaluation of ultimate loads while in a parked configuration during a 1-year extreme steady wind event with extreme yaw misalignment.	1

load cases were chosen because they have been show to most likely drive and effect the wind turbine design [45].

A given design load case is simulated using a set of wind input files, where FAST simulates the dynamic loading for each wind file. Non-turbulent wind files were generated using IECWind[4]. A short description and number of wind files generated for each non-turbulent DLC is in Table 3.2. Depending on the particular design load case, either a single wind input file is sufficient to fully simulate the intended environmental conditions (such as DLC 6.1 and 6.3), while others require multiple wind input files.

The turbulent wind files were generated using TurbSim[5]. Specifications used in TurbSim are listed in Table 3.3. For each average wind speed, six 11 minute wind files were created. To expedite the wind input file generation process, a python wrapper for TurbSim was created to quickly create and save the generated files. Table 3.4 gives a brief description of the design load cases simulated using the wind input files generated by TurbSim.

In the main input file for FAST, we specified the length of the simulation Tmax, integration time step DT, and the start of the recorded data TStart, and that it would be a time marching solution (using the FAST input AnalMode). We used different simulation times for the turbulent and non-turbulent wind input files; specifically the turbulent simulations were 640 seconds, and the non-turbulent simulations were 100 seconds. Also within the main input file, we specified what

---

[4]http://wind.nrel.gov/designcodes/preprocessors/iecwind
[5]https://nwtc.nrel.gov/TurbSim

Table 3.3: Specifications used when generating wind files.

Parameter	Value
V_{in}	3 m/s
V_{out}	25 m/s
V_{rated}	11.4 m/s
Average wind speed	5 - 23 m/s
Turbulence Model	Kaimal
IEC Class	I
Turbulence Class	B

Table 3.4: Design load cases using turbulent wind files.

DLC	Description	Num. of Wind Files
DLC 1.2 (NTM)	Evaluation of fatigue damage during normal power production in normal turbulence	60
DLC 1.3 (ETM)	Evaluation of ultimate loads during normal power production in extreme turbulence	60

outputs from the simulation we wanted to record. We recorded the root bending moments in the edgewise and flapwise directions, as well as the bending moments at the positions specified using the FAST parameter BldGagNd. These specifications are listed in table 3.5.

3.2   Fatigue Calculation

To determine the blade fatigue, we first calculated damage equivalent moments (DEM) from the simulation data. DEMs represent a constant amplitude load that can be used in the de-

Table 3.5: Specifications used in FAST simulations.

FAST Parameter	Value
Tmax	640.0 / 100.0 s
DT	0.0125 s
TStart	20.0 s
AnalMode	1
OutList	RootMxb1
	RootMyb1
	Spn[1-7]MLxb1
	Spn[1-7]MLyb1

termination of fatigue damage. The cycle ranges and peaks of each simulation are recorded in a rainflow routine. The formula for a damage equivalent moment (DEM) is derived in MLife[6]:

$$DEM = \left(\frac{\sum_i [n_i(L_i)^m]}{n^{ST}} \right)^m \tag{3.1}$$

where L_i is the load range for the ith cycle, m is the Wöhler exponent, n^{ST} is the equivalent number of counts for the time series, and n_i is the specific cycle value (0.5 for a half count and 1.0 for a full count). The recorded data from the rainflow routine is used in equation 3.1 to calculate the DEMs. This is done for the edgewise damage equivalent moments DEM_x and flapwise damage equivalent moments DEM_y. When the load range L does not vary cycle to cycle, the damage equivalent moment calculation reduces to

$$DEM = L \tag{3.2}$$

and was preliminarily used as a check for the calculated equivalent damage in non-turbulent design load cases (see Figure 3.2). As can be seen from this figure, the range cycle varies little over the length of the simulation, and we can estimate the damage equivalent moment from a typical cycle range. However, this check isn't valid when a turbulent design load case is specified (see Figure 3.3). As seen from this figure, there is a wide variety of cycle ranges and lengths. Thus, a rainflow routine is needed to determine these values so that a DEM can be calculated using equation 3.1.

Once the edgewise and flapwise damage equivalent moments are determined, they are transferred to the elastic center and principal axes of the blade section. We then calculate the edgewise strain ε_U and flapwise strain ε_L, where

$$\varepsilon_U = -\left(\frac{DEM_1}{EI_{11}} y_U - \frac{DEM_2}{EI_{22}} x_U + \frac{F_z}{EA} \right) \tag{3.3}$$

$$\varepsilon_L = -\left(\frac{DEM_1}{EI_{22}} y_L - \frac{DEM_2}{EI_{22}} x_L + \frac{F_z}{EA} \right) \tag{3.4}$$

The stiffnesses EI, cross sectional areas A, distances x and y and Young's modulus E are determined in the RotorSE framework. The number of cycles to failure in the edgewise direction
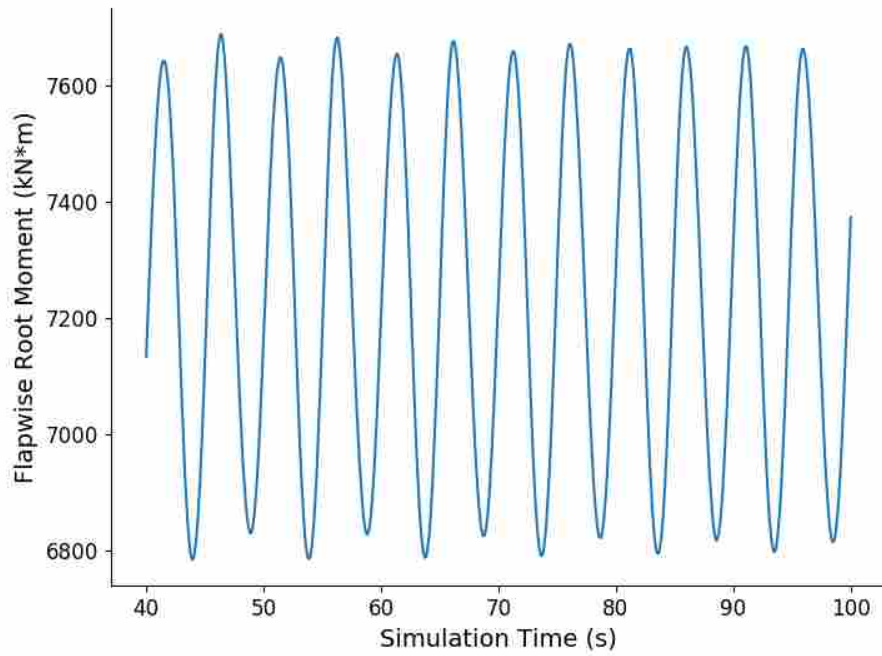
---

[6]https://nwtc.nrel.gov/MLife

Figure 3.2: The root flapwise bending moment (RootMyb1) for a non-turbulent input file.
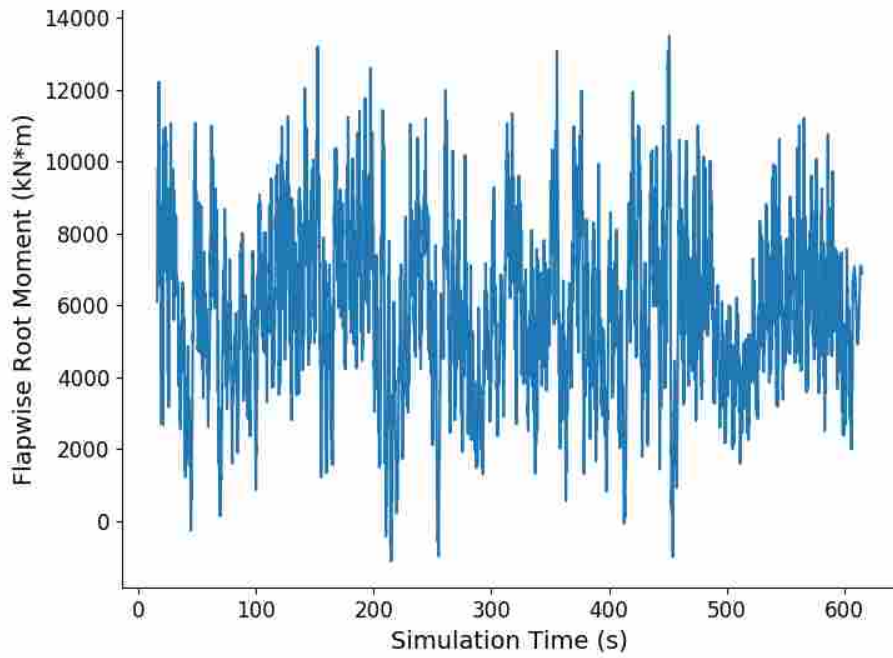

Figure 3.3: The root flapwise bending moment (RootMyb1) for a turbulent input file.

18

N_{fU} and number of cycles to failure in the flapwise direction N_{fL} are defined using the following expressions:

$$N_{fU} = \left(\frac{\varepsilon_{max}}{\eta \varepsilon_U}\right)^m \tag{3.5}$$

$$N_{fL} = \left(\frac{\varepsilon_{max}}{\eta \varepsilon_L}\right)^m \tag{3.6}$$

where ε_{max} is the ultimate strain, η is the material safety factor (a value of 1.3 was used), and m is the Wöhler exponent (typically 10 for composite materials that are glass-reinforced [46]). Once the number of cycles to failure has been calculated, we determine the equivalent damage in the edgewise and flapwise direction, or

$$\text{damage}_U = \frac{N}{N_{fU}} \tag{3.7}$$

$$\text{damage}_L = \frac{N}{N_{fL}} \tag{3.8}$$

where N is the planned lifetime number of cycles. For this study, we used a design life of 20 years. A calculated damage greater than 1 corresponds to failure due to fatigue.

3.3 Extreme Moment Extrapolation

From the simulated dynamic response, the peaks can be determined and visualized as a distribution (see the green histogram in Figure 3.4). As defined by the IEC Design Standard for onshore wind turbines [6], an extreme moment should be extrapolated from the calculated peak distribution and then used to constrain the blade design. We fit a Gaussian distribution to the loading data (see the fit in Figure 3.4) and use this to estimate the extrapolated loads, as done in other works [47]. The probability density P for a Gaussian distribution is defined as:

$$P(F) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(F-\mu)^2}{2\sigma^2}} \tag{3.9}$$

where μ is the mean, σ is the standard deviation, and F is the extreme load. For a given 10 minute simulation, the IEC Design Standard showed that the extreme event F should have a probability P of 3.8×10^{-7}. Equation 3.9 was used to estimate the extreme load case. We did this for moments at the blade root, as well as along the length of the blade in both the edgewise and flapwise direction.
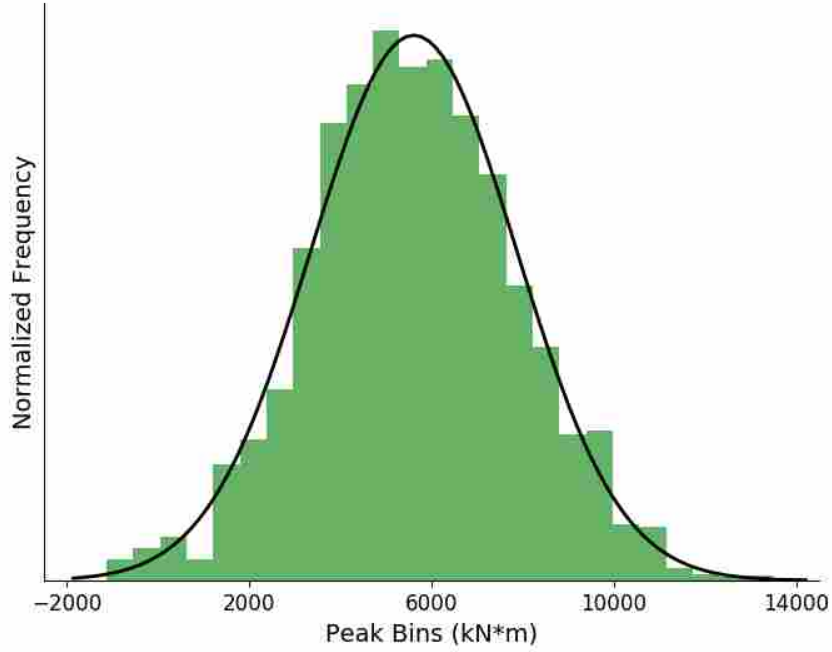
Figure 3.4: The bending moment distribution (green) at the blade root in the flapwise direction for a turbulent wind file. A Gaussian distribution (black) is fit to the data.

The extreme moments were incorporated as constraints in the blade design process by using them to calculate the maximum strain in the upper and lower sections of the spar cap and trailing edge, where

$$\varepsilon_U = -\left(\frac{M_{e,x}}{EI_{11}} y_U - \frac{M_{e,y}}{EI_{22}} x_U + \frac{F_z}{EA} \right) \tag{3.10}$$

$$\varepsilon_L = -\left(\frac{M_{e,x}}{EI_{22}} y_L - \frac{M_{e,y}}{EI_{22}} x_L + \frac{F_z}{EA} \right) \tag{3.11}$$

The calculated strain is then constrained by the ultimate strain (a value of 1.0e-2 was used for the spar cap, and 0.5e-2 for the trailing edge).

CHAPTER 4.   FIXED DYNAMIC LOADING OPTIMIZATION APPROACH

As described in the previous section, the routine to determine the dynamic loading of the wind turbine using FAST provides useful data that can be incorporated into the wind turbine blade optimization process. The calculated damage equivalent moments and extreme extrapolated events from this routine can be incorporated into design constraints that drive the design. In addition, because of the higher-fidelity of the method used to calculate these constraints, the accuracy of the overall design process is improved and the final design is more realistic.

Unfortunately, the computational cost to evaluate the wind turbine's dynamic loading using FAST within the optimization routine is prohibitively expensive. The sequential evaluation of the blade design using all the prescribed input wind files (132 in total) varies depending on the machine being used, but can be on the order of two hours. Within a single optimization, this routine is run several hundred or even thousands of times. The computation time is lessened when we parallelize this calculation, but a full optimization routine may take several days or weeks to complete, or potentially months if done sequentially.

The prohibitively expensive cost led us to explore methods where the blade optimization process could benefit from the routine to determine the dynamic loading of the wind turbine using FAST while not necessarily running the routine within the optimization process. As proposed by Ning et al. [22, 24], the first proposed approach involves calculating and updating the damage equivalent and extreme moments along the blade outside the optimization routine. We freeze the damage equivalent and extreme moments during the initial turbine blade optimization. We theorize that while the calculated damage and stresses will have a first order change with the blade design, the calculated loading will change less significantly, and it will suffice to update the loading only periodically. Once the optimization is complete, the damage equivalent and extreme moments of the final design are calculated, and the design is again optimized using the updated damage
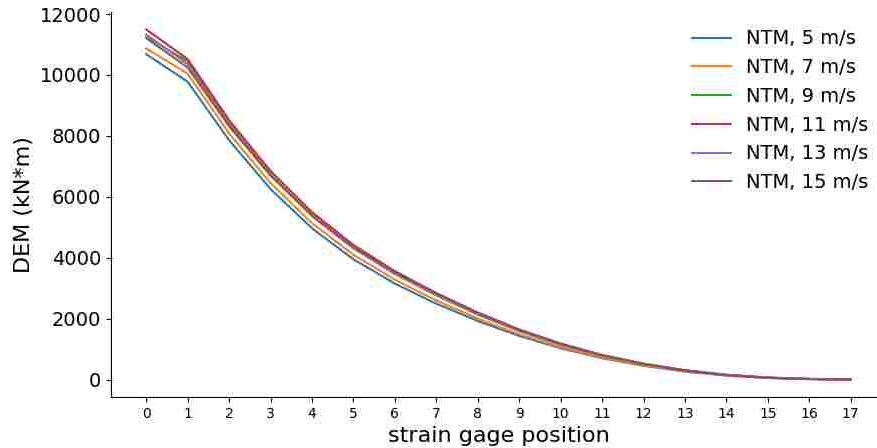
21

Figure 4.1: The edgewise damage equivalent moments calculated using a subset of wind files corresponding to DLC 1.2. In total, the dynamic response is evaluated 132 at various wind conditions.

equivalent and extreme moments. We show that this iterative method converges within four or five iterations to a solution similar to calculating the dynamic loading within the loop.

4.1 Fixed Dynamic Loading Methodology

To begin, initial values for the design variables are chosen. The baseline design of the NREL 5MW was used. We evaluate the dynamic response using FAST for the wind input files listed in the previous section. Within this routine, the damage equivalent moments in the edgewise and flapwise directions are calculated, as well as the extreme extrapolated moments. This is done for all wind input files (132 in total), so only a subset of edgewise DEMs are shown in Figure 4.1 for illustrative purposes. The individual runs are not dependent on each other, so this process can be parallelized and done in a matter of minutes.

Once the dynamic loading of the wind turbine has been evaluated, we compare the edgewise and flapwise DEM values, as well as the extreme moments, determined from using the different wind input files at each virtual strain gage position. The maximum damage equivalent and extreme moments at each position can then be used in the fatigue and strain constraints in the blade optimization routine. Examples of the maximum edgewise and flapwise damage equivalent moments are seen in Figure 4.2.
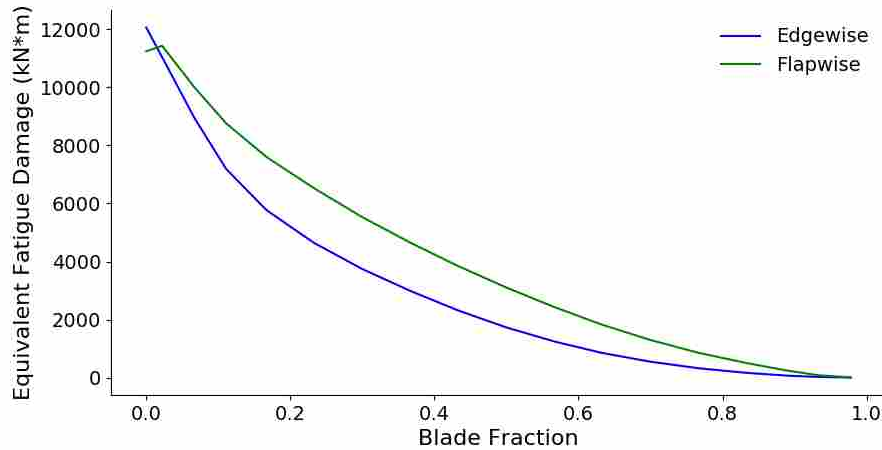
22

Figure 4.2: The maximum damage equivalent moments in the edgewise and flapwise directions. These are fixed during the blade optimization routine.

We then optimize the wind turbine blade to determine an improved blade design. Once the initial optimization is complete, we perform a dynamic simulation of the new wind turbine to calculate new DEMs and extreme moments based on the updated design. The blade design is re-optimized with the updated loading. We repeat this process until there is a small change in the objective value (less than 0.01%), COE. In summary, the iterative method follows this procedure:

1. Choose initial turbine blade design

2. Calculate DEMs and extreme moments using all wind input files

3. Determine maximum DEMs & extreme moments

4. Optimize with pre-calculated dynamic loading for use in strain and fatigue constraints

5. Set optimized design as new initial design

6. Repeat steps 2-5 until convergence

We validate this method by running a wind turbine blade optimization with FAST in-the-loop. We reduce the computational cost of running FAST by previously recording what wind input files are active, or what input files cause a DEM or extreme load to be the largest at a given point along the length of the blade. This shortened list is created in step 3 of the summary given above.

23

These wind files are then used in the optimization. After the optimization is complete, we calculate the dynamic loading using all the wind input files and determine whether the design constraints are still met. If a design constraint is violated, we include the wind input files that caused a violation of a design constraint and rerun the wind turbine blade optimization. We also include an optimized blade design when static loading with an additional safety factor of 1.35 to account for dynamic effects is used to constrain the blade design, which is done in other works [22].

4.2   Results

As can be seen in Figure 4.3, this method modifies the blade design and the wind turbine's COE in each successive iteration. In fact, after each successive iteration, the objective value of the optimization problem actually slightly worsens. This is due to the updating of the damage equivalent moments. After each iteration, the calculated damage equivalent moments are higher than the previous iteration, due to a slimmer blade. This can be seen when observing the chord spline in Figure 4.4. The twist distribution is also shown.

We note that when static loading with an additional safety factor to account for dynamic effects is used to constrain the blade design, the resulting COE and aeroelastic design are significantly different. As seen in Figure 4.3, the optimized COE is significantly lower, and the chord distribution is slimmer near the blade root (see Figure 4.4). However, when the fatigue damage using dynamic loading is calculated for this blade design, fatigue constraints are violated at the root of the blade, making the blade design as a whole is infeasible. Thus, the fatigue damage constraints have a significant effect on the design.

In addition, we completed an blade optimization where the damage equivalent moments and extreme moments are determined within the optimization loop for a subset of active wind input files. This result is shown in Figure 4.3, and the resulting COE is similar to the converged COE using the iterative method. In addition, the dynamic response of the wind turbine was evaluated at all wind files at the final design, and was found to not violate any constraints.

While using a subset of wind input files for the dynamic in-the-loop blade optimization worked for this specific wind turbine, that most likely will not be the usual case. If the final design violated constraints determined from design load cases not included in the optimization, the optimization would need to be redone. We needed to know what specific wind input files drive the
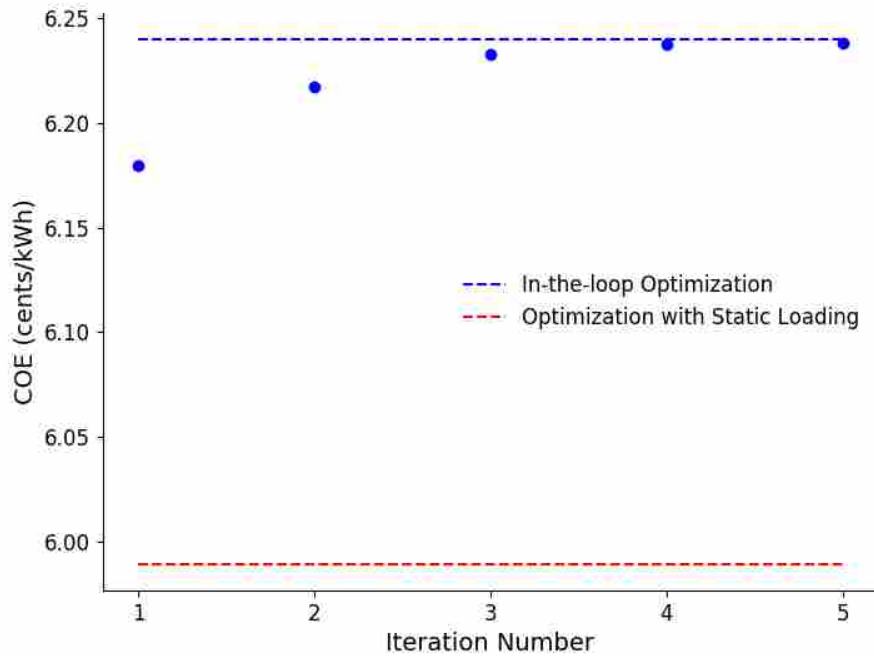
Figure 4.3: The cost of energy (COE) of the optimized design at the end of each iteration. By the fifth iteration, there is negligible change in the COE. The dashed blue represents the optimized result when the dynamic loading is calculated in the loop. The red line represents the optimized COE when static loading is implemented.

(a) Optimized chord splines

(b) Optimized twist splines
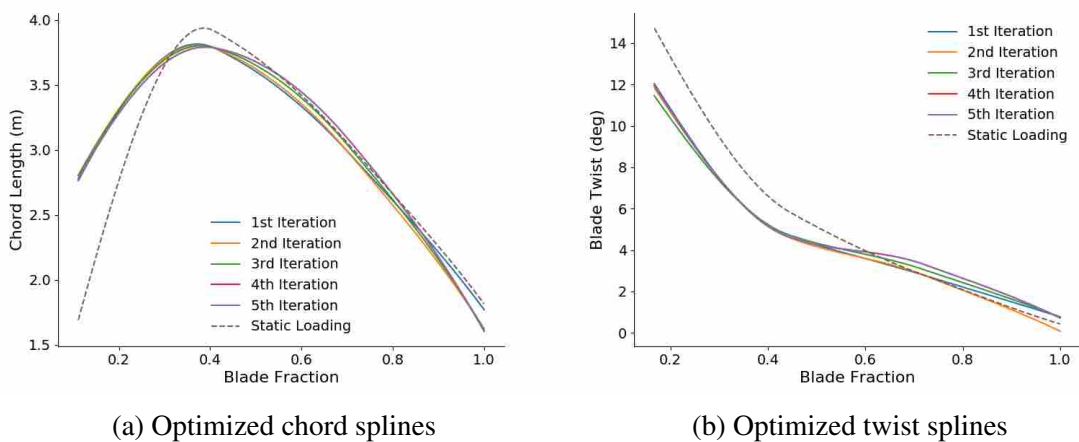
Figure 4.4: The optimized chord and twist splines at the end of each iteration. There is a noticeable difference between the splines where dynamic fatigue constraints are used, compared to when static loading constraints are used.

blade design, and only determined this by completing the iterative fixed-DEM blade optimization method. Finally, this optimization took on the order of two days to complete, where the fixed-DEM method took about five hours. While updating the effective fatigue damage is quite involved, the iterative approach is less computationally expensive than including the evaluation of the dynamic loading in the loop. In this case of optimizing the NREL 5MW reference turbine, we only needed to determine the dynamic loading four times. The fifth optimization showed that the method had converged.

CHAPTER 5.   SURROGATE APPROXIMATION OF DYNAMIC LOADING

The iterative method explained in the previous section significantly reduces the computational cost of incorporating the dynamic response of the wind turbine in its blade optimization routine. However, there is still significant work that must be done to incorporate a dynamic simulation code outside of the optimization routine. The wind input files for the wind turbine to be optimized must be generated as done in section 3.1. The pitch and torque control routines must also be specified for the wind turbine, as well as the turbine specifications used in FAST. These specifications must be translated into the correct format for the various input files that are used in FAST. In addition, we still must evaluate the dynamic response several times in the optimization to find an optimal design. While only four iterations were needed to optimize the NREL 5MW reference turbine, this may not be true for different rated wind turbines.

We have implemented a surrogate model that predicts the constraint values that are calculated using FAST outputs. This has the benefit of incorporating the dynamic response of the wind turbine without needing to have any familiarity with FAST or a dynamic response code. There is a sizable upfront cost to calculate the necessary data to train the surrogate model. However, by doing a proper dimensional analysis of the input and output parameters of the model, the calculated data can be generalized and used in the optimization of a wind turbine at various different conditions, including wind turbines of different power ratings.

This chapter will first define what a surrogate model is and how it is used. Next, the desired outputs from the model are defined, as well as the necessary inputs. We then discuss how the parameter values for the model inputs are determined. The fit type is then discussed, as well as the method to determine how accurate the surrogate model is. Optimization results are discussed, as well as the surrogate model's ability to be used for different wind turbine designs. Finally, drawbacks and needed future work are noted.

27

5.1 Construction of Surrogate Model

The dynamic simulation of a wind turbine using FAST and other routines produces a huge variety and depth of information. Some examples include output data related to nacelle yaw, platform and rotor-furl motions, tower, shaft, and hub loading, as well as generator and HSS information, all of which are calculated in this routine. For our purpose, though, we are only interested in the DEMs determined from the FAST-calculated moments along the length of the blade, as well as the extreme extrapolated moments M_e. These outputs are chosen because they are used to define constraints in the blade optimization problem, and are dependent on aspects of the blade design as well as the wind input files used in FAST. We can define these relationships as g and h, where

$$DEM = g(\text{blade design, wind input files}) \qquad (5.1)$$

$$M_e = h(\text{blade design, wind input files}) \qquad (5.2)$$

Not all aspects of the blade design and wind input files significantly affect the damage equivalent and extreme moments. We determined which aspects most affect these values by varying each specific design aspect x_i used in FAST. Each aspect was increased and then decreased by 10% from its nominal value. If varying x_i either way created more than a 3% relative difference in a value of the damage equivalent or extreme moments with respect to the calculated loading using the baseline configuration, we chose to include in g and h. We chose a high threshold to determine what design aspects would be included to reduce the complexity of the surrogate model, and the threshold value we used for the relative difference reflects this.

From this analysis, we determined the blade design aspects that most significantly affected the damage equivalent and extreme bending moments. For three blade horizontal axis wind turbines, these design aspects include the blade length r, chord distribution c, twist distribution θ, and rated torque Q. If the same set of design load cases are used, the turbulence intensity I_T used to generate the wind input files also affects the damage equivalent and extreme bending moments. We can then rewrite equations 5.1 and 5.2 as

$$DEM = g(c, r, \theta, Q, I_T) \qquad (5.3)$$

$$M_e = h(c, r, \theta, Q, I_T) \tag{5.4}$$

Using the Buckingham Pi theorem, we can rearrange equations 5.3 and 5.4 into non-dimensional Π groups by inspecting the units of each term. Specifically, the basic dimensions of each term are

$$DEM \doteq FL$$
$$M_e \doteq FL$$
$$c \doteq L$$
$$r \doteq L$$
$$\theta \doteq [\text{unitless}]$$
$$I_T \doteq [\text{unitless}]$$
$$Q \doteq FL$$

where L is length and F is force. By inspection, we can determine non-dimensional Π groups. These groups can be rearranged from equation 5.3 as

$$\Pi_1 = G(\Pi_2, \Pi_3, \Pi_4), \text{ where}$$
$$\Pi_1 = \frac{DEM}{Q} = \overline{DEM}$$
$$\Pi_2 = \frac{c}{r} = \overline{c}$$
$$\Pi_3 = \theta$$
$$\Pi_4 = I_T$$

Note that a new function G is used to relate the non-dimensional groups, and that G is dependent on only three variables, as opposed to g, which is dependent on five variables. A similar function H can be created for the extrapolated moments. The non-dimensional groups are exactly the same except for the first group, where M_e replaces DEM. The first Π group for both functions

29

Table 5.1: Outputs included in the surrogate model.

Output Name	Output Description
$\overline{DEM_y}$	DEMs in flapwise direction divided by Q
$\overline{DEM_x}$	DEMs in edgewise direction divided by Q
$\overline{M_{e,y}}$	extreme moments in flapwise direction divided by Q
$\overline{M_{e,x}}$	extreme moments in edgewise direction divided by Q

Table 5.2: Inputs used in the surrogate model.

Input Name	Input Description
I_T	IEC turbulence value
$\bar{c}$	Non-dimensional chord distribution
θ	Twist distribution

were set as the outputs of our surrogate model, and are listed in Table 5.1. The second, third and fourth Π groups are set as the outputs of our surrogate model, and are listed in Table 5.2.

Once the inputs and outputs of the surrogate model are defined, a number of training points must be chosen. A training point defines specific values for the surrogate model inputs. This point is then trained using FAST to determine the dynamic loading. Two methods to choose the values of the training points for the surrogate model were explored. The first option was a full factorial method. In this routine, a set number of values are chosen for each surrogate model input. This set number needed to be large enough to capture the interesting behavior of which our surrogate needs to model. These values span the input parameter domain. Then, the turbine response is calculated at each value combination. For example, as seen in Figure 5.1, ten values are chosen for two design variables (first and second point in the chord distribution). Thus, one hundred total training points are used and the dynamic loading is calculated at each point. The second option was to use Latin hypercube sampling (LHS) of the design space. LHS uses a stratified sampling scheme to improve on the coverage of a given input space. Figure 5.2 shows a sampling of one hundred points using the same design variables as before. Ultimately, most of the work produced was done using Latin hypercube sampling to choose the training point values. As noted in other works [48, 49], LHS reduces random sampling error, while requiring a similar amount of resources. Conversely, while a full factorial sampling yields the confident results, there is an associated extremely high cost in experimental resources.
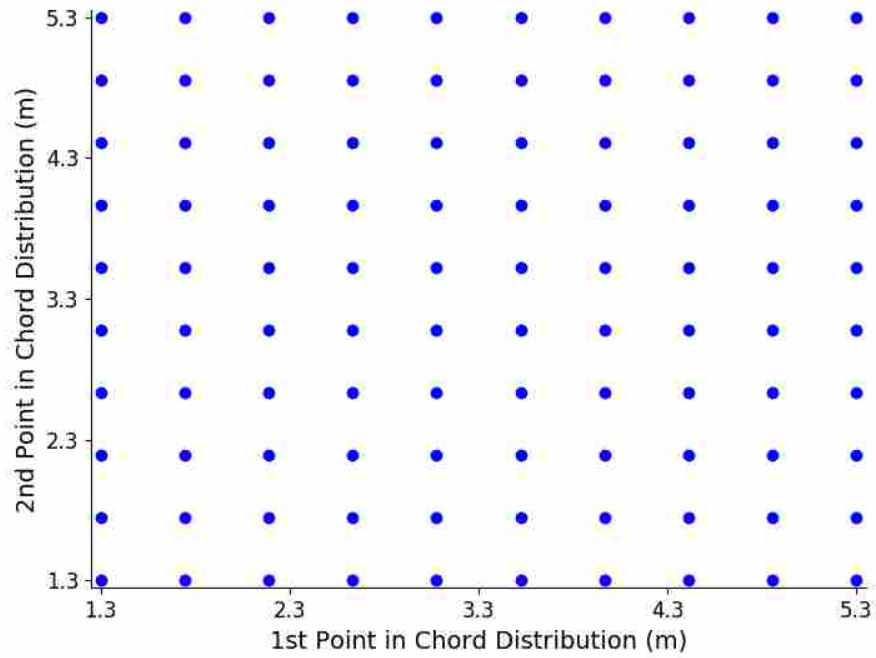
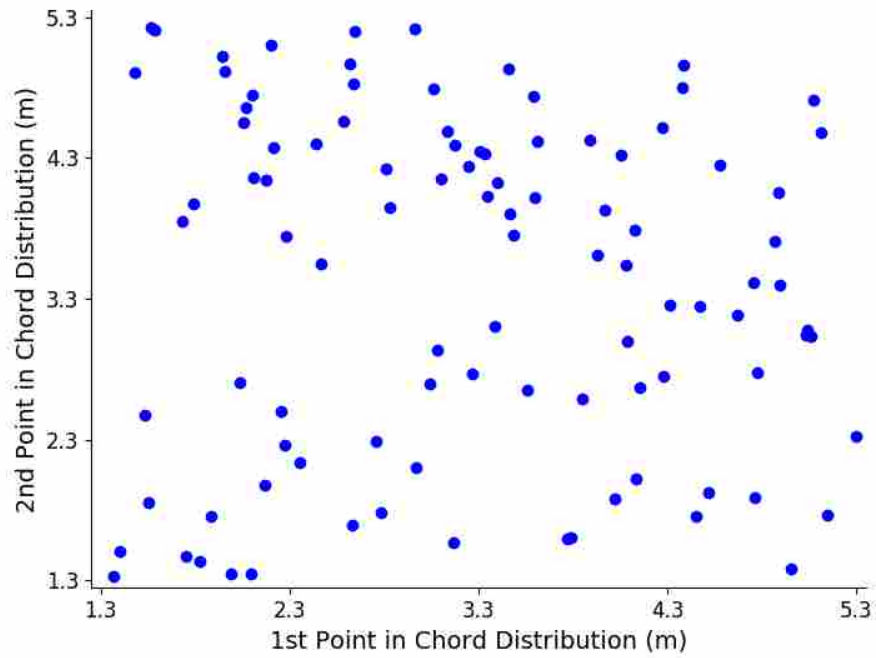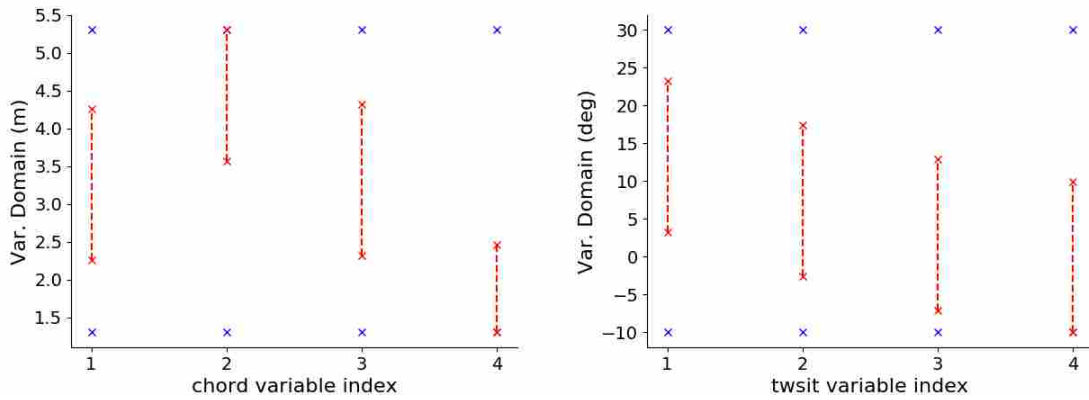Figure 5.1: Training points chosen that are linearly spaced across the design space.

Figure 5.2: Training points chosen using Latin hypercube sampling method.

(a) Restricted domain of the chord distribution.   (b) Restricted domain of the twist distribution.

Figure 5.3: An example of restricting the domain of the chord and twist control points. The blue markers represent the original bounds on the design variables, and the red markers represent the new, more restrictive bounds. Doing this increases the number of points trained where the optimization will likely drive the design.

After becoming familiar with this optimization problem, we can further restrict the domains of the chord and twist distributions. By doing so, more points are trained where the optimization will likely drive the design. This will also improve the accuracy of the surrogate model. An example of this is shown in Figure 5.3, where each control point of the chord and twist distributions is a given more restrictive bounds.

Once the selected points are trained, a cheap-to-evaluate function is created. A number of different function types can be used, such as a least squares approximation or a polynomial fit. In this work, we used a surrogate modeling toolbox[1] which provides a number of different fit types. Specifically, we incorporated a least squares, 2nd order polynomial, radial basis function, Kriging, Kriging partial least squares (KPLS), and a KPLS-kernel (KPLSK) method. As part of the construction of the surrogate model, we compared the accuracy of each of these fits.

5.2   K-Fold Cross Validation

We determined the accuracy of the surrogate model and the appropriateness of the fit type with cross-validation. We initially calculated a number of training points to create the surrogate model, and then additional points were trained to cross validate the calculated the fit. An example

---

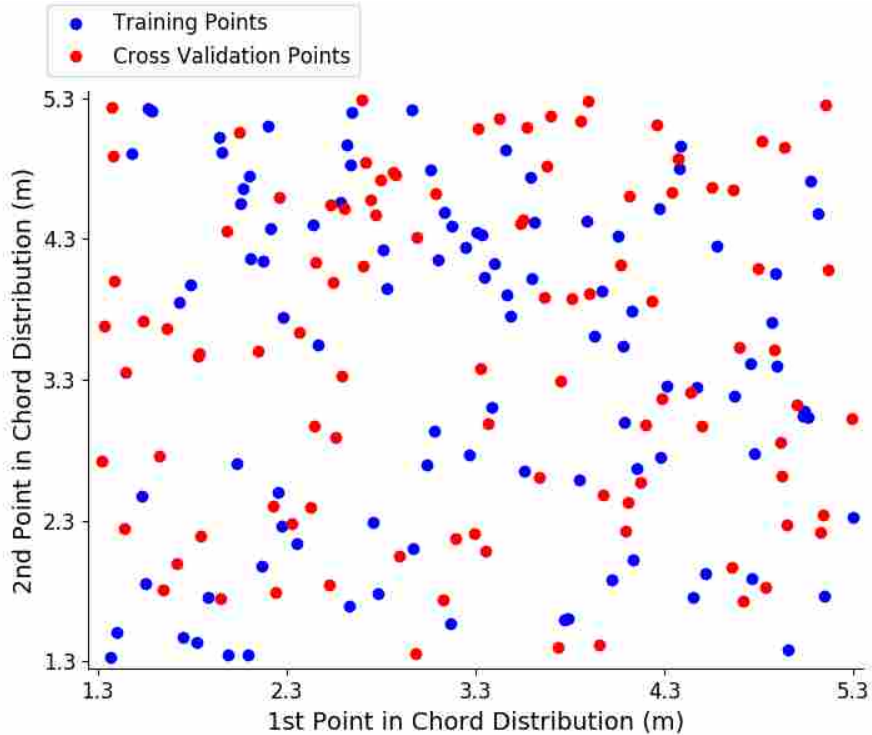[1]https://smt.readthedocs.io/en/latest/

Figure 5.4: An example of training and cross-validation points chosen for a 2D problem using Latin hypercube sampling. The sampling of the first and second control points in the chord distribution are shown.

of this cross-validation points overlying the initial training points in a 2D problem is shown in Figure 5.4.

While the approach seen in Figure 5.4 may be appropriate when there is not a significant computational cost to train additional points, this is not the case when evaluating f, the computation of the dynamic response of a wind turbine using FAST. Specifically, if we take the time to train additional points, we would want that these points are included in the creation of the surrogate model to improve its accuracy. Because of this, we implemented k-fold cross validation, which is a method to cross validate the surrogate model and use all calculated training points in the final surrogate model. As suggested by Kohavi [50], we did a 5-fold cross-validation, as this has been shown empirically to yield error estimates that have neither high bias nor variance.

We determined the accuracy of the surrogate model and the appropriateness of the fit type with k-fold cross validation. We show a rudimentary example in Figure 5.5. In this example, one hundred points are chosen, and the training point values for the first and control point of the chord
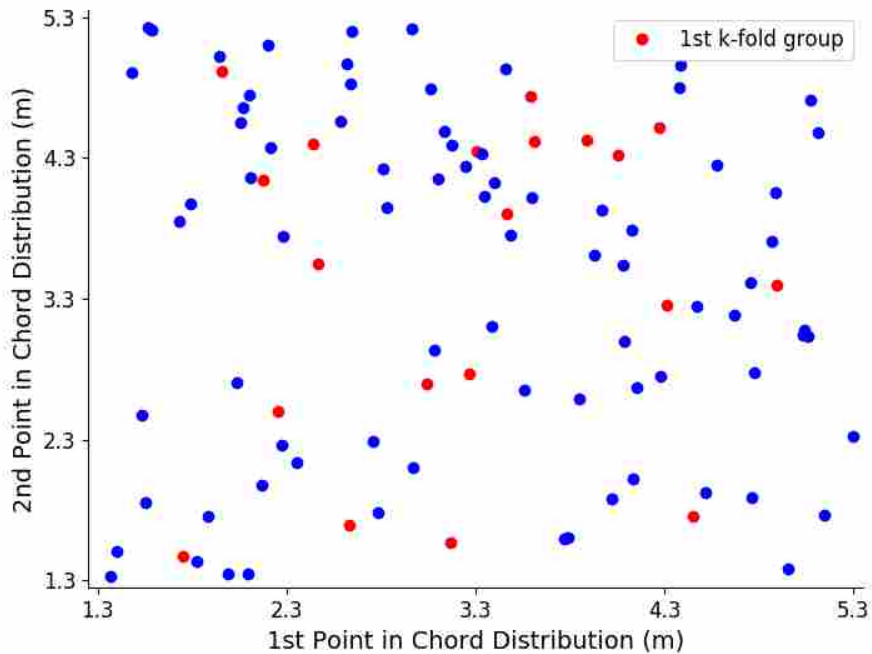
33

Figure 5.5: An example of training and cross-validation points chosen for a 2D problem using Latin hypercube sampling and K-fold cross-validation. The distribution of the first and second points in the chord distribution are shown.

distribution are shown. As stated before, a 5-fold cross-validation is used, where twenty points are removed from the training portion of the surrogate model. Once the surrogate model has been trained, the left-out twenty points are used to validate the created surrogate model. Specifically, the error between the predicted values and the actual values of the twenty points is determined. The root mean square (RMS) of these errors is then determined. This is then repeated for all folds, and the root mean square of all errors is then calculated. This results in a single error value, which can be compared to other error values using different fit types and number of training points.

To cross validate the created surrogate model, we implemented a 5-fold cross validation. The overall RMS error was calculated and compared using various fits and number of training points, as shown in Figure 5.6. The Kriging, Kriging partial least squares (KPLS), and KPLS-kernel (KPLSK) method generally resulted in similar error values ($\sim 3.06\%$), where the 2nd order polynomial, radial basis function and least squares fits resulted in higher errors ($\sim 3.52 - 6.83\%$). Therefore, we used a Kriging fit because it took less time to construct than the KPLS and PKLSK
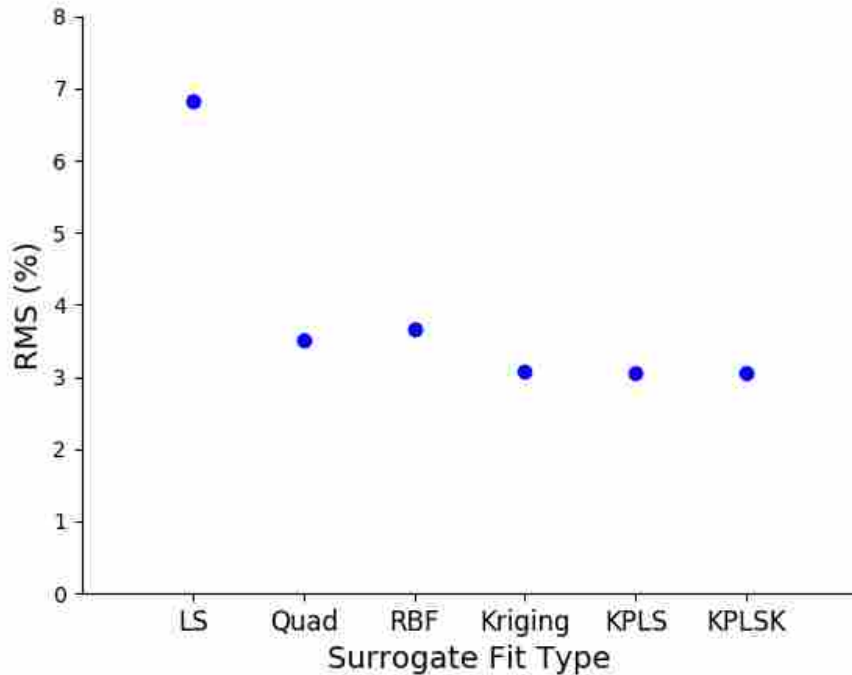
Figure 5.6: The root mean square error using different fits for a 5-fold cross validation.

methods. We show resulting error values when the number of training points is varied in the following section.

5.3 Results

To test the validity of our surrogate model, we trained it using four different rated wind turbines [43]. The turbine ratings for the four wind turbines were 750 kW, 1.5 MW, 3.0 MW, and 5.0 MW. Once we had calculated the points and trained the surrogate model using the data from all four wind turbines, we performed a 5-fold cross validation using various number of training points. The resulting root mean square error is shown in Figure 5.7. As expected, as the number of training points increases, the error of the model decreases. Also, using more than 1000 points for each turbine configuration does not significantly increase the surrogate model accuracy. The root mean square error for the equivalent fatigue damage along the length of the blade is shown in Figure 5.8. A Kriging fit was used to create the surrogate model.
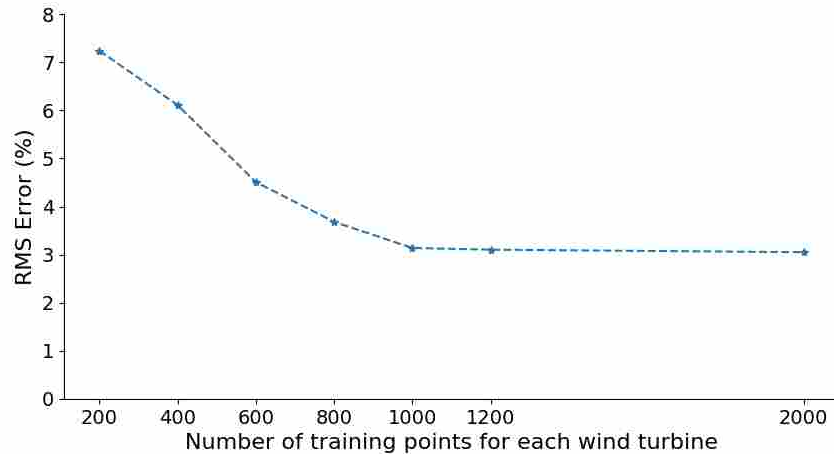
Figure 5.7: The root mean square error using a Kriging fit. There isn't a significant increase in surrogate model accuracy when more than 1000 training points are used for each reference turbine.

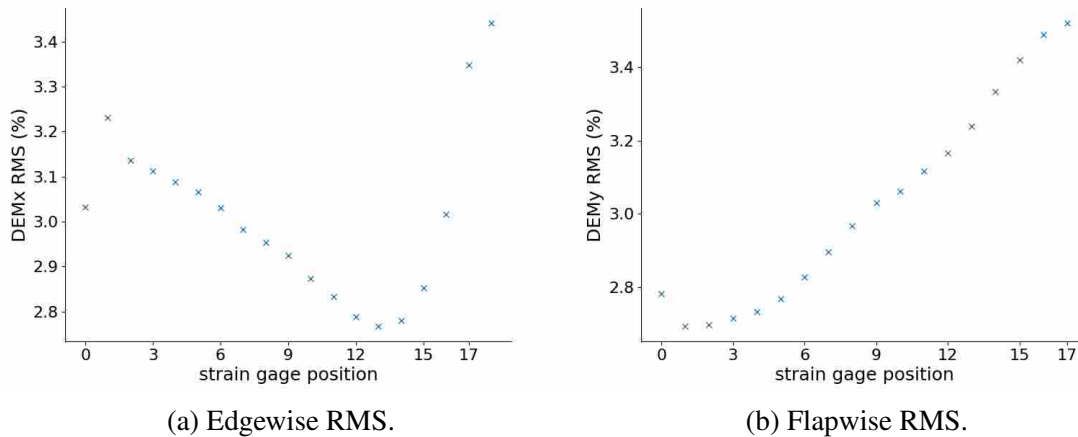(a) Edgewise RMS.

(b) Flapwise RMS.

Figure 5.8: The root mean square error using 1000 points for each reference turbine with a Kriging fit.

An important trait of this surrogate model is that it can predict the fatigue damage for a wind turbine that has not been used to in the training of the model. We tested this by training a surrogate model with the 1.5 and 5.0 MW WindPACT reference turbines. We next calculated the damage equivalent moments using FAST for the 3.0 MW reference (see Figure 5.9). Once the dynamic response had been evaluated, we compared the estimated damage from the surrogate model with the FAST calculated damage. This is shown in Figure 5.10. The surrogate model
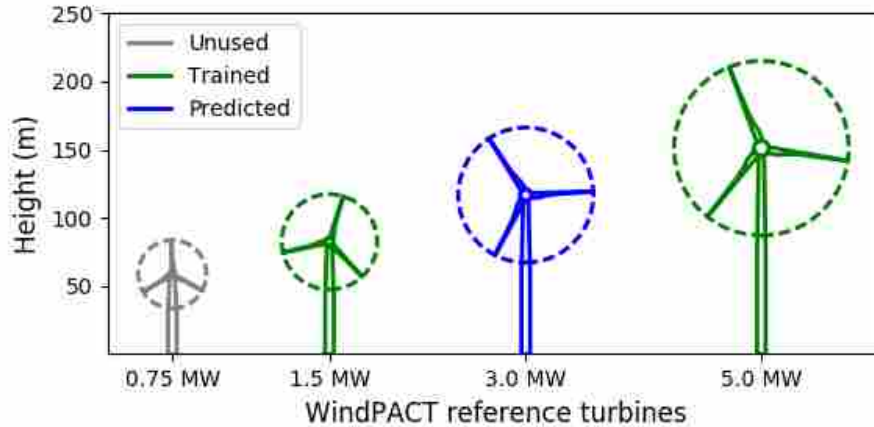
36

Figure 5.9: The predicted fatigue damage of the 3.0 MW turbine (blue) was determined from a surrogate model trained with the 1.5 & 5.0 MW turbines (green).

predicts the equivalent fatigue damage of the 3.0 MW WindPACT reference turbine, with a root mean square error of 1.15 %.

In addition, we wanted to determine how well the surrogate model could predict the fatigue damage for a wind turbine that not only was not used to train the model, but also was not in the range of rated power of wind turbines used to train the model. We did this by training the surrogate model using the 1.5 MW, 3.0 MW, and 5.0 MW WindPACT reference turbines. We next calculated the damage equivalent moments using FAST for the 0.75 MW reference (see Figure 5.11). Once the dynamic response had been evaluated, we compared the estimated damage from the surrogate model with the FAST calculated damage. This is shown in Figure 5.12. The surrogate model predicts the equivalent fatigue damage of the 0.75 MW WindPACT reference turbine with a root mean square error of 4.77 %.

The third cross validation of the surrogate model was to train the model using the four WindPACT reference turbines to predict the fatigue damage of the NREL 5 MW reference turbine. The NREL 5 MW turbine differs from the WindPACT 5.0 MW reference turbine with different
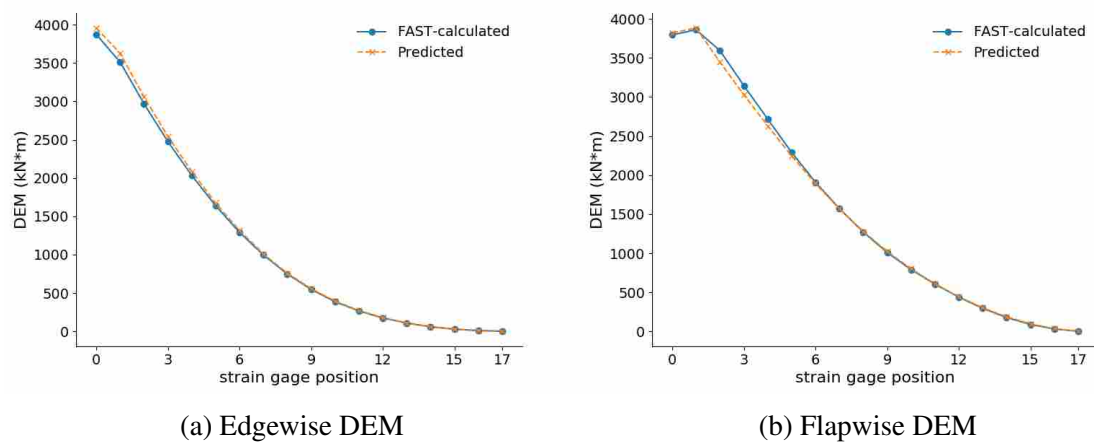
(a) Edgewise DEM

(b) Flapwise DEM

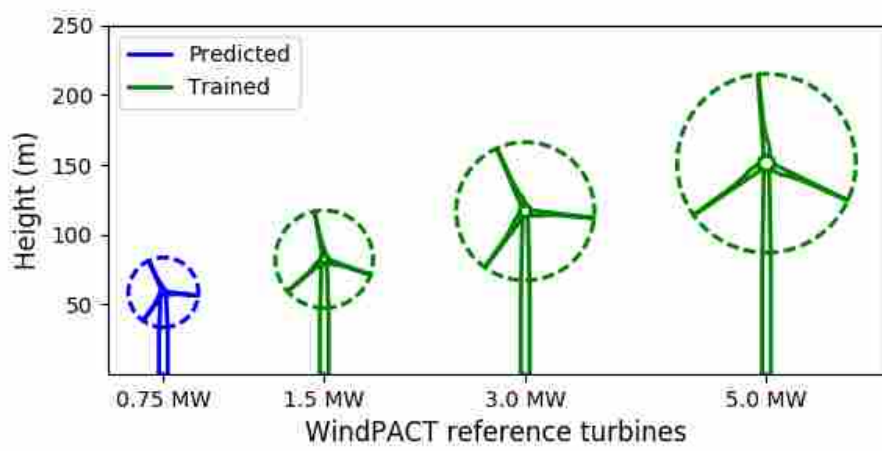Figure 5.10: The predicted and actual fatigue damage of the 3.0 MW WP reference turbine.

Figure 5.11: The predicted fatigue damage of the 0.75 MW turbine (blue) was determined from a surrogate model trained with the 1.5, 3.0 & 5.0 MW turbines (green).
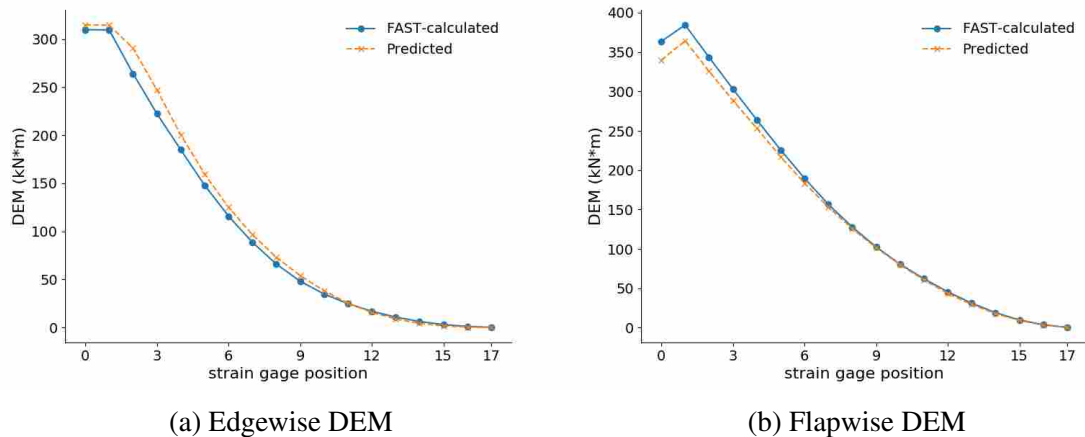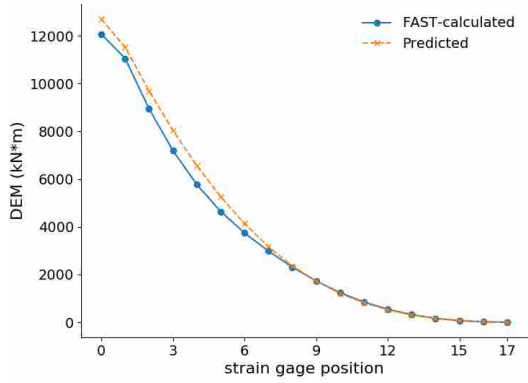
(a) Edgewise DEM

(b) Flapwise DEM

Figure 5.12: The predicted and actual fatigue damage of the 0.75 MW WP reference turbine.

nacelle, hub and generator properties, as well as the group of airfoils used to define the blade cross section. The results are shown in Figure 5.13. As can be seen in Figure 5.13a, the surrogate model over-predicts the fatigue damage in the edgewise direction, and as seen in Figure 5.13b, the surrogate model under-predicts the fatigue damage in the flapwise direction, with an overall root mean square error of 4.4 %.

Error plots for the predicted fatigue damage of this case is shown in Figure 5.14. As seen in both the edgewise and flapwise directions, there is a large relative error between the predicted and FAST-calculated damage equivalent moments at the tip (at the 17th strain gage) compared to the rest of the strain gage positions. However, the absolute error is about 0.23 kNm in the edgewise direction at the tip and 0.65 kNm in the flapwise direction at the tip.

Once we had cross validated the surrogate model, we tested the surrogate model within the optimization loop. The NREL 5MW reference turbine was used. The optimized objective (COE) was 6.229 cents/kWh, and the final aerodynamic design is shown in Figure 5.15. The final design was similar to the design determined using the iterative fixed-damage method. The computation time was on par with one iteration of iterative method, which shows that if we don't consider the upfront cost to train the surrogate model, this method is quite efficient in incorporating the wind turbine's dynamic response within the optimization loop.

(a) Edgewise DEM
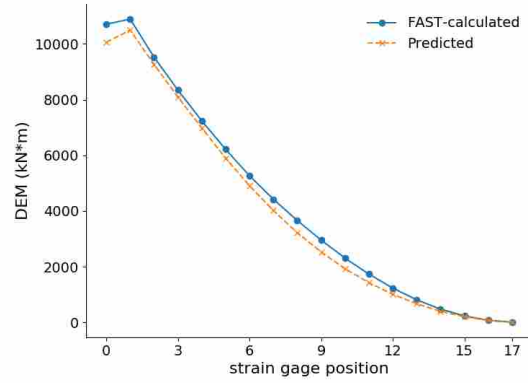
(b) Flapwise DEM

Figure 5.13: The predicted and FAST-calculated fatigue damage of the NREL 5.0 MW reference turbine.
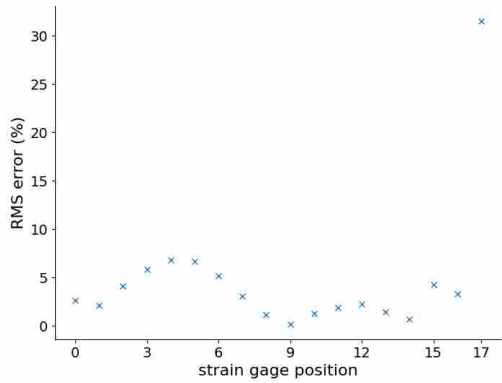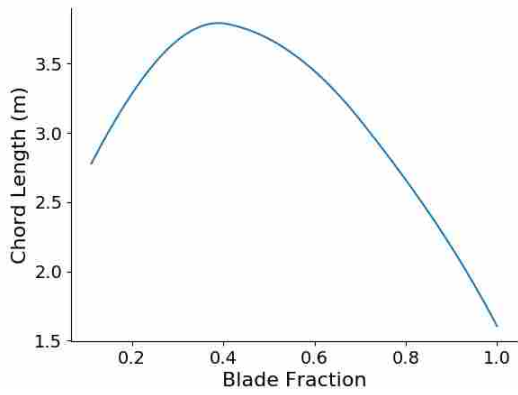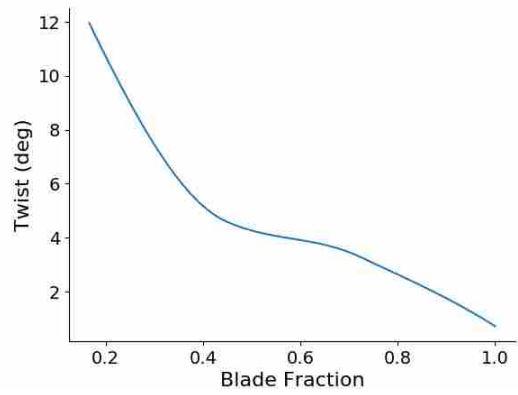

(a) Edgewise DEM
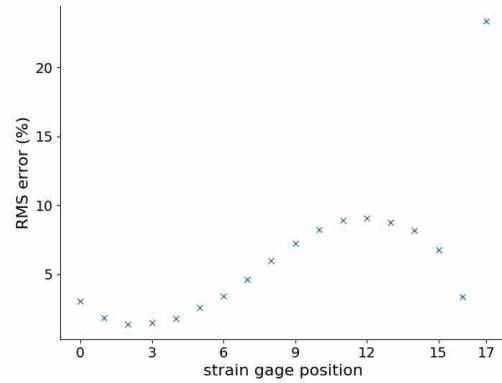
(b) Flapwise DEM Error

Figure 5.14: The error between the predicted fatigue damage and FAST-calculated fatigue damage using FAST of the NREL 5.0 MW reference turbine.

(a) Chord Distribution   (b) Twist Distribution

Figure 5.15: The optimized chord and twist distributions using the surrogate model.

CHAPTER 6.   CONCLUSIONS & FUTURE WORK

As shown using the iterative method, we were able to incorporate extreme moments and fatigue effects determined using the wind turbine's dynamic loading without having to necessarily calculate it inside the optimization loop. Within four to five iterations, the fixed-DEM method was able to determine a much improved blade design. This design had a similar COE compared to the COE of the optimized design when calculating the dynamic loading of the wind turbine within the optimization loop (see table 6.1). This method is considerably less expensive than evaluating the dynamic loading of the wind turbine within the loop. While we demonstrated the iterative method using a subset of all design load cases, the computational cost does not significantly increase when additional wind input files are included. This is because this computation is done outside of the optimization loop.

We developed a surrogate model that can be used to estimate the damage equivalent and extreme bending moments experienced by a wind turbine. As an extensive set of training data is built up, its use for additional turbines can be increased. Training data for specific studies can be also determined and used when we want to quickly estimate the dynamic loading for a given turbine.

Using a surrogate model to estimate the damage equivalent and extreme moments separates the work needed to calculate the dynamic loading with different studies that can be done to improve the efficiency of wind turbines. Once the dynamic loading has been generated, the surrogate model can be trained and used for a variety of different conceptual studies.

We showed that using a surrogate model results in an improved design with significantly less computational cost than including the evaluation of the dynamic loading within the loop. Once the upfront calculation for the training of surrogate model is complete, the optimization method is quite efficient, where the computation time was on par with one iteration of the fixed-DEM method. Comparative information for the developed methods in listed in table 6.1. Note that the

Table 6.1: Comparison of developed methods.

Method	Opt. Time	DLCs used	Opt. COE ($ / kWh)	Error in Dyn. Loading
In-the-loop	∼ 2 days	select few	0.062379	exact
Iterative	∼ 5 hours	full set	0.062374	<1%
Surrogate	∼ 1 hour	full set	0.062287	∼ 5 %

error in the dynamic loading using the iterative method varies during the optimization process, and progressively becomes smaller as more iterations are completed.

There are areas of improvement when integrating the surrogate model to estimate fatigue damage constraints. Work could be done to determine a reasonable safety factor that accounts for the error of the model-estimated dynamic loading compared to the dynamic loading calculated in FAST. We desired to limit the complexity of the surrogate model and used a high threshold value when determining what design aspects to include. This value could be lowered to include additional design aspects and increase the fidelity of the surrogate model.

In this work, we used a simple Gaussian fit for the simulated loading data to extrapolate extreme events. Calculating a consistent extreme event-extrapolation can be very difficult. Both Ragan [51] and Lott [52] discuss some of these problems, such as when data outliers must be accounted for and what statistical fit is appropriate for the collected data. More complex methods could be incorporated to better estimate the extrapolated loads in future work. In addition, we limited this study to three-blade wind turbines. Additional studies might include incorporating two-blade wind turbines.

Finally, we experimented with different wind turbine designs to incorporate into the surrogate model. One design we considered was the TUM 3.35 MW reference turbine. The chord and twist distributions used for this reference turbine are not well represented by an Akima spline with four control points, as seen in Figure 6.1. Future work could be to increase the number of control points used to define the aerodynamic shapes of the wind turbine blade, as well as to parameterize the control point positions.
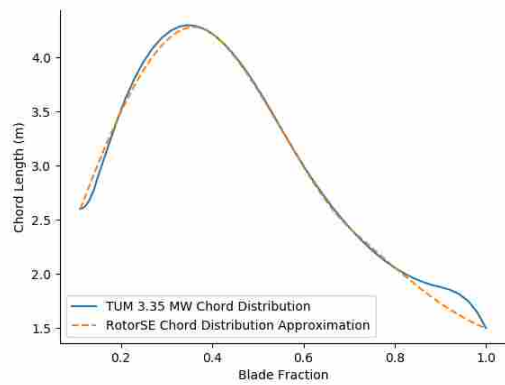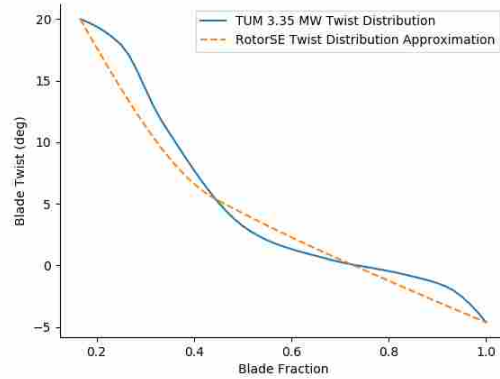
(a) Chord Distribution   (b) Twist Distribution

Figure 6.1: The TUM 3.35 MW chord and twist distributions, as well as the approximation used in RotorSE.

REFERENCES

[1] Kulunk, E., 2011. "Aerodynamics of wind turbines." In *Fundamental and Advanced Topics in Wind Power*. InTech. 1

[2] Hansen, M., 2013. *Aerodynamics of Wind Turbines*. Earthscan. 1, 7

[3] Wind energy basics. http://windeis.anl.gov/guide/basics/ Accessed: 2018-04-15. 1

[4] Johansson, T. B., Reddy, A. K., Kelly, H., Williams, R. H., and Burnham, L., 1993. *Renewable energy: sources for fuels and electricity*. Island press. 2

[5] Teleke, S., Baran, M. E., Bhattacharya, S., and Huang, A. Q., 2010. "Rule-based control of battery energy storage for dispatching intermittent renewable sources." *IEEE Transactions on Sustainable Energy,* **1**(3), pp. 117–124. 3

[6] IEC, 2005. "Wind turbines part 1: Design requirements, iec 61400-1, 3rd edition." *International Electrotechnical Commission (IEC)*. 3, 5, 13, 19

[7] Feng, J., and Shen, W., 2015. "Solving the wind farm layout optimization problem using random search algorithm." *Renewable Energy,* **78**, pp. 182 – 192. 4

[8] Jensen, N., 1983. *A note on wind generator interaction*. 4

[9] Gebraad, P. M. O. "Wind plant power optimization through yaw control using a parametric model for wake effects—a cfd simulation study." *Wind Energy,* **19**(1), pp. 95–114. 4

[10] Schubel, P., and Crossley, R., 2012. "Wind turbine blade design." *Energies,* **5**(9), pp. 3425–3449. 4, 5

[11] Negm, H. M., and Maalawi, K. Y., 2000. "Structural design optimization of wind turbine towers." *Computers and Structures,* **74**(6), pp. 649 – 666. 4

[12] Abdel-Hamid, R. H., Adma, M. A., Fahamy, A., and Samed, S. A., 2008. "Selection of wind turbine generator according to site data." In *European Wind Energy Conference*. 4

[13] Kong, C., Bang, J., and Sugiyama, Y., 2005. "Structural investigation of composite wind turbine blade considering various load cases and fatigue life." *Energy,* **30**(11-12), pp. 2101–2114. 5

[14] Manwell, J. F., McGowan, J. G., and Rogers, A. L., 2010. *Wind Energy Explained: Theory, Design and Application*. John Wiley & Sons. 5

[15] Lee, H. G., and Park, J., 2016. "Static test until structural collapse after fatigue testing of a full-scale wind turbine blade." *Composite Structures,* **136**, pp. 251–257. 5

[16] Bazilevs, Y., Korobenko, A., Deng, X., and Yan, J., 2016. "Fluid-structure interaction modeling for fatigue damage prediction in full-scale wind turbine blades." *Journal of Applied Mechanics,* **83**(6), p. 061010. 5

[17] Meng, H., Lien, F.-S., and Li, L., 2018. "Elastic actuator line modeling for wake-induced fatigue analysis of horizontal axis wind turbine blade." *Renewable Energy,* **116**, pp. 423–437. 5

[18] Bottasso, C. L., Campagnolo, F., and Croce, A., 2012. "Multi-disciplinary constrained optimization of wind turbines." *Multibody System Dynamics,* **27**(1), pp. 21–53. 5

[19] Bottasso, C. L., Croce, A., Sartori, L., and Grasso, F., 2014. "Free-form design of rotor blades." *Journal of Physics Conference Series,* **524**(1), June, pp. 12–41. 5

[20] Caboni, M., Campobasso, M. S., and Minisci, E., 2016. "Wind turbine design optimization under environmental uncertainty." *Journal of Engineering for Gas Turbines and Power,* **138**(8), p. 082601. 5

[21] Barrett, R., and Ning, A., 2016. "Comparison of airfoil precomputational analysis methods for optimization of wind turbine blades." *IEEE Transactions on Sustainable Energy,* **7**(3), July, pp. 1081–1088. 5

[22] Andrew Ning, S., Damiani, R., and Moriarty, P. J., 2014. "Objectives and constraints for wind turbine optimization." *Journal of Solar Energy Engineering,* **136**(4), 06, pp. 041010–041010. 5, 9, 21, 24

[23] Kenway, G., and Martins, J., 2008. *Aerostructural Shape Optimization of Wind Turbine Blades Considering Site-Specific Winds*. American Institute of Aeronautics and Astronautics. 5

[24] Ning, A., and Petch, D., 2016. "Integrated design of downwind land-based wind turbines using analytic gradients." *Wind Energy,* **19**(12), pp. 2137–2152 we.1972. 5, 9, 21

[25] Sessarego, M., Ramos-García, N., Hua, Y., and Shen, W. Z., 2016. "Aerodynamic wind-turbine rotor design using surrogate modeling and three-dimensional viscous–inviscid interaction technique.". 6

[26] Carrasco, A. D. V., Valles-Rosales, D. J., Mendez, L. C., and Rodriguez, M. I., 2016. "A site-specific design of a fixed-pitch fixed-speed wind turbine blade for energy optimization using surrogate models." *Renewable Energy,* **88**, pp. 112 – 119. 6

[27] Ning, S. A., 2013. Rotorse Tech. rep., National Renewable Energy Laboratory (NREL), Golden, CO (United States). 7, 9

[28] Gray, J., Moore, K., and Naylor, B., 2010. "Openmdao: An open source framework for multidisciplinary analysis and optimization." In *13th AIAA/ISSMO Multidisciplinary Analysis Optimization Conference*, p. 9101. 7

[29] Ning, A., 2014. "A simple solution method for the blade element momentum equations with guaranteed convergence." *Wind Energy,* **17**(9), Sept., pp. 1327–1345. 7, 10

[30] Ning, A., and Dykes, K., 2014. "Understanding the benefits and limitations of increasing maximum rotor tip speed for utility-scale wind turbines." *Journal of Physics: Conference Series,* **524**(1), p. 012087. 7

[31] Grasso, F., 2014. "Design of a family of new advanced airfoils for low wind class turbines." *Journal of Physics: Conference Series,* **555**(1), p. 012044. 8

[32] Van Rooij, R., and Timmer, N., 2004. "Design of airfoils for wind turbine blades." *Delft University of Technology, The Netherlands.* 8

[33] Mandell, J. F., Samborsky, D. D., Wang, L., and Wahl, N. K., 2003. "New fatigue data for wind turbine blade materials." In *ASME 2003 Wind Energy Symposium*, American Society of Mechanical Engineers, pp. 167–179. 8

[34] Kensche, C. W., 2006. "Fatigue of composites for wind turbines." *International Journal of Fatigue,* **28**(10), pp. 1363–1374. 8

[35] Kong, C., Bang, J., and Sugiyama, Y., 2005. "Structural investigation of composite wind turbine blade considering various load cases and fatigue life." *Energy,* **30**(11-12), pp. 2101–2114. 8

[36] Korobenko, A., Hsu, M.-C., Akkerman, I., Tippmann, J., and Bazilevs, Y., 2013. "Structural mechanics modeling and fsi simulation of wind turbines." *Mathematical Models and Methods in Applied Sciences,* **23**(02), pp. 249–272. 8

[37] Fuglsang, P., and Madsen, H. A., 1999. "Optimization method for wind turbine rotors." *Journal of Wind Engineering and Industrial Aerodynamics,* **80**(1-2), pp. 191–206. 9

[38] Veers, P., Lobitz, D., and Bir, G., 1998. Aeroelastic tailoring in wind-turbine blade applications Tech. rep., Sandia National Labs., Albuquerque, NM (United States). 9

[39] Larwood, S., 2009. "Dynamic analysis tool development for advanced geometry wind turbine blades.". 10

[40] Nwtc information portal (pbeam) `https://nwtc.nrel.gov/pbeam` Last modified 21-September-2014 ; Accessed 14-May-2018. 10

[41] Jonkman, J., and Buhl, M., 2005. "Fast user's guide." *NREL/EL-500e38230.* 12

[42] Jonkman, J., Butterfield, S., Musial, W., and Scott, G., 2009. "Definition of a 5-mw reference wind turbine for offshore system development.". 13

[43] Dykes, K., and Rinker, J. Windpact reference wind turbines Tech. rep., National Renewable Energy Lab.(NREL), Golden, CO (United States). 13, 35

[44] Jonkman, J., and Musial, W., 2010. "Offshore code comparison collaboration (oc3) for iea task 23 offshore wind technology and deployment offshore code comparison collaboration (oc3) for iea task 23 offshore wind technology and deployment." *NREL, Golden, CO.* 13

[45] Resor, B., 2013. "Definition of a 5mw/61.5 m wind turbine blade reference model." *Albuquerque, New Mexico, USA, Sandia National Laboratories, SAND2013-2569 2013.* 15

[46] Mandell, J. F., and Samborsky, D. D., 1997. Doe/msu composite material fatigue database: test methods, materials, and analysis Tech. rep., Sandia National Labs., Albuquerque, NM (United States). 19

[47] Madsen, P., Pierce, K., and Buhl, M., 1998. Predicting ultimate loads for wind turbine design Tech. rep., National Renewable Energy Lab., Golden, CO (US). 19

[48] Iman, R., 2008. "Latin hypercube sampling." *Wiley StatsRef: Statistics Reference Online*. 30

[49] Stein, M., 1987. "Large sample properties of simulations using latin hypercube sampling." *Technometrics,* **29**(2), pp. 143–151. 30

[50] Kohavi, R., et al., 1995. "A study of cross-validation and bootstrap for accuracy estimation and model selection." In *Ijcai*, Vol. 14, Montreal, Canada, pp. 1137–1145. 33

[51] Ragan, P., and Manuel, L., 2007. *Statistical Extrapolation Methods for Estimating Wind Turbine Extreme Loads*. American Institute of Aeronautics and Astronautics, 2018/04/25. 43

[52] Lott, S., and Cheng, P. W., 2016. "Load extrapolations based on measurements from an offshore wind turbine at alpha ventus." *Journal of Physics: Conference Series,* **753**(7), p. 072004. 43