

Feed Forward Wind Turbine Control Using Upwind Turbines as Sensors

By

ERIC WILLIAM ANDERSON

B.S. (University of California, Berkeley) 2000
M.S. (Georgia Institute of Technology) 2004

DISSERTATION

Submitted in partial satisfaction of the requirements for the degree of

DOCTOR OF PHILOSOPHY

in

Mechanical and Aerospace Engineering

in the

OFFICE OF GRADUATE STUDIES

of the

UNIVERSITY OF CALIFORNIA

DAVIS

Approved:

Professor C.P. van Dam, Chair

Professor Stephen K. Robinson

Dr. Jonathan White

Committee in Charge

2017

Copyright © 2017 by
Eric William Anderson
All rights reserved.

CONTENTS

List of Figures	vi
List of Tables	xvi
Abstract	xviii
List of Symbols	xix
List of Abbreviations	xxi
1 Introduction and Background	1
1.1 Wind Power	1
1.2 Wind Turbine Control	2
1.3 Advanced Turbine Control Research	4
1.4 Goals of This Research	6
1.5 Dissertation Outline	8
2 Using a Wind Turbine as a Sensor	11
2.1 Introduction	11
2.2 Data for evaluating sensor techniques	12
2.3 Wind speed estimate based on nacelle top anemometer	13
2.3.1 Feasibility	14
2.4 Turbine Model Based Wind Speed Estimator	20
2.4.1 Feasibility	22
2.4.2 Effect of rotor yaw misalignment	25
2.5 Convection speed estimation	30
2.5.1 A few thoughts on the limitations of Taylor's Frozen Turbu- lence Hypothesis	34
2.6 Other Potentially Useful Turbine Sensors	34
2.7 Summary and Conclusions	35

3 Feed Forward Optimum Pitch Control	37
3.1 Introduction	37
3.2 Using FAST to Model a Two Turbine System	41
3.3 Controller Design	43
3.3.1 Implementing Feed Forward Control	43
3.3.2 Closed Loop Controller	45
3.3.3 Feed Forward Controller	47
3.4 Performance With Perfect Wind Speed	
Knowledge	49
3.4.1 Gust Response	50
3.4.2 Turbulent Wind Response	55
3.5 Performance With Wind Speed Estimate From Upwind Turbine . . .	60
3.5.1 Gust Response	61
3.5.2 Turbulent Wind Response	65
3.6 Sensitivity to Errors in Feed Forward Data	
Timing	70
3.7 Summary and Conclusions	74
4 Turbine Derating Based on Feed Forward Signal From Upwind Turbine	76
4.1 Introduction	76
4.2 Turbine Derating	78
4.3 Effect of Derating on Dynamic Turbine	
Response	85
4.4 Transitioning Between Rated and Derated	
Operation	89
4.5 Control System Design	95
4.6 System Performance	98
4.6.1 Response to IEC Extreme Operating Gust	99
4.6.2 Response to Turbulent Wind With Large Gust	103

4.7 Conclusions	108
5 A Comparison of the NREL 5-MW Wake Characteristics Using Both SOWFA and OVERFLOW2	111
5.1 Introduction	111
5.2 Hybrid LES Method	112
5.2.1 SOWFA	112
5.2.2 Actuator Line Model	112
5.2.3 Aeroelastic Turbine Model	113
5.2.4 LES Computational Domain	114
5.3 RANS Computational Method	115
5.3.1 OVERFLOW2	115
5.3.2 RANS Computational Meshes	115
5.3.3 RANS NREL 5-MW Rotor Near-body Grid	116
5.4 Comparison of SOWFA and OVERFLOW2 Results	117
5.5 Effects of Near-Wake Grid Length	125
5.6 Effects of Near-Wake Grid Resolution	128
5.7 Conclusions	132
7 Conclusions	172
7.1 Summary and Findings	172
7.2 Future Work	176
A Sample FAST Simulation Input Files	178
A.1 primary.fst	179
A.2 Aerodyn.ipt	184
B WT_Perf	185
B.1 Example WT_Perf input file.	187

C Simulink Controllers	188
C.1 Baseline NREL 5-MW Feedback Control.	188
C.2 NREL 5-MW Turbine With Feed Forward Optimal Pitch Control. . .	193
C.3 NREL 5-MW Turbine With Feed Forward Derating Control.	195
D Damage Equivalent Load (DEL) Calculation	200
E SOWFA Implementation of Feed Forward Derating Control	204
E.1 Subroutine PitchCtrl()	204
E.2 Subroutine UserVSCntrl()	207
E.3 Subroutine updateControlParameters()	211
E.4 Subroutine UserHSSBr()	213
E.5 Module EAControl()	214
E.6 Additions to input file primary.fst	214
E.7 Additions to FAST_IO.f90	214

LIST OF FIGURES

1.1 Global wind power capacity.[1]	1
1.2 Torque-speed curve for NREL 5-MW turbine.[2]	3
1.3 Power-speed curve for NREL 5-MW turbine.	4
1.4 Wind turbine blades with trailing edge flaps for active load control.[3]	6
1.5 Illustration of upwind LIDAR measurement distances.[4]	6
1.6 Steady state behavior of the NREL 5-MW.[2]	7
2.1 Steady state rotor speed, blade pitch, generator torque, and tip speed ratio as a function of wind speed for the NREL 5-MW turbine.[2] . .	12
2.2 Illustration of TurbSim flow field.[5]	13
2.3 Vorticity downstream of an NREL 5-MW rotor in uniform 11 m/s wind.	15
2.4 Measurement locations for the rotor center wind speed and rotor average wind speed.	16
2.5 Anemometer measurement and rotor average wind speed for simulation #13 (8 m/s average wind speed).	16
2.6 Zooming in on Figure ?? shows high frequency oscillations are present in both anemometer measurements and rotor average wind speed. . .	17
2.7 Filtered anemometer and rotor average wind speed for simulation #13.	18
2.8 Discrepancy between anemometer and rotor average wind speed for simulation #13.	18
2.9 Discrepancy between anemometer and rotor average wind speed for all 8 m/s simulations.	19
2.10 Average magnitude of discrepancy (in m/s) as a function of simulation mean wind speed.	19
2.11 Average magnitude of discrepancy (in %) as a function of simulation mean wind speed.	20
2.12 C_p dependence on λ and collective blade pitch angle ($^\circ$) for the NREL 5-MW reference rotor.	22

2.13 C_p/λ^3 dependence on λ and collective blade pitch angle ($^\circ$) for the NREL 5-MW reference rotor.	23
2.14 Wind speed estimate and rotor average wind speed for simulation #13.	24
2.15 Filtered wind speed estimate and rotor average wind speed for simulation #13.	24
2.16 Discrepancy between wind speed estimate and rotor average wind speed for simulation #13.	25
2.17 Average magnitude of discrepancy as a function of simulation mean wind speed.	25
2.18 Average magnitude of discrepancy as a function of simulation mean wind speed.	26
2.19 Effect of rotor misalignment on wind speed estimate for simulation #13.	27
2.20 Effect of misalignment on average magnitude of wind speed estimate discrepancy.	28
2.21 Effect of rotor misalignment on apparent wind.	28
2.22 Angle corrected wind speed estimates for simulation #13.	29
2.23 Average magnitude of discrepancy in angle corrected wind speed estimates.	29
2.24 Estimating U_{mean} using a 60 s average of wind speed estimates for simulation #13.	32
2.25 Mean discrepancy between true U_{mean} and estimated U_{mean} for a variety of averaging times.	33
2.26 Max discrepancy between true U_{mean} and estimated U_{mean} for a variety of averaging times.	33
3.1 Steady state behavior of the NREL 5-MW.[2]	38
3.2 An example of wind speed fluctuations over 24 hours.[6]	38
3.3 NREL 5-MW response to wind speed fluctuations in control region 2.	39
3.4 NREL 5-MW response to wind speed fluctuations in control region 3.	40

3.5	Illustration of the two turbine system modeled in this chapter	42
3.6	The process used to simulate a two turbine system with FAST	43
3.7	Simple block diagram of turbine control w/o feed forward control.	44
3.8	Simple block diagram of turbine with feed forward control.	45
3.9	Block diagram of the NREL 5-MW closed loop control system, as mod- eled in this dissertation.	46
3.10	Block diagram of the NREL 5-MW controller with feed forward and closed loop control.	48
3.11	Steady state pitch (θ_s), in deg, as a function of incoming wind speed, in m/s.	48
3.12	Extreme Operating Gust (EOG) in 16 m/s wind.	51
3.13	Blade pitch caused by 16 m/s EOG when feed forward signal is based on true wind speed.	52
3.14	Rotor speed caused by 16 m/s EOG when feed forward signal is based on true wind speed.	52
3.15	Power generation caused by 16 m/s EOG when feed forward signal is based on true wind speed.	53
3.16	Tower base fore-aft moment caused by 16 m/s EOG when feed forward signal is based on true wind speed.	54
3.17	Blade root bending moment (BRM) caused by 16 m/s EOG when feed forward signal is based on true wind speed.	54
3.18	Rotor average wind speed and feed forward wind speed for 16 m/s turbulent wind.	56
3.19	Blade pitch caused by 16 m/s turbulent wind when feed forward sig- nal is based on true wind speed.	57
3.20	Rotor speed caused by 16 m/s turbulent wind when when feed forward signal is based on true wind speed.	57
3.21	Power generation caused by 16 m/s turbulent wind when when feed forward signal is based on true wind speed.	58

3.22 Tower base fore-aft moment caused by 16 m/s turbulent wind when feed forward signal is based on true wind speed.	59
3.23 Blade root bending moment (BRM) caused by 16 m/s turbulent wind when feed forward signal is based on true wind speed.	59
3.24 Extreme operating gust (EOG) and feed forward wind speed estimate based on gust.	62
3.25 Blade pitch caused by 16 m/s EOG when feed forward signal is based on wind speed estimate.	62
3.26 Rotor speed caused by 16 m/s EOG when feed forward signal is based on wind speed estimate.	63
3.27 Power generation caused by 16 m/s EOG when feed forward signal is based on wind speed estimate.	64
3.28 Tower base fore-aft moment caused by 16 m/s EOG when feed forward signal is based on wind speed estimate.	64
3.29 Blade root bending moment (BRM) caused by 16 m/s EOG when feed forward signal is based on wind speed estimate.	65
3.30 Rotor average wind speed and feed forward wind speed estimate of turbulent 16 m/s wind.	66
3.31 Blade pitch caused by 16 m/s turbulent wind when feed forward signal is based on wind speed estimate.	66
3.32 Rotor speed caused by 16 m/s turbulent wind when feed forward signal is based on wind speed estimate.	67
3.33 Power generation caused by 16 m/s turbulent wind when feed forward signal is based on wind speed estimate.	67
3.34 Tower base fore-aft moment caused by 16 m/s turbulent wind when feed forward signal is based on wind speed estimate.	68
3.35 Blade root bending moment (BRM) caused by 16 m/s turbulent wind when feed forward signal is based on wind speed estimate.	68

3.36 Tower base DEL for various timing errors in feed forward wind speed estimates.	72
3.37 Blade root DEL for various timing errors in feed forward wind speed estimates.	73
3.38 Standard deviation of power production for various timing errors in feed forward wind speed estimates.	73
4.1 Power generated during 16 m/s EOG for 30% derated turbine.	81
4.2 Tower fore-aft bending moment for 30% derated turbine during 16 m/s EOG.	81
4.3 Blade root moment for 30% derated turbine during 16 m/s EOG.	82
4.4 Rotor speed for 30% derated turbine during 16 m/s EOG.	82
4.5 Effect of sudden rating changes on blade root bending moment.	85
4.6 Effect of derating on avg. power for EOGs at various wind speeds.	86
4.7 Effect of derating on max tower moment for EOGs at various wind speeds.	87
4.8 Effect of derating on max BRM for EOGs at various wind speeds.	88
4.9 Effect of derating on max rotor speed for EOGs at various wind speeds.	88
4.10 Effect of derating on max overspeed for EOGs at various wind speeds.	88
4.11 Effect of sudden rating changes on rotor speed.	90
4.12 Effect of sudden rating changes on tower bending moment.	90
4.13 Effect of sudden rating changes on blade root bending moment.	91
4.14 Smoothing effect of the derating input filter when $P_{filter} = 0.2$	92
4.15 Improvements in rotor behavior from filtered derate commands.	92
4.16 Improvements in tower bending moment from filtered derate commands.	93
4.17 Improvements in blade root moment from filtered derate commands.	93
4.18 Effect of input filter settling time on rotor overspeed.	94
4.19 Effect of input filter settling time on peak tower bending moment.	94
4.20 Effect of input filter settling time on peak blade root moment.	95

4.21 Control system overview.	96
4.22 Extreme operating gust (EOG).	99
4.23 Effect of feed forward derating control on rotor speed caused by a 16 m/s EOG.	100
4.24 Effect of feed forward derating control on tower base fore-aft moment caused by a 16 m/s EOG.	100
4.25 Effect of feed forward derating control on BRM caused by a 16 m/s EOG.	101
4.26 Effect of feed forward derating control on power generation caused by a 16 m/s EOG.	102
4.27 Turbulent wind with 16 m/s mean wind speed and large gust.	104
4.28 Effect of feed forward derating control on rotor speed caused by 16 m/s turbulence with large gust.	105
4.29 Effect of feed forward derating control on tower base fore-aft moment caused by 16 m/s turbulence with large gust.	105
4.30 Effect of feed forward derating control on BRM caused by 16 m/s turbulence with large gust.	106
4.31 Effect of feed forward derating control on power generation caused by 16 m/s turbulence with large gust.	107
5.1 Near-body overset topology of the NREL 5-MW medium resolution grid system. (Source: R. Chow)	117
5.2 Iso-vorticity surfaces of the NREL 5-MW wake system for OVERFLOW2 and SOWFA simulations (Freestream flows from left to right. Images a,b, and c provided by R. Chow).	120
5.3 Instantaneous vorticity magnitude at a center cut plane of the NREL 5-MW rotor at $U_\infty = 11$ m/s using OVERFLOW2 (Freestream flows from left to right. Source: R. Chow).	121

5.4 Instantaneous vorticity magnitude at a center cut plane of the NREL 5-MW rotor for SOWFA simulations at $U_{\infty}=8, 11,$ and 15 m/s , as well as the SOWFA near-wake grid used in the simulations (Freestream flows from left to right)	122
5.5 Axial momentum deficit predicted by SOWFA and OVERFLOW2 for the NREL 5-MW rotor at $U_{\infty} = 8, 11,$ and 15 m/s	124
5.6 Instantaneous vorticity magnitude at a center cut plane of the NREL 5-MW rotor for SOWFA simulations at $U_{\infty} = 11 \text{ m/s}$ using near-wake grids of various lengths (Freestream flows from left to right).	127
5.7 Axial momentum deficit predicted by SOWFA for NREL 5-MW rotor at $U_{\infty} = 11 \text{ m/s}$ using near-wake grids of various lengths.	128
5.8 Instantaneous vorticity magnitude at a center cut plane of the NREL 5-MW rotor for SOWFA simulations at $U_{\infty} = 8, 11,$ and 15 m/s , using a 4 m near-wake grid (Freestream flows from left to right).	130
5.9 Axial momentum deficit predicted by SOWFA for NREL 5-MW rotor at $U_{\infty} = 8, 11,$ and 15 m/s , using both fine (1 m) and coarse (4 m) near-wake grids.	131
B.1 Blade element of width \hat{l}_r at radius $r.$ [9]	186
C.1 Top level Simulink system for model of NREL 5-MW with baseline (no feed forward) control.	189
C.2 Pitch Controller subsystem for Simulink model of NREL 5-MW with baseline control.	190
C.3 Gain Schedule subsystem for Simulink model of NREL 5-MW with baseline control.	190
C.4 Total Pitch Command subsystem for Simulink model of NREL 5-MW with baseline control.	191
C.5 Torque Controller subsystem for Simulink model of NREL 5-MW with baseline control.	191

C.6	Embedded Matlab function GenTorque for baseline control	192
C.7	Top level Simulink system for model of NREL 5-MW with feed forward optimal pitch control.	193
C.8	Feed Forward Controller subsystem for Simulink model of NREL 5-MW with feed forward optimal pitch control.	194
C.9	Pitch Controller subsystem for Simulink model of NREL 5-MW with feed forward optimal pitch control.	195
C.10	Top level Simulink system for model of NREL 5-MW with feed forward derating control.	196
C.11	Feed Forward Controller subsystem for Simulink model of NREL 5-MW with feed forward derating control.	197
C.12	Pitch Controller subsystem for Simulink model of NREL 5-MW with feed forward derating control.	197
C.13	Total Pitch Command subsystem for Simulink model of NREL 5-MW with feed forward derating control.	198
C.14	Torque Controller subsystem for Simulink model of NREL 5-MW with feed forward derating control.	199
C.15	Embedded Matlab function GenTorque for feed forward derating control.	199
D.1	Example of component loading.	201

LIST OF TABLES

3.1	EOG response with feed forward signal based on true wind speed.	55
3.2	16 m/s turbulent wind response when feed forward signal is based on true wind speed.	60
3.3	Short term damage equivalent loads (DEL) for 570 s simulation of turbine in 16 m/s turbulent wind when feed forward signal is based on true wind speed.	60
3.4	EOG response with feed forward signal based on wind speed estimate.	65
3.5	16 m/s turbulent wind response when feed forward signal is based on wind speed estimate.	69
3.6	Short term damage equivalent loads for 570 s simulation of turbine in 16 m/s turbulent wind when feed forward signal is based on wind speed estimate.	69
4.1	Effect of derating methods for test case 1: 16 m/s EOG with 30% derating.	83
4.2	Effect of derating methods for test case 2: 12 m/s EOG with 30% derating.	83
4.3	Effect of FF Control on dynamic response to 16 m/s EOG.	102
4.4	Effect of FF Control on damage equivalent loads for a large gust in turbulent wind with 12 m/s mean wind speed.	107
4.5	Effect of FF Control on damage equivalent loads for a large gust in turbulent wind with 16 m/s mean wind speed.	108
4.6	Effect of FF Control on damage equivalent loads for a large gust in turbulent wind with 20 m/s mean wind speed.	108
5.1	Extents and resolutions of the SOWFA grid refinement regions (distances extending from the center of the rotor).	115
5.2	NREL 5-MW simulation cases	118

5.3 Comparison of power and thrust predicted by three turbine simulation tools.	119
5.4 Power and thrust predicted by coarse grid SOWFA simulations as well as the percent difference with fine grid simulations.	129

ABSTRACT**Feed Forward Wind Turbine Control
Using Upwind Turbines as Sensors**

This dissertation seeks to improve the performance of a downwind turbine using information obtained from an upwind turbine. Methods for using an upwind turbine as a sensor are examined and evaluated. Next, a feed forward optimum pitch control scheme is introduced and investigated using a series of FAST simulations. In this control scheme an upwind is used as a wind speed sensor. Wind speed measurements are passed from the upwind turbine to a downwind turbine. The downwind turbine uses the knowledge of incoming wind speed to proactively adjust turbine blade pitch. Initial simulations are promising, but ultimately the feed forward control system is found to be very sensitive to timing errors. This make the control scheme impractical in real world applications. A feed forward derating control scheme is then developed. In this control scheme the rotor speed of the upwind turbine is monitored for evidence of an extreme gust event. If an extreme gust is detected the downwind turbine is derated to mitigate the negative effects of the gust. A series of FAST simulations show that this control scheme can reduce structural loads and rotor overspeeds in the downwind turbine. Unlike feed forward optimal pitch control, this system is insensitive to timing errors. A more sophisticated wind farm simulation tool, SOWFA, is then used to simulate and evaluate feed forward derating control. First, a verification and validation study of SOWFA is carried out. Good agreement is found between SOWFA, OVERFLOW2 and FAST results. SOWFA is then used to simulate two test cases where turbines are subjected to extreme coherent gusts. In both test cases feed forward derating control significantly improves performance, reducing damage equivalent loads and increasing energy capture.

LIST OF SYMBOLS

C_p	coefficient of power	—
D	downwind distance between turbines	m
$D_{fatigue}$	total damage due to cyclic loading	—
$I_{Drivetrain}$	low speed shaft equivalent inertia	$kg \cdot m^2$
L_{mean}	mean load over a components load history	—
$L_{range,i}$	range of a cyclic load	—
$L_{ultimate}$	ultimate design load	—
N_{Gear}	gear box ratio	—
$N_{equivalent}$	number of cycles to failure for the damage equivalent load	—
$N_{fatigue,i}$	number of cycles to failure for the i^{th} loading cycle	—
R	rotor radius	m
R	rotor radius	m
T_{Aero}	aerodynamic torque on the rotor due to the wind	$N \cdot m$
T_{Gen}	torque supplied by the generator	$N \cdot m$
T_{ff}	torque command issued by the feed forward controller	$N \cdot m$
U	incoming wind speed at the turbine rotor	$m \cdot s^{-1}$
U_{conv}	convection speed	$m \cdot s^{-1}$
U_{ffd}	dynamic feed forward velocity	$m \cdot s^{-1}$
U_{ff}	feed forward wind speed estimate supplied by upwind turbine	$m \cdot s^{-1}$
U_{mean}	local mean wind speed	$m \cdot s^{-1}$
U_{turb}	wind speed fluctuations due to turbulence	$m \cdot s^{-1}$
$\Omega_{GenFilt}$	low pass filtered rotational speed of the generator	$rad \cdot s^{-1}$
$\Omega_{GenR_Derated}$	rated rotational speed of the generator when derated	$rad \cdot s^{-1}$
Ω_{GenR}	rated rotational speed of the generator	$rad \cdot s^{-1}$
Ω_{Gen}	rotational speed of the generator	$rad \cdot s^{-1}$

Ω_{Rotor}	rotational speed of the rotor	$rad \cdot s^{-1}$
η_{DR}	derate factor	—
λ	tip speed ratio	—
ω_o	undamped natural frequency of the pitch actuator	Hz
ρ	air density	$kg \cdot m^{-3}$
θ	collective blade pitch angle of the turbine	deg
θ_c	pitch command issued by the closed loop controller	deg
θ_{ff}	pitch command issued by the feed forward controller	deg
θ_{ss}	steady state pitch function	deg
ξ	damping ratio of the pitch actuator	—
$k_{gen_Derated}$	region 2 generator torque constant when derated	$N \cdot m \cdot s^2 \cdot rad^{-2}$
k_{gen}	region 2 generator torque constant	$N \cdot m \cdot s^2 \cdot rad^{-2}$
m	Wöhler exponent	—
t_D	time required for a gust to pass from the upwind to downwind turbine	s
t_{DR_end}	time when downwind turbine begins return to full rating	s
t_{DR_start}	time when downwind turbine begins derating	s
t_{Event}	time when upwind turbine detects an extreme wind event	s
t_{conv}	estimate time for gust to travel between turbines	s
t_s	time required for turbine to transition into/out of derated operation	s

LIST OF ABBREVIATIONS

BRM	Blade Root bending Moment.
CFD	Computational Fluid Dynamics.
CGT	Chimera Grid Tools.
COE	Cost Of Energy.
DEL	Damage Equivalent Load.
ECG	Extreme Coherent Gust.
EOG	Extreme Operating Gust.
FAST	Fatigue, Aerodynamics, Structures, and Turbulence.
IEC	International Electrotechnical Commission.
LES	Large Eddy Simulation.
NREL	National Renewable Energy Laboratory.
NWTC	National Wind Technology Center.
PI	Proportional and Integral (Control).
RANS	Reynolds Averaged Navier Stokes.
.	
SST	Shear Stress Transport.

Chapter 1

Introduction and Background

1.1 Wind Power

Wind power is the fastest growing energy source in the world. From 2000 to 2015, global wind power capacity grew from 17.4 GW to 432.9 GW, with an average increase of 24% per year (Figure 1.1). In 2015, the increase in wind generation was equal to almost half of global electricity growth.[1] In the United States, installed wind capacity grew from approximately 2,500 MW in the year 2000 to 82,143 MW in 2016.[7] Though wind power only represents a small fraction of total global power generation, it is an important source of clean renewable energy and its importance will increase in the future.

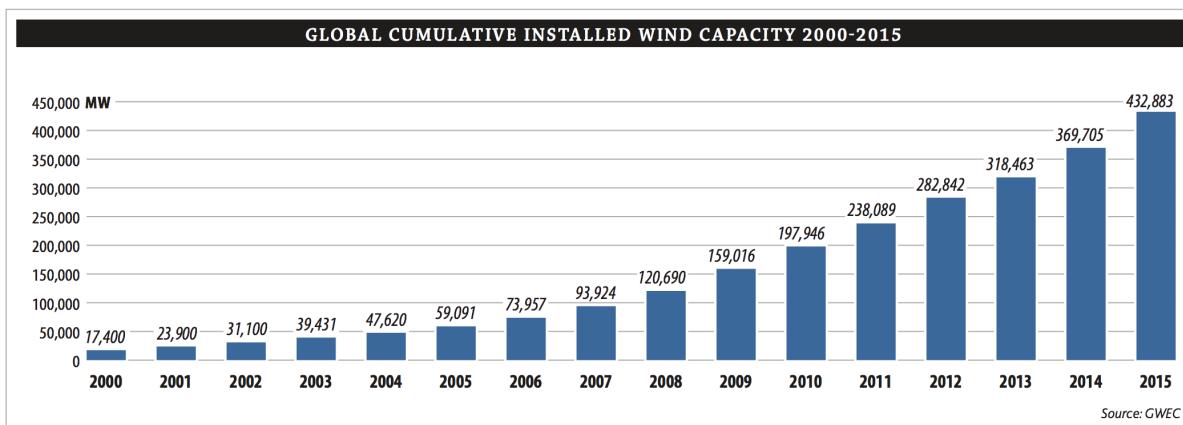


Figure 1.1. Global wind power capacity.[1]

Much of wind power's success is due to two factors: the widespread availability

of wind resources and the relatively low Cost of Energy (COE) when compared to other renewable energy sources. For example, utility scale wind energy typically costs 5-12 cents per KW-hr while utility scale solar-PV typically costs 15-30 cents per KW-hr.[8] Though some renewable energy sources such as geothermal plants or large hydro plants can produce energy at lower COE than wind farms, the natural resources needed for those plants are much more limited.

The cost of wind power has been steadily decreasing. The decreases in wind power COE have gone hand in hand with increases in both the size and complexity of wind turbines. As the turbines have grown in size and complexity, more sophisticated methods for controlling those turbines have been required.

1.2 Wind Turbine Control

A control system typically consists of sensors, actuators, and a controller. A modern wind turbine might include sensors such as: an anemometer, a wind vane, at least one rotor speed sensor, an electric power sensor, accelerometers, load sensors, pitch position sensors, various limit switches, vibration sensors, temperature and oil level indicators, hydraulic pressure sensors, operator switches, and push buttons. Actuators might include: hydraulic or electric pitch actuators, an electric generator that can actively control generator torque, generator contacts, switches for activating shaft brakes, and yaw motors. The controller collects information from sensors, processes that data, the issues commands to actuators. For a wind turbine, the controller typically consists of a computer based controller used for normal operation, and a highly reliable hard wired safety system that overrides the computer based controller and brings the turbine to a safe state if a serious problem occurs.[9]

The control system performs a variety of functions, but this dissertation is primarily concerned with the closed loop control system that optimizes turbine performance during power production. The majority of utility scale turbines built today use variable speed and collective pitch to feather control. In this control scheme

both the rotational speed of the turbine and the collective pitch of the turbine's three blades are controlled. At low and medium wind speeds the pitch is held constant while the turbine speed is varied. At high wind speeds both the turbine speed and the blade pitch can be varied.

At low and medium wind speeds, the rotational speed of the rotor is manipulated to maximize aerodynamic efficiency and maximize energy capture. The variations in rotational speed are achieved by torque control, which is implemented by using a pre-determined generator torque vs. generator speed curve like the one shown in Figure 1.2. The torque controller receives measurements from a generator speed sensor, looks up the appropriate generator torque on the speed-torque curve and commands the generator to produce that torque. The blue line in Figure 1.2 shows the torque vs. speed curve for the torque controller on the NREL 5-MW reference turbine. The black line shows the torque-speed curve corresponding to maximum aerodynamic efficiency of the turbine. Figure 1.3 shows the relationship between generator speed and generated power. In region 1 the generator produces no torque and no power, in region 2 the generator tracks optimum aerodynamic efficiency, and in region 3 the generator tracks constant power output. Regions 1.5 and 2.5 simply serve as smooth transitions between the other regions.

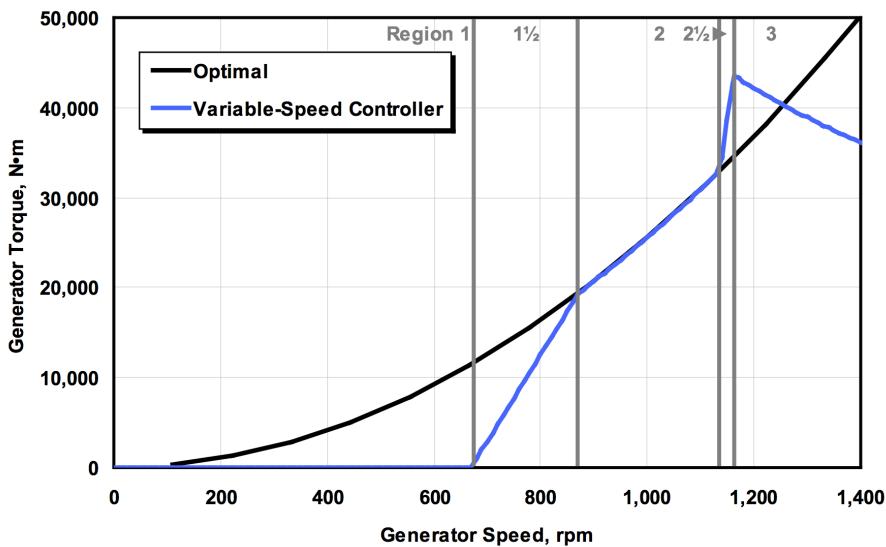


Figure 1.2. Torque-speed curve for NREL 5-MW turbine.[2]

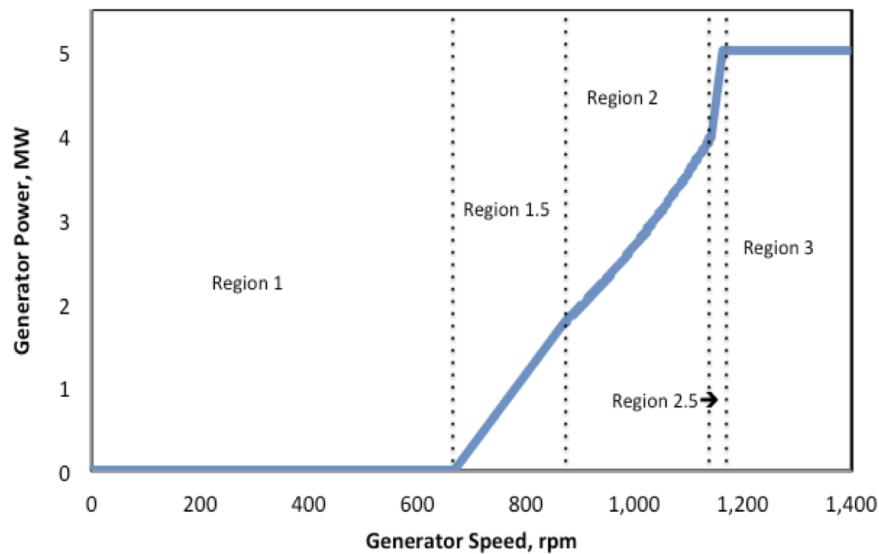


Figure 1.3. Power-speed curve for NREL 5-MW turbine.

At high wind speeds the collective blade pitch is manipulated to keep turbine speed and power generation constant. As wind speed increases, the turbine blades are pitched such that the turbine extracts a smaller fraction of the wind's power. As a result, the turbine is able to maintain the same power production even as the total available power in the wind increases. The purpose of pitch control is to manage loading on the turbine's mechanical components, and to prevent the turbine's power electronics from being damaged by excessive power production. For the NREL 5-MW turbine pitch control is achieved with a nonlinear PI controller in which the controller gains vary depending on the current blade pitch angle.

1.3 Advanced Turbine Control Research

The advanced wind turbine control concepts being researched today are numerous and diverse. A comprehensive review of this field is outside the scope of this proposal, but in the following paragraphs I will briefly describe several areas of advanced wind turbine control that are related to the work in this thesis: Individual pitch control, active load control, and LIDAR assisted control.

In individual pitch control, the pitch of each turbine blade is controlled independently and the blades can be set at different pitch angles. This has several advan-

tages over the collective pitch control described in the previous section. Individual pitch control can be used to reduce loading caused by rotor tilt and yaw alignment errors, airflow effects caused by wind turbine tower, or any other loads that vary cyclically with the rotation of the blade. Research has shown that cyclic pitch control could lead to a 30-40% reduction in fatigue loads on the turbine hub and a 20-30% reduction in fatigue loads at the blade roots.[10, 11] However, there is concern that the additional pitching will lead to premature failure of the blade pitching mechanism, or that it will require larger, more costly blade pitching mechanisms.[12]

Active flow control is the control of local airflow surrounding the blade. In the past, a great deal of research has focused on active flow control for aircraft. Recent research has sought to apply active flow technologies to commercial wind turbines as well. Active flow control typically operates on a small scale. It uses devices, such as the red trailing edge flaps shown in Figure 1.4, to exercise local control over the aerodynamic properties of the blade. These devices can independently control the aerodynamic properties of spanwise portions of a wind turbine blade. The devices also typically have very quick response times. Because they respond quickly and because they can exercise local control over smaller portions of the turbine blade, active flow control devices have the potential to mitigate some loads that other control methods cannot.

In LIDAR assisted control a LIDAR (Light Detection and Ranging) system is mounted on the nacelle of a turbine and used to scan the wind field as it approaches the wind turbine. As illustrated in Figure 1.5, the LIDAR system is capable of measuring wind speed at several discrete distances upwind of the turbine. These measurements are used to estimate the wind speed that the turbine will experience in the future. The turbine controller can use these future wind speed estimates to respond preemptively to changes in wind speed and to counteract changes in aerodynamic loading as they occur. The primary disadvantage to LIDAR control is the prohibitive cost of the LIDAR equipment.



Figure 1.4. Wind turbine blades with trailing edge flaps for active load control.[3]

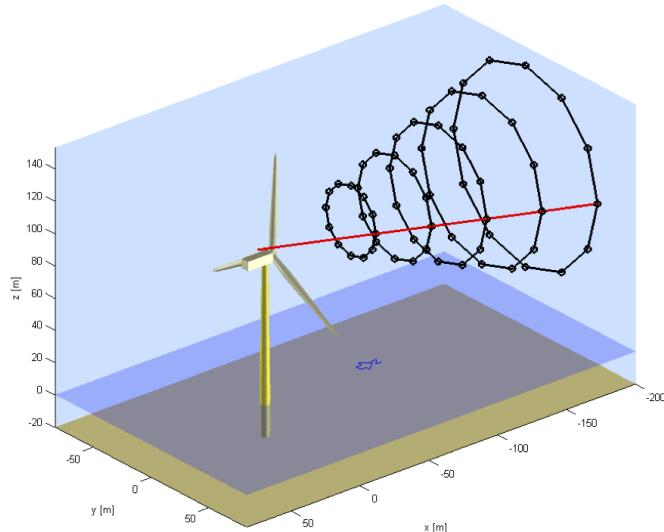


Figure 1.5. Illustration of upwind LIDAR measurement distances.[4]

1.4 Goals of This Research

This dissertation seeks to improve wind turbine control by using information obtained from upwind turbines. In particular, pre-existing sensors on upwind turbines will measure wind speed fluctuations and dynamic turbine behavior. That information will then be used by the control system of a downwind turbine to anticipate wind speed fluctuations before they reach the downwind turbine. This tech-

nique holds the potential to improve power production and/or decrease turbine loads by allowing the downwind turbine to react preemptively to changes in wind speed and track the optimum turbine conditions more closely.

Figure 1.6 shows the steady state power curve for the NREL 5-MW turbine. It represents the optimum power production at each wind speed. If the wind blows at a constant speed for long enough, the control system will bring the turbine's power production to a point on the blue line. However, since rapid wind speed fluctuations are common, the turbine often deviates from this power curve during normal operation. If the magnitude and/or time of these deviations can be reduced then the power curve can be tracked more closely and several potential benefits can be achieved.

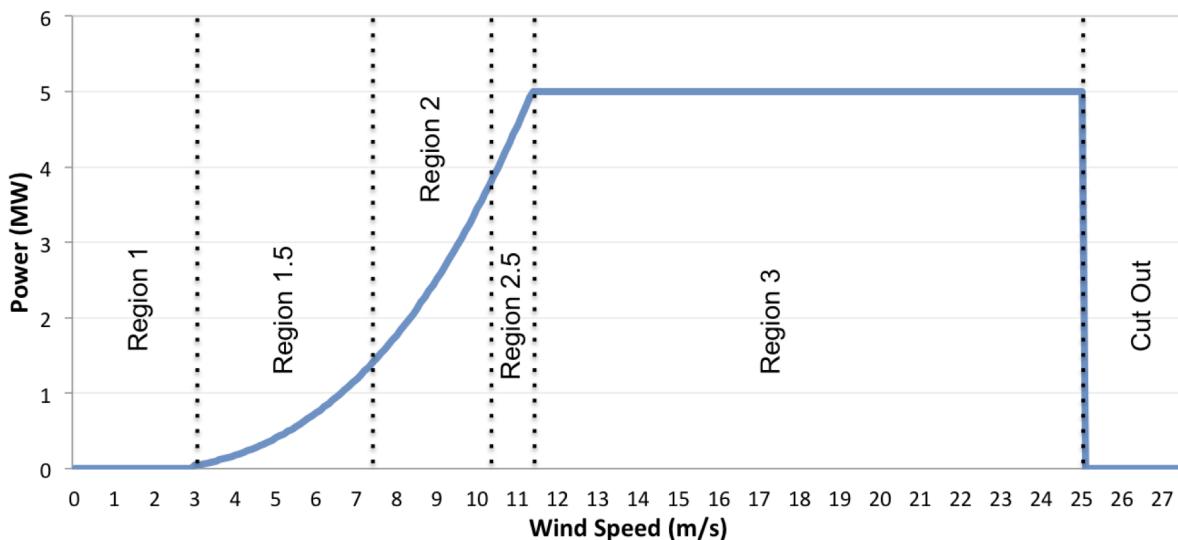


Figure 1.6. Steady state behavior of the NREL 5-MW.[2]

In region 2, deviations from the steady state power curve result in decreased aerodynamic efficiency and decreased power production. By tracking the power curve more closely in this region the power production of the turbine can be increased. In region 3, deviations from the steady state power curve cause higher loads on the mechanical components of the turbine and fluctuations in the power output of the turbine. By tracking the power curve more closely the peak mechanical loads and fluctuation in power production can be reduced.

Reducing peak mechanical loads has several advantages. It reduces wear and tear on the turbine's mechanical components, thereby reducing the long-term repair and maintenance costs. It can also lead to lower turbine-manufacturing costs or increased power production. Reductions in manufacturing costs are achieved by designing and sizing the turbine components based on the reduced loads. Alternately, increased power is achieved by "growing the rotor". Which means to increase the size of the turbine blades until the loading on the turbine components matches the original design loads.

Reducing the power fluctuations in region 3 can also be used to increase power production without any physical changes to the turbine. In part, the rated power of a turbine is set by the limitations of the turbine's power electronics. If improvements in wind turbine control can reduce deviations from rated power then the turbine can safely operate at a higher rated power without fear of overloading the power electronics. This technique is currently being used by General Electric with its WindBOOST control system. The WindBOOST control system update allows the GE 1.5 MW wind turbine to operate at a rated power of 1.6 MW without any changes to the turbine hardware.[13]

1.5 Dissertation Outline

Chapter 2 investigates using a wind turbine as a sensor. It discusses the sensors that are present on a typical wind turbine and how sensor data can be used to gather useful information for feed forward control. Much of the chapter focuses on using turbine dynamics to estimate wind speed and convection velocity. Chapter 2 also discusses the accuracy that can be expected from such estimates and discusses some of the factors that influence accuracy.

In Chapter 3, a feed forward optimum pitch control scheme is introduced and investigated. In this control scheme an upwind is used as a wind speed sensor. Wind speed measurements are passed from the upwind turbine to a feed forward controller on a downwind turbine. The downwind turbine uses the knowledge of in-

coming wind speed to proactively adjust turbine blade pitch, potentially improving energy capture and reducing damage to the turbine. The control scheme is investigated through three rounds of simulation using the NREL FAST turbine simulation tool. The initial round of simulations uses a simplified idealistic model of the two turbine system, while the subsequent rounds systematically incorporate complicating factors that a real world system would encounter.

Chapter 4 investigates the benefits and feasibility of derating a downwind turbine based on a feed forward signal from an upwind turbine. Several derating strategies from literature are investigated to determine which is most feasible for feed forward control. The effects of derating on turbine loading and rotor overspeeds are examined. The transition into and out of derated mode is also investigated, as well as ways to mitigate undesirable turbine behavior during the transition. A feed forward selective derating scheme based on these findings is then developed. The performance of that feed forward control system is tested and evaluated using a series of FAST simulations. The FAST simulation results are promising, but it is determined that testing Feed Forward Derating Control with a different and more sophisticated wind turbine simulation tool would be beneficial.

Chapter 5 is a verification and validation study of SOWFA (Simulator fOr Wind Farm Applications), a wind farm simulation tool. FAST must model a single turbine at time, but SOWFA can model a group of wind turbines as well as the atmospheric flow field surrounding those turbines. This gives SOWFA several advantages over FAST when modeling a two turbine system with feed forward control. However, SOWFA is much more computationally expensive than FAST and is less established in the wind turbine community. In Chapter 5 the NREL 5-MW rotor is simulated using SOWFA, OVERFLOW2 and FAST. The results from these fundamentally different simulation tools are compared to see how well they agree. In addition, the chapter evaluates how near-rotor grid resolution and near-rotor grid length affect SOWFA simulation results. The results shown in Chapter 5 increase confidence in SOWFA and provide guidance about how to properly tune and use SOWFA.

In Chapter ??, Feed Forward Derating Control is simulated and evaluated using SOWFA. The first portion of the chapter documents background work required to simulate the control system in SOWFA. Sections discuss implementing the feed forward controller, tuning SOWFA's actuator line model, choosing an appropriate computational domain, and gust modeling in SOWFA. The second portion of the chapter documents SOWFA simulation results for two test cases. One test case that matches the conditions assumed for Chapter 4 FAST simulations, and one test case where turbine wake effects are prominent. In each test case a two turbine system is subjected to a large gust. The behavior of the upwind turbine, a downwind turbine without feed forward control, and a downwind turbine with Feed Forward Derating Control are compared and analyzed.

Chapter 7 summarizes the dissertation and presents areas for future work.

The appendices contain supporting information that may be helpful to the reader. Appendix B includes descriptions of the simulation tools used in this dissertation, while Appendix D describes Damage Equivalent Loads (DEL) calculation. Appendix A contains examples of the input files used for FAST based simulations. The final two appendices document the control system implementations that were developed to model feed forward control in FAST (Appendix C) and SOWFA (Appendix E)

Chapter 2

Using a Wind Turbine as a Sensor

2.1 Introduction

When using a wind turbine as a sensor, the fundamental questions are: What should we measure, and how should we measure it? In this chapter several possible answers to those questions are investigated. To minimize cost, we will restrict ourselves to using sensors that are already present on a typical wind turbine. This chapter will first evaluate several methods for estimating wind speed, but will also discuss other potentially useful measurements.

There are two advantages to measuring wind speed. First, both the steady state torque and steady state collective blade pitch are ultimately functions of the incoming wind speed (as shown in Figure 2.1). Second, the wind speed provides an estimate of the speed at which turbulent eddies in the wind travel. As a result, the wind speed can be used to estimate how long it will take for the wind gusts observed by the upwind turbine to reach the downwind turbine.

Section 2.2 describes a set of 66 turbulent wind fields. Those wind fields are used in subsequent sections to evaluate sensor techniques. Section 2.3 discusses wind speed estimates based on nacelle top anemometer readings. Section 2.4 discusses the use of turbine dynamics for wind speed estimation. Section 2.5 discusses convection speed estimation, while Section 2.6 briefly discusses other turbine sensors that could be useful for feed forward control.

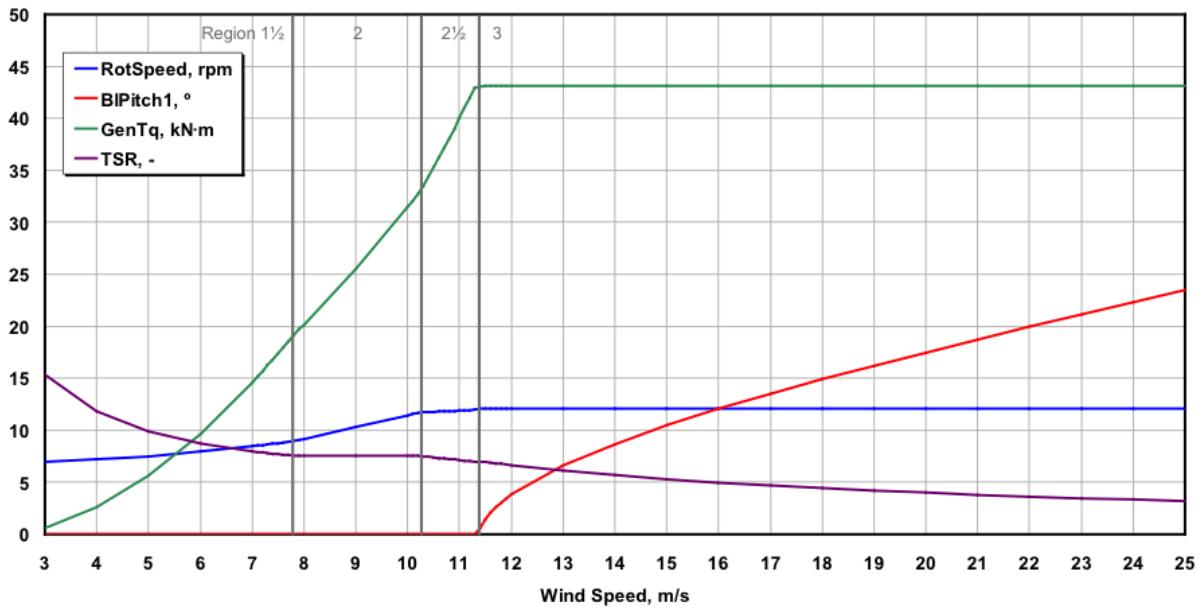


Figure 2.1. Steady state rotor speed, blade pitch, generator torque, and tip speed ratio as a function of wind speed for the NREL 5-MW turbine.[2]

2.2 Data for evaluating sensor techniques

The rest of this chapter discusses several wind turbine sensor scheme's that could potentially be used for feed forward control of downwind turbines. To gain insight into the performance of these scheme's, a series of simulations were conducted. The simulations were carried out in FAST (Fatigue, Aerodynamics, Structures, and Turbulence), using wind conditions simulated by TurbSim. FAST is a medium fidelity wind turbine simulation tool developed by NREL and Oregon State University. FAST models both the aerodynamics and structural dynamics of horizontal axis wind turbines, and has been independently validated and verified[14]. TurbSim is a stochastic, full-field, turbulent-wind simulator. It uses a statistical model (as opposed to a physics based model) to numerically simulate a time series of three-component wind-speed vectors at points on a two-dimensional vertical rectangular grid (see Figure 2.2)[5]. All simulations described in this section assumed an NREL 5-MW turbine and use TurbSim's great plains low level jet (GPLLJ) atmospheric model. The GPLLJ model is based on measurements from a 120 m tower and from an acoustic SODAR (sonic detection and ranging) wind profiler obtained during the

Lamar Low Level Jet Project in southeastern Colorado.[?] The site has flat terrain and most data was gathered in stable atmospheric conditions.

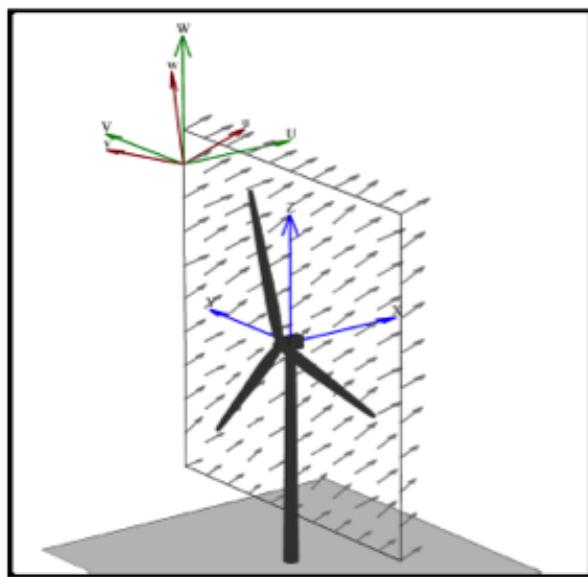


Figure 2.2. Illustration of TurbSim flow field.[5]

A variety of mean wind speeds were simulated to capture performance over the turbine's whole operating range. Eleven (11) mean wind speeds were simulated ranging from 4 m/s to 24 m/s in 2 m/s increments. At each wind speed 6 unique turbulent wind fields were generated using TurbSim. Each of these wind fields were used to drive a FAST simulation of the NREL 5-MW turbine. Each simulation was 10 minutes long. In total 66 10-minute simulations of an NREL 5-MW turbine in GPLLJ winds were performed. Sample FAST input files for the simulations can be found in Appendix A.

2.3 Wind speed estimate based on nacelle top anemometer

All utility scale wind turbines have nacelle-mounted anemometers. Anemometer measurements are used for purposes such as assessing wind turbine performance and forecasting future power production. Anemometer measurements are not used by conventional wind turbine control systems, but could potentially be used for wind

turbine control.

In some ways, using a nacelle top anemometer is the simplest method to estimate incoming wind speed. The anemometer is designed to measure wind speed. It is located on top of the nacelle, which is near the center of the rotor's swept area. Anemometers and data acquisition system are already in place on all modern utility scale wind turbines.

However, this method has several complications as well. The anemometer only measures wind speed for a very small portion of the flow field. On modern utility scale wind turbines, where rotors are often more than 100 meters in diameter, the anemometer reading may not be sufficient to accurately represent wind speed across the whole rotor. In addition, suitable anemometer data may not be readily available. For example, the data acquisition system on a Vestas V90 turbine does not provide raw anemometer data. Instead, it provides 10 minute average wind speeds that are based on anemometer data and a proprietary data processing technique. For the sake of this investigation we will assume that raw anemometer data can be obtained, perhaps with the help of the turbine manufacturer.

If suitable anemometer data is available, there may be difficulty obtaining an accurate wind speed measurement. Anemometer data has high frequency noise and occasional false values. The anemometer could have errors due to poor calibration. In addition, the nacelle top location of the anemometer means it is measuring wind that has passed through the rotor plane. The simulation results shown in Figure 2.3 were generated using SOWFA (Simulator fOr Wind Farm Applications), a simulation tool that will be discussed in Chapter 5. As Figure 2.3 shows, wind in the nacelle area can be significantly affected by the rotor. For the sake of this investigation, we will assume the anemometer data can be processed to obtain an accurate measurement of the incoming wind speed at the center of the rotor.

2.3.1 Feasibility

The simulation data set described in Section 2.2 was used to evaluate the feasibility of using a nacelle top anemometer to estimate incoming wind speed. FAST does not

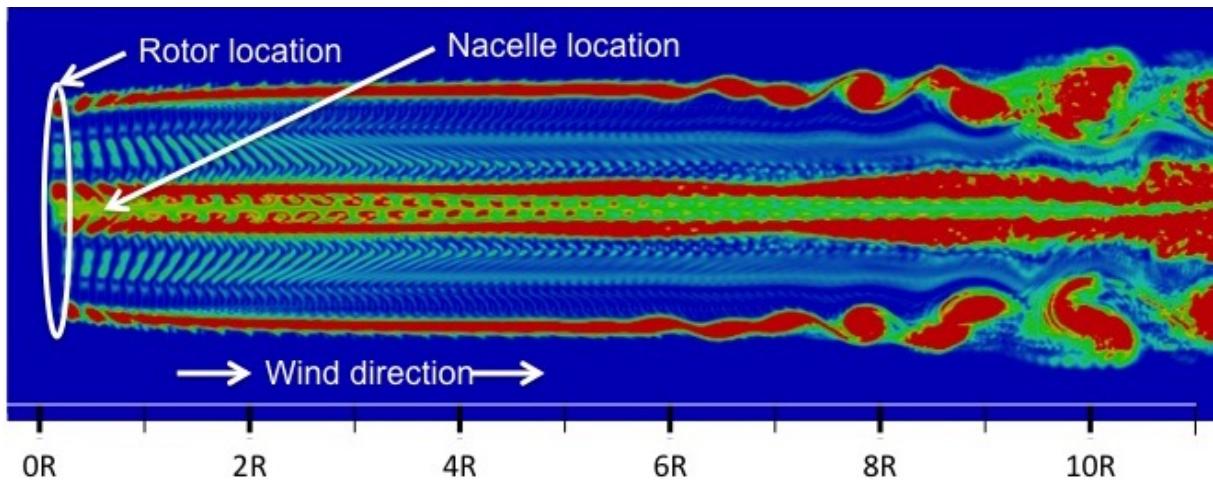


Figure 2.3. Vorticity downstream of an NREL 5-MW rotor in uniform 11 m/s wind.

simulate the nacelle top anemometer. However, if we make the assumptions stated in the previous two paragraphs an anemometer would perfectly measure wind speed at the center of the turbine rotor. The incoming wind speed at the rotor center can be extracted from the TurbSim flow field, and will be treated as the nacelle top anemometer measurement for the remainder of this discussion (Figure 2.4-a). The average wind speed passing through the rotor of the turbine can also be calculated from the TurbSim flow field (Figure 2.4-b). By comparing these two values we get a feel for how well a nacelle top anemometer can represent the wind passing through the rotor of a utility scale turbine.

Figure 2.5 shows wind speed at the anemometer as well as the average wind speed across the swept area of the rotor. The data shown in the figure is from the 13th simulation in the data set described above, but this data is typical of the other 65 simulations as well. At first glance the anemometer wind speed looks noisy compared to the rotor average wind speed. Upon closer examination (Figure 2.6) we see that both the anemometer and the rotor average wind speed have high frequency fluctuations. However, the high frequency fluctuations in the anemometer wind speed have a much larger magnitude.

The magnitude of these high frequency fluctuations is not very important from a wind turbine control point of view. Because utility scale wind turbines are such

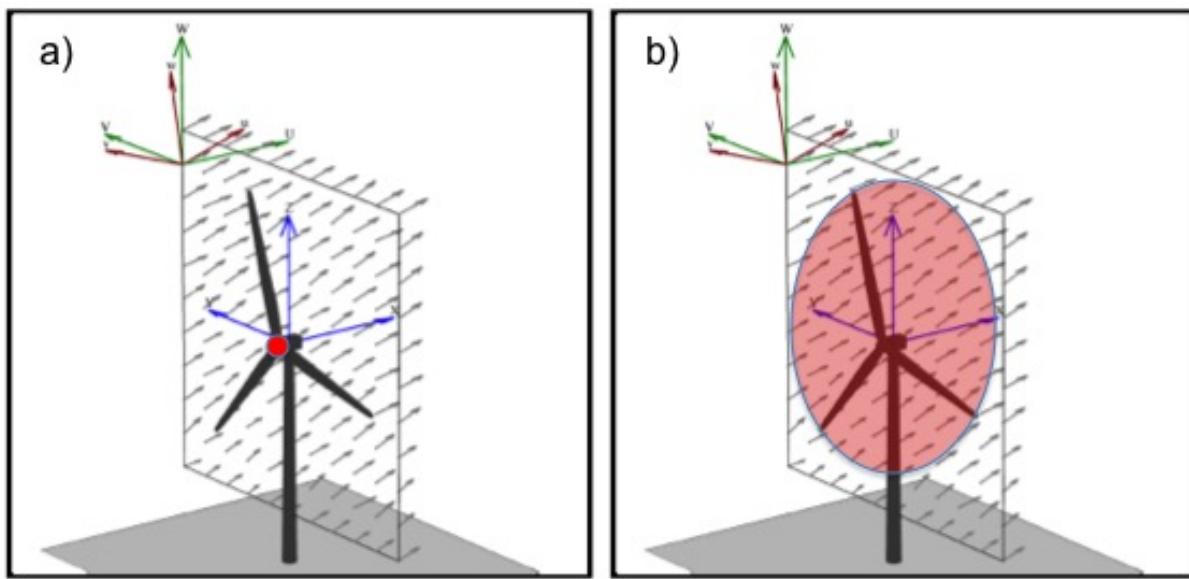


Figure 2.4. Measurement locations for the rotor center wind speed and rotor average wind speed.

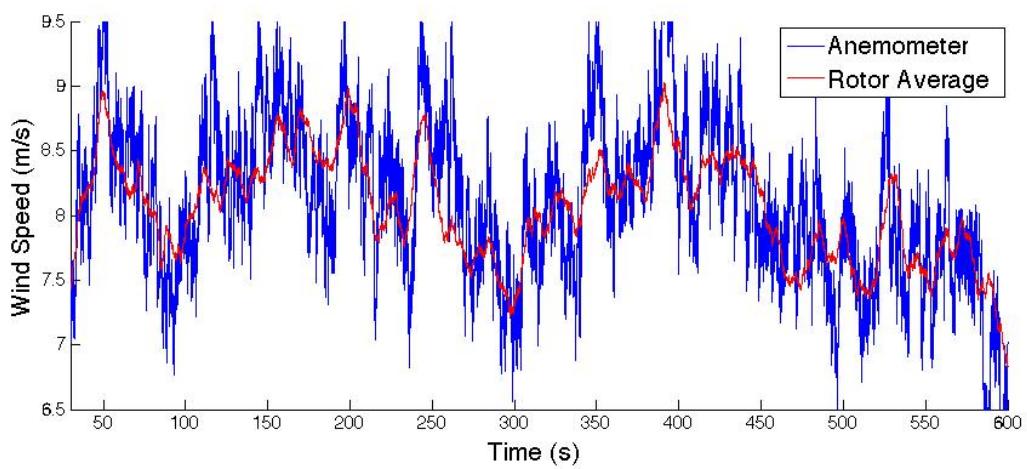


Figure 2.5. Anemometer measurement and rotor average wind speed for simulation #13 (8 m/s average wind speed).

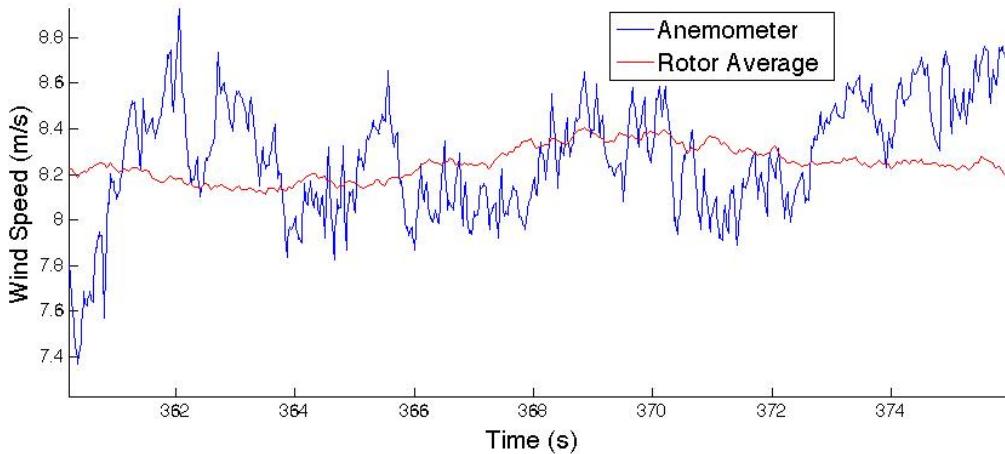


Figure 2.6. Zooming in on Figure 2.5 shows high frequency oscillations are present in both anemometer measurements and rotor average wind speed.

large machines, it isn't practical for the blade pitch and generator torque to track high frequency fluctuations in wind speed. In a typical utility scale turbine controller feedback signals are passed through a low pass filter before they are used for turbine control. To get a more meaningful comparison between anemometer wind speed and rotor average wind speed we should filter out the high frequency fluctuations.

Figure 2.7 shows anemometer wind speed and rotor average wind speed from simulation #13 after they have been passed through a low pass filter. I chose to use a 4th order zero-phase filter with a 0.25 Hz cutoff frequency. A zero phase filter was chosen because it does not result in a delay of the filtered signal. A 0.25 Hz cutoff frequency was chosen because it is the corner frequency of the low pass filter used by the NREL 5-MW control system.

Figure 2.8 shows the discrepancy between the filtered anemometer wind speed and the filtered rotor average wind speed for simulation #13. For this run the magnitude of the discrepancy ranges from almost 0 to as much as 0.954 m/s, and the average magnitude of the discrepancy is 0.236 m/s. All 6 simulations run with an 8 m/s wind speed yield similar data, as can be seen in Figure 2.9. Between these 6 simulations there are 57 minutes of data. If the magnitudes of the discrepancies are averaged over all data points in these 6 simulations, we find the discrepancy

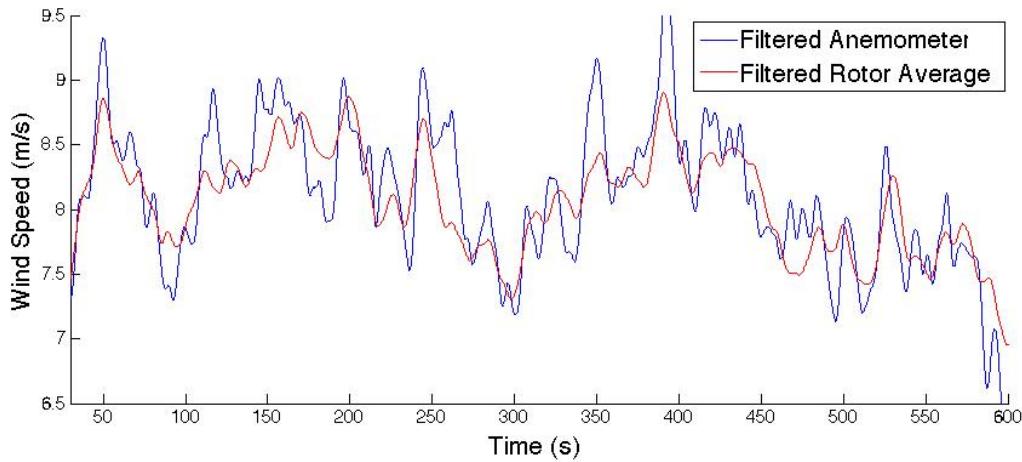


Figure 2.7. Filtered anemometer and rotor average wind speed for simulation #13.

between the filtered anemometer wind speed and the filtered rotor average wind speed averages 0.272 m/s.

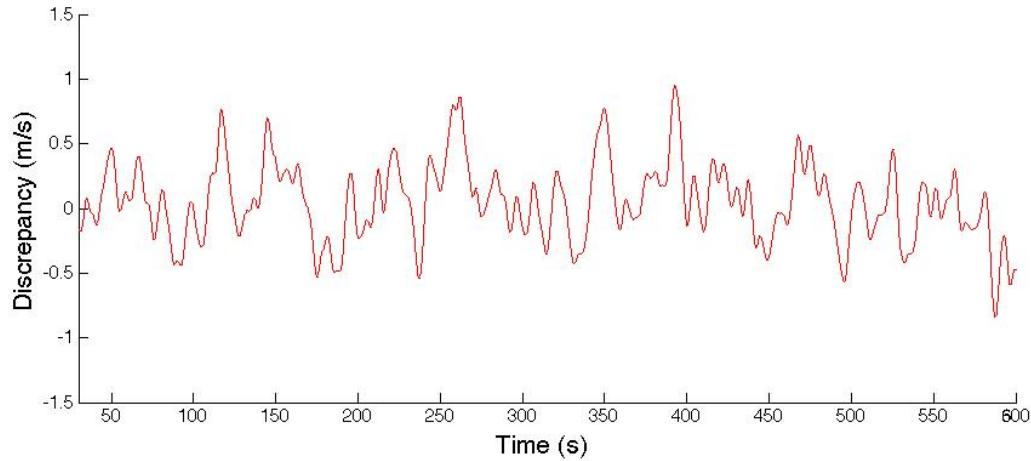


Figure 2.8. Discrepancy between anemometer and rotor average wind speed for simulation #13.

Repeating that process for all the mean wind speeds simulated in the data set yields Figure 2.10 and Figure 2.11. Figure 2.10 shows average magnitudes of the discrepancies between anemometer wind speed and rotor average wind speed in m/s, while Figure 2.11 shows the same quantity expressed as a percentage the mean wind speed. Each * on the plot represents an average over all 57 minutes of simulation data at the corresponding wind speed. Figure 2.10 shows that the smallest

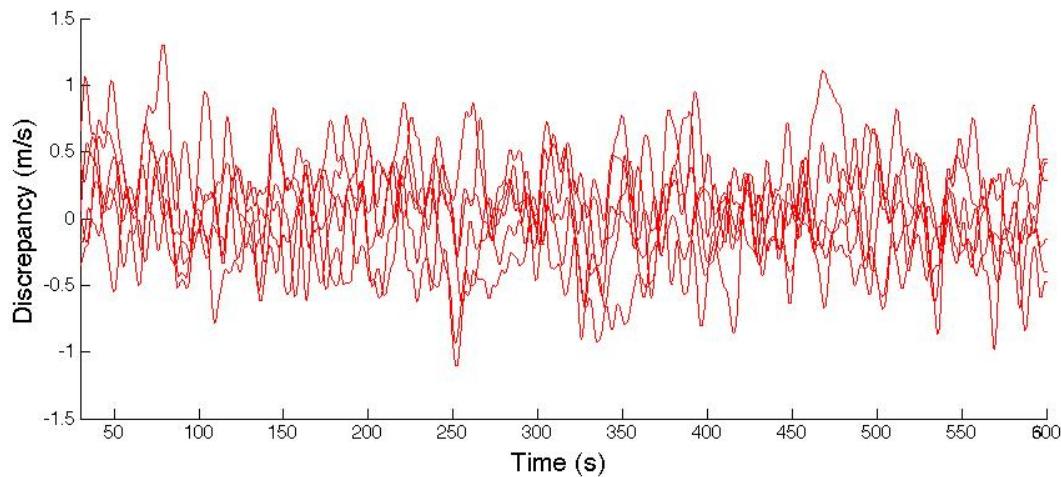


Figure 2.9. Discrepancy between anemometer and rotor average wind speed for all 8 m/s simulations.

average discrepancy is 0.269 m/s, which occurs at a mean wind speed of 6 m/s. When we examine the discrepancy as a percentage of the mean free stream wind speed (Figure 2.11) we see that the average discrepancy is approximately 3% of the mean wind speed over most of the operating range of the NREL 5-MW turbine, though the average discrepancy is a larger percentage at low wind speeds.

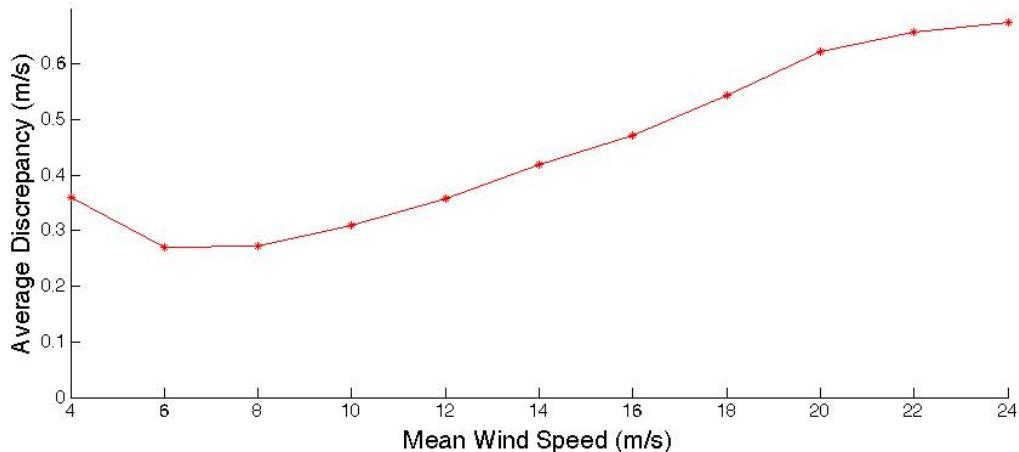


Figure 2.10. Average magnitude of discrepancy (in m/s) as a function of simulation mean wind speed.

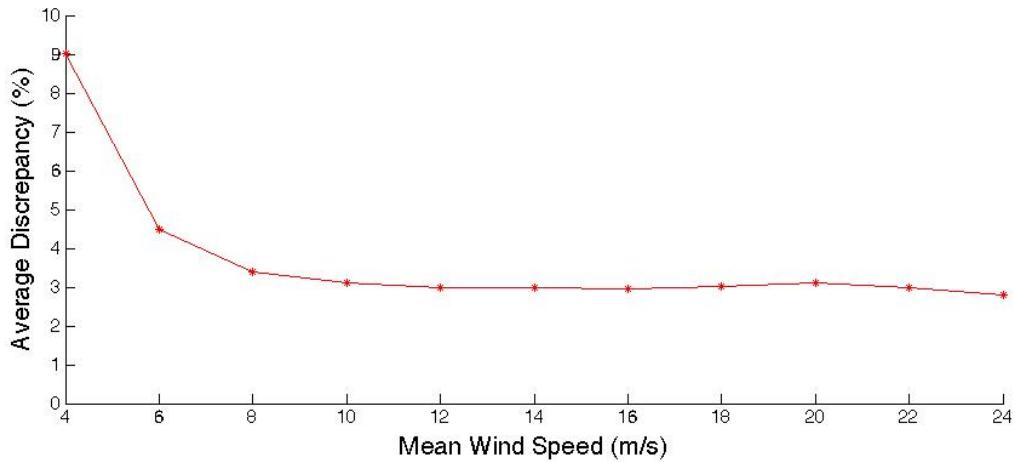


Figure 2.11. Average magnitude of discrepancy (in %) as a function of simulation mean wind speed.

2.4 Turbine Model Based Wind Speed Estimator

Incoming wind speed can also be estimated using an inverted dynamic model of the turbine. First the generator speed, generator torque, and pitch angle are measured. The inverted model is then used to determine the mean incoming wind speed that would cause the observed turbine operating conditions. In effect, the turbine itself is used as a wind speed sensor.[15–17]

We begin with a simplified model of the wind turbine in which the only degrees of freedom are the rotational speed of the rotor (Ω_{Rotor}), the collective blade pitch (θ), and the torque supplied by the generator (T_{Gen}). The dynamics of the turbine are given by equation 2.1, where $I_{Drivetrain}$ is the low speed shaft equivalent inertia, T_{Aero} is the aerodynamic torque on the rotor, and N_{Gear} is the gear box ratio:

$$\dot{\Omega}_{Rotor} I_{Drivetrain} = T_{Aero} - T_{Gen} N_{Gear} \quad (2.1)$$

Essentially, equation 2.1 says the rotational acceleration of the rotor multiplied by the effective inertia of the rotor is equal to the sum of the torques applied to the rotor. The low speed shaft equivalent inertia ($I_{Drivetrain}$) is the rotational inertia of the rotor plus the apparent inertia due other components of the drivetrain. It can be calculated from the rotational inertias of the drivetrain components and the

gearbox ratio (N_{Gear}). There are two torques applied to the rotor. The torque exerted by the generator is equal to the generator torque multiplied by the gearbox ratio. The aerodynamic torque exerted on the rotor by the incoming wind (U) is given by equation 2.2, where ρ is air density, R is the rotor radius, C_p is the coefficient of power, and λ is the tip speed ratio:

$$T_{Aero} = \frac{1}{2}\rho R^3 \frac{C_p(\lambda, \theta)}{\lambda} U^2 \quad (2.2)$$

The tip speed ratio λ is the ratio between the speed of the turbine blade tips and the speed of the incoming wind. It can be calculated using equation 2.3. The coefficient of power C_p is a non-linear function of both tip speed ratio and collective blade pitch angle. A method for determining C_p will be discussed below

$$\lambda = \frac{\Omega_{Rotor} R}{U} \quad (2.3)$$

Equation 2.2 has four quantities that are not directly measurable: T_{Aero} , C_p , λ , and U . If equation 2.1 is used to substitute for T_{Aero} equation 2.2 becomes:

$$\dot{\Omega}_{Rotor} I_{Drivetrain} + T_{Gen} N_{Gear} = \frac{1}{2}\rho R^3 \frac{C_p(\lambda, \theta)}{\lambda} \left(\frac{\Omega_{Rotor} R}{\lambda} \right)^2 \quad (2.4)$$

Which can be re-arranged to yield equation 2.5. All the terms on the right side of equation 2.5 are known or easily measured from sensors on the wind turbine. Therefore, equation 2.5 can be used to calculate C_p/λ^3 .

$$\frac{C_p(\lambda, \theta)}{\lambda^3} = \frac{2\dot{\Omega}_{Rotor} I_{Drivetrain} + 2T_{Gen} N_{Gear}}{\rho R^5 \Omega_{Rotor}^2} \quad (2.5)$$

The coefficient of power C_p is a non-linear function of both tip speed ratio and collective blade pitch angle. It isn't possible to calculate C_p exactly, but for a given tip speed ratio and collective blade pitch it is possible to estimate the C_p of a turbine through simulation. Figure 2.12 illustrates how C_p of the NREL 5-MW turbine varies with λ and collective blade pitch angle (in $^\circ$). This plot was generated from a large number of simulations carried out with WT_Perf.[18] Note that the maximum

C_p corresponds to a 0° collective pitch angle and a tip speed ratio of 7.55. The turbine operates near this point when using variable speed control in region 2.

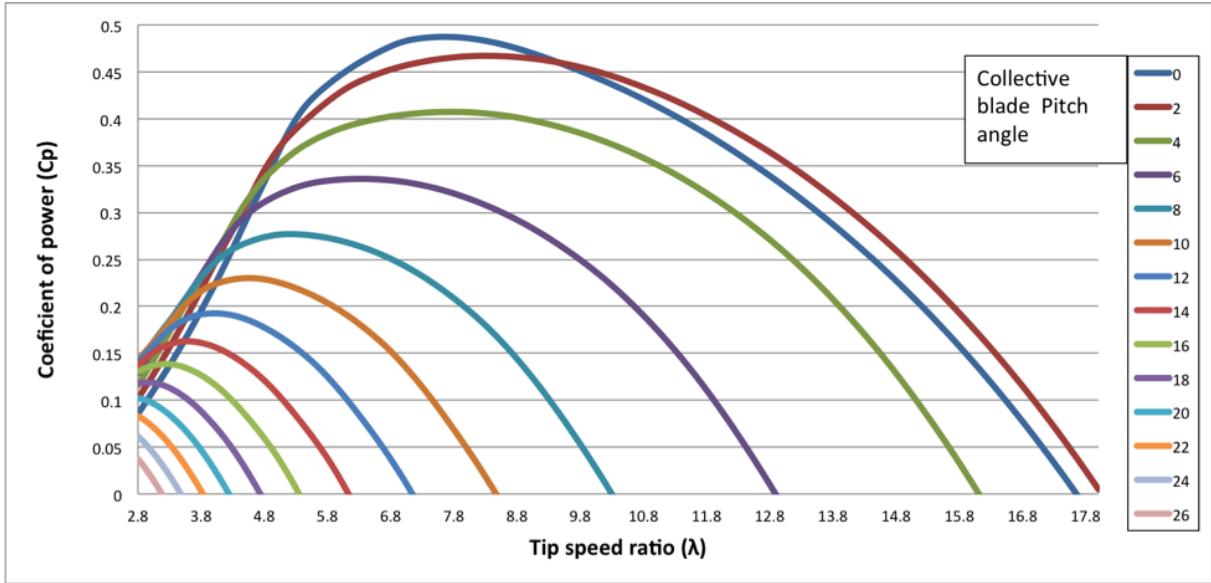


Figure 2.12. C_p dependence on λ and collective blade pitch angle ($^\circ$) for the NREL 5-MW reference rotor.

The collective pitch angle is measured by turbine sensors, but both C_p and λ are unknown quantities that can not be directly measured or easily calculated. However, this problem can be fixed. If each C_p value estimated by WT_Perf is divided by the corresponding λ cubed we get a relationship between $C_p\lambda^3$, λ , and collective pitch angle. Figure 2.13 illustrates how C_p/λ^3 of the NREL 5-MW turbine varies with λ and collective blade pitch angle (in $^\circ$).

The incoming wind speed can now be estimated. First the collective pitch angle is measured and C_p/λ^3 is calculated using equation 2.5. λ is then estimated by interpolating between the WT_Perf simulation data points illustrated in Figure 2.13. Finally, λ is converted to incoming wind speed (U) using equation 2.3.

2.4.1 Feasibility

The feasibility of using a turbine model based wind speed estimator was evaluated using similar methods to those used in Section 2.3.1. We begin by comparing the average wind speed passing through the rotor of the turbine (calculated from the

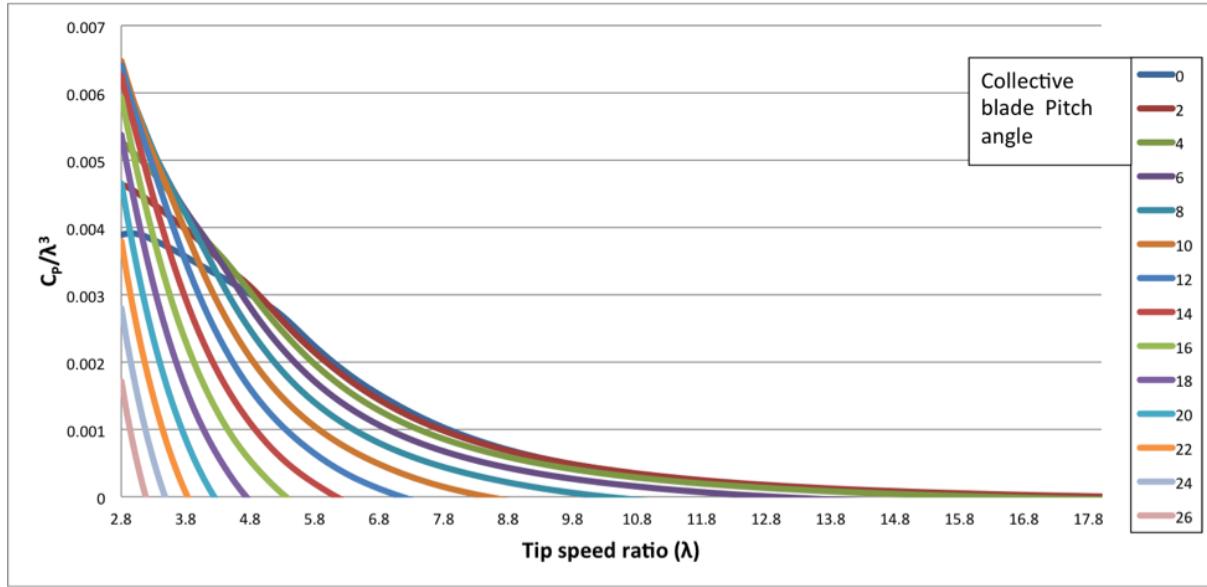


Figure 2.13. C_p/λ^3 dependence on λ and collective blade pitch angle ($^\circ$) for the NREL 5-MW reference rotor.

TurbSim flow field) to the wind speed estimated by the model based wind speed estimator. This gives us a feel for how well the estimator represents the wind passing through the rotor of a utility scale turbine.

The data in Figure 2.14 is from the 13th simulation in the data set described above (the same simulation shown in Figure 2.5), but this data is typical of the other 65 simulations as well. The figure shows that the wind speed estimator is “noisier” than the rotor average wind speed. In this way it’s similar to the nacelle anemometer measurements shown in Figure 2.5. However, the high frequency fluctuations in the wind speed estimator are smaller than the high frequency fluctuations in the anemometer measurements.

Filtering the data in Figure 2.14 to remove the high frequency fluctuations yields Figure 2.15. The wind speed estimator doesn’t match the rotor average wind speed perfectly, but a comparison between Figure 2.7 and Figure 2.15 shows that the wind speed estimator does a much better job than the nacelle top anemometer. This conclusion is reinforced by comparing Figure 2.8 to Figure 2.16.

The data shown in Figures 2.14-2.16 are from a single simulation, but the data and the conclusions drawn from it are typical of all the simulation data exam-

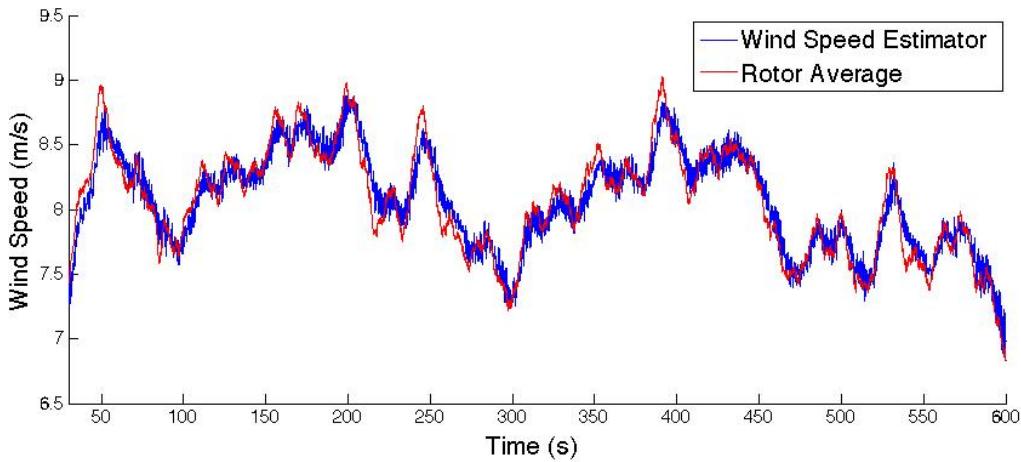


Figure 2.14. Wind speed estimate and rotor average wind speed for simulation #13.

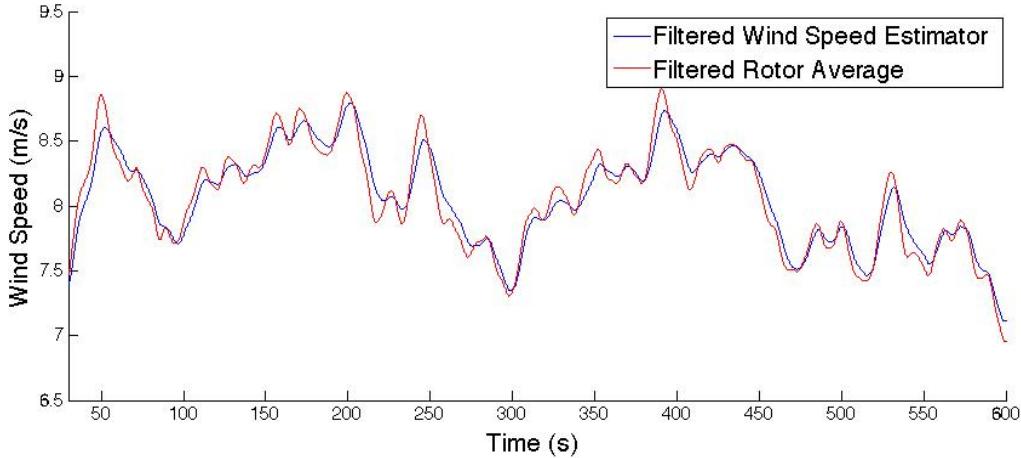


Figure 2.15. Filtered wind speed estimate and rotor average wind speed for simulation #13.

ined. The wind speed estimator performed significantly better than the nacelle top anemometer. If we analyze the wind speed estimator data for all 66 simulations using the same process used to create Figure 2.10 and Figure 2.11 we can get an idea of how much better the wind speed estimator performs. Figure 2.17 and Figure 2.18 compare the average discrepancy between the wind speed estimator and the rotor average wind speed to the average discrepancy between the nacelle top anemometer and the average wind speed. It can be seen from these figures that the wind speed estimator outperforms the anemometer across the entire operating

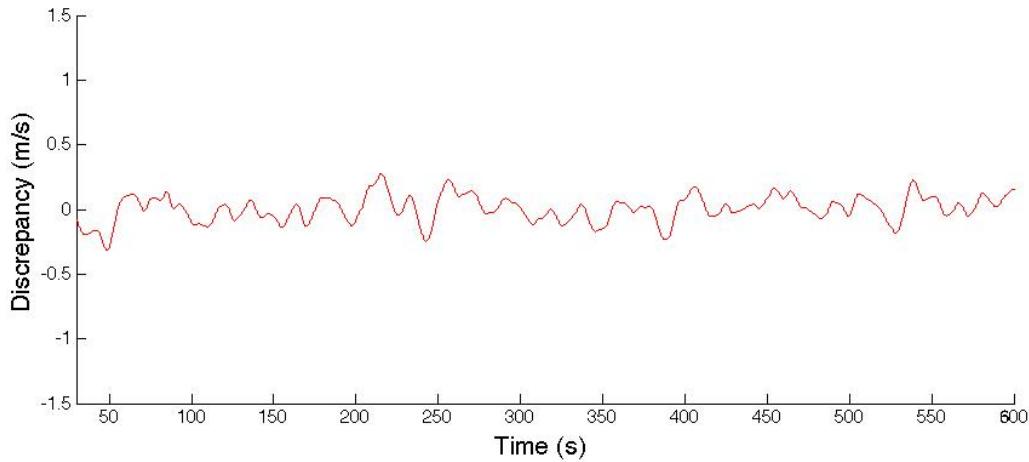


Figure 2.16. Discrepancy between wind speed estimate and rotor average wind speed for simulation #13.

range of the turbine.

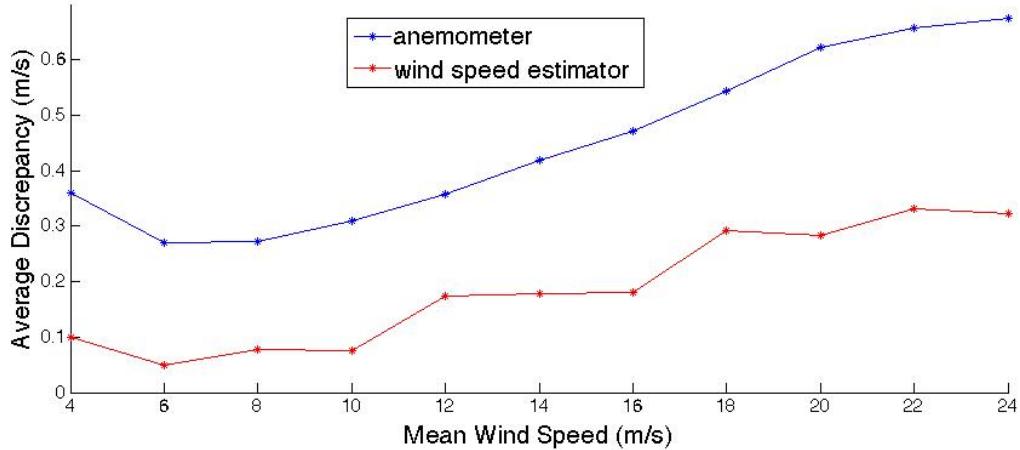


Figure 2.17. Average magnitude of discrepancy as a function of simulation mean wind speed.

2.4.2 Effect of rotor yaw misalignment

The simulations analyzed in the previous sections assume the turbine is pointed directly into the wind. However, in normal turbine operation there is often some misalignment between the wind direction and the pointing direction, or yaw, of the turbine. It is possible to design a turbine that is stable in yaw and will point itself into the wind without the help of a control system or actuators, but in practice this

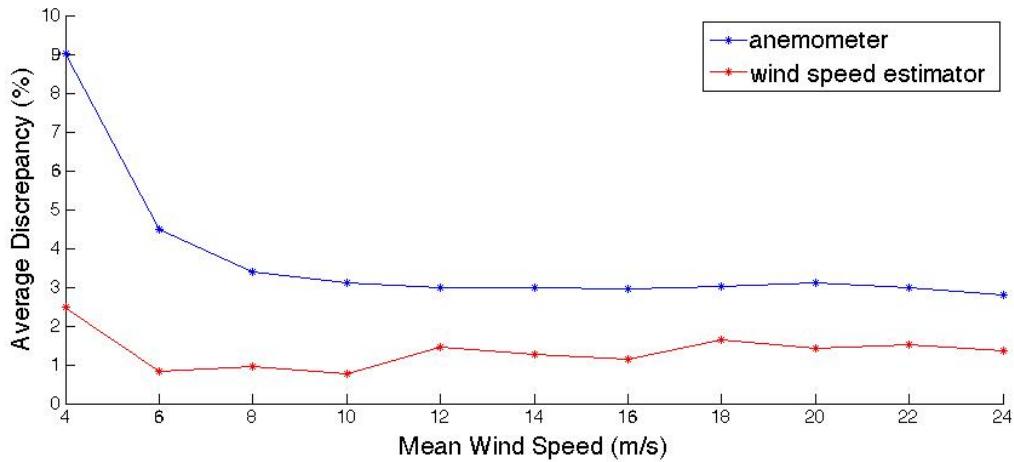


Figure 2.18. Average magnitude of discrepancy as a function of simulation mean wind speed.

passive control technique is only used for very small turbines. Modern utility scale turbines use active yaw control.[9]

The yaw control on a typical utility scale turbine uses a dead band controller with a nacelle mounted wind vane as a sensor and a nacelle yaw motor as an actuator.[9] In this control scheme the signal from the nacelle mounted wind vane is heavily averaged to reduce the effect of small brief changes in wind direction and to reduce the effect of the turbulence generated by the turbine rotor. That averaged wind vane measurement is then treated as an estimate of the wind direction. The dead band controller compares the wind direction estimate to the current yaw of the turbine. If the misalignment is smaller than a pre-defined limit the controller does nothing, but if the misalignment exceeds the limit the controller engages the yaw motor and yaws the turbine into alignment with the wind. Since the NREL 5-MW turbine does not define a yaw control system, we will assume our turbine uses dead band control.

The amount of misalignment allowed by the dead band yaw controller will vary from one turbine to another. To evaluate the effect of yaw misalignment on wind speed estimation, misalignments of 5°, 10°, 15°, and 20° were simulated. For each of these yaw misalignments the 66 test cases described above were simulated.

Figure 2.19 shows the effect of rotor misalignment on wind speed estimation for simulation case #13. As misalignment angle increases the wind speed estimator

underestimates wind speed. For 5° and 10° misalignments this effect appears small, but the effect is much larger for 15° and 20° misalignments. It's worth noting that the shapes of the wind speed estimate curves do not change much with increasing rotor misalignment. The misalignment primarily affects the magnitude of the wind speed estimates.

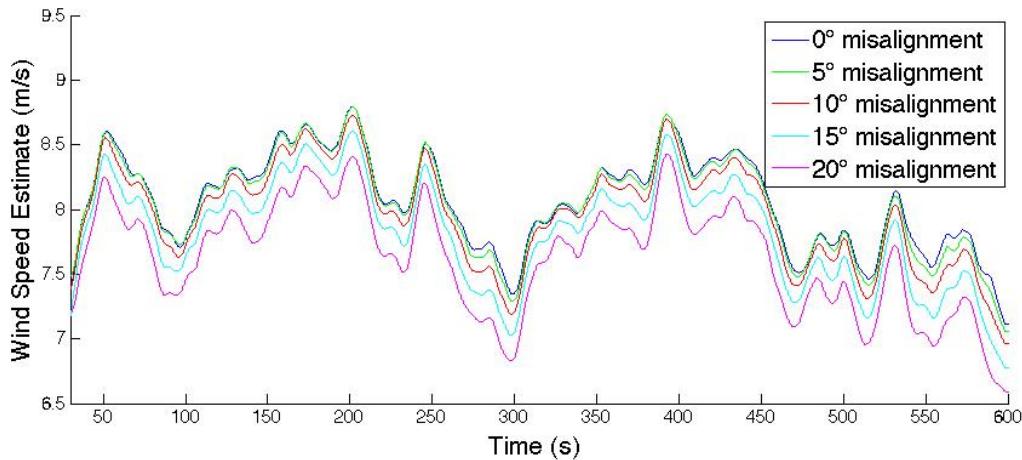


Figure 2.19. Effect of rotor misalignment on wind speed estimate for simulation #13.

Though the data shown in Figure 2.19 is from a single simulation case the conclusions drawn from it are typical of all simulation data that was examined. If we analyze the wind speed estimator data for all simulations cases using the same process used to generate Figure 2.11, we can get an idea of how rotor misalignment affects wind speed estimation error. As we saw in Figure 2.19 a misalignment of 5° or 10° causes only a small increase in the discrepancy between estimated wind speed and average rotor wind speed. Larger misalignments cause significantly larger discrepancies. It is worth noting only the 20° misalignment performs worse than the nacelle top anemometer (Figure 2.11).

It should not be surprising that a misaligned rotor underestimates wind speed. Our wind speed estimator is based on a simplified model of turbine rotor dynamics. Since wind parallel to the rotor plane does not exert an aerodynamic torque on the rotor our estimator does not measure that component of the wind. Essentially, our wind speed estimator is designed to measure the component of the wind that is

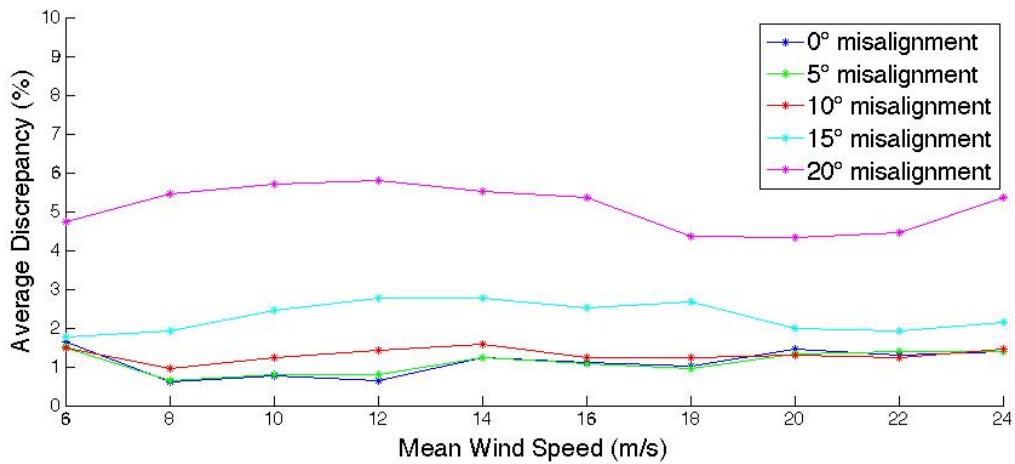


Figure 2.20. Effect of misalignment on average magnitude of wind speed estimate discrepancy.

perpendicular to the rotor plane. As Figure 2.21 shows, the apparent wind seen by the wind speed estimator is equal to the actual wind multiplied by the cosine of the misalignment angle.

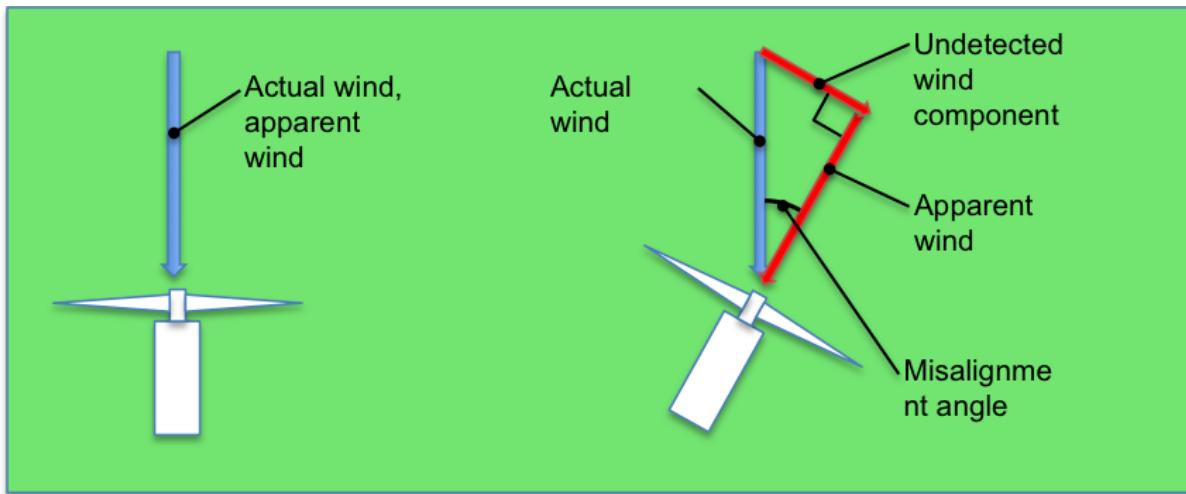


Figure 2.21. Effect of rotor misalignment on apparent wind.

If the misalignment angle is known, it can be used to compensate for the difference between the apparent wind seen by the estimator and the actual wind. If the wind speed estimates shown in Figure 2.19 are divided by the cosine of the misalignment angle it yields Figure 2.22. As the figure shows, the effect of misalignment on wind speed estimation has been greatly reduced.

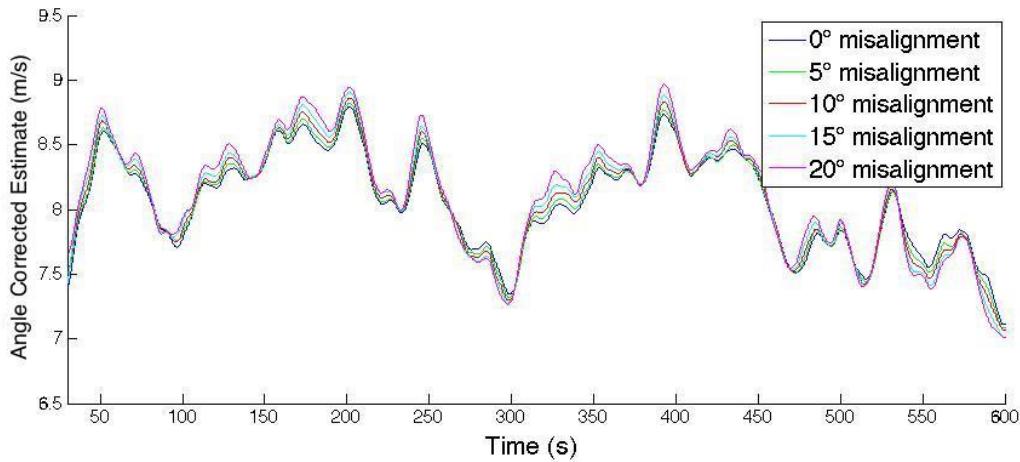


Figure 2.22. Angle corrected wind speed estimates for simulation #13.

Figure 2.23 shows the average discrepancy between misalignment-compensated wind speed estimates and rotor average wind speed. The misaligned simulations generally show more discrepancy than the perfectly aligned simulations. However, comparing Figure 2.20 to Figure 2.23 shows that misalignment compensation greatly reduces the discrepancy. In the 20° misalignment simulations the average discrepancy is reduced 60-75% by angle compensation.

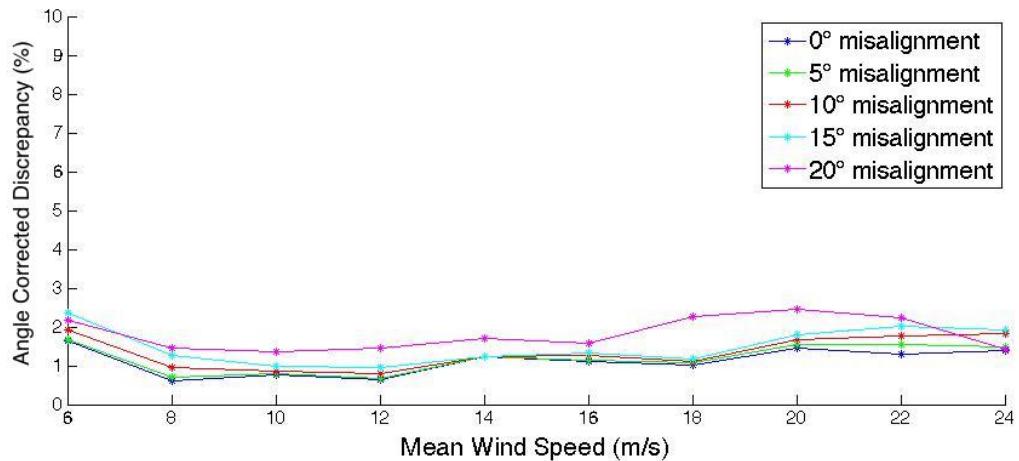


Figure 2.23. Average magnitude of discrepancy in angle corrected wind speed estimates.

Though the misalignment angle will not be perfectly known, an estimate of the misalignment angle will be available and can be used for compensation. As stated

earlier in this section, the dead band yaw controller continuously estimates the misalignment angle but doesn't act unless the misalignment exceeds a pre-defined limit.

Yaw misalignment does have an effect on turbine model based wind speed estimation, but it does not appear that it will be a significant hindrance to using turbine model based wind speed estimation. Misalignments less than 10° only have a minor effect on performance. Larger misalignments do have a significant effect on performance, but it should be possible to reduce the effect by using misalignment angle compensation.

2.5 Convection speed estimation

The previous sections have discussed instantaneous wind speed estimation. These wind speed estimates can be used to observe how the wind speed is fluctuating as it passes through a wind turbine rotor. However, knowledge of how the wind speed is fluctuating is not sufficient information for feed forward control of downwind turbines. We must also know when these wind speed fluctuations will reach the downwind turbine. The time required for the wind speed fluctuations to pass from the upwind turbine to the downwind turbine can be calculated with the simple equation $t = D/U_{conv}$, where D is the downwind distance between the turbines and U_{conv} is the convection speed. The convection speed is the rate at which wind speed fluctuations propagate downwind. This section will discuss using a wind turbine to estimate the convection speed.

If we assume Taylor's frozen turbulence hypothesis we can estimate the convection speed based on wind speed measurements taken at a single location. In this section the convection speed will be estimated using the turbine model based wind speed estimates discussed in Section 2.4. Taylor said "If the speed of the air stream which carries the eddies is very much greater than the turbulence speed, one may assume that the sequence of changes in u at a fixed point are simply due to the passage of an unchanging pattern of turbulent motion over the point." [19]

In other words, the wind experienced by a point downwind of our sensor will be the same as the wind experienced by the sensor but it will be delayed by a time $t_D = (\text{distance})/(U_{\text{conv}})$. It follows from Taylor's hypothesis that the convection speed of the turbulent fluctuations is equal to the local mean speed (U_{mean}).

There is much research into the validity and limitations of Taylor's frozen turbulence hypothesis for estimating the convection speed.[20–23] Though research shows that the convection speed is not always equal to the mean speed they are frequently assumed to be equal in turbulent flow research. $U_{\text{conv}} = U_{\text{mean}}$ will be assumed for the remainder of this section. A more accurate estimate of the convection speed could be generated using more than one wind speed measurement location. For example, we could measure the time it took a gust of wind to travel between two turbines. However, this technique can not be used with the data set described in Section 2.2 because FAST only simulates a single turbine.

For the data set described in Section 2.2 the mean speed (U_{mean}) is known. U_{mean} is one of the TurbSim input parameters used to generate the data set. As shown in equation 2.6 we can consider the wind passing through the turbine rotor to be a combination of the underlying mean wind speed (U_{mean}) and the wind speed fluctuations due to turbulence (U_{turb}). If we average the hub height wind speed over a long enough time the wind speed fluctuations due to turbulence will cancel out and we'll be left with the mean wind speed.

$$U = U_{\text{mean}} + U_{\text{turb}} \quad (2.6)$$

Figure 2.24 shows the wind speed estimate, the sixty second average wind speed, and the true mean wind speed for simulation #13. The wind speed estimate is generated using the method described in section 2.4. 60 second average values are generated by averaging 60 seconds of wind speed estimate data (from 30 seconds before to 30 seconds after the time of interest). The true mean wind speed is known and was specified as an input when simulating the incoming wind. There are two things worth noting in Figure 2.24. First, there are significant discrepancies be-

tween the 60 second average wind speed and the known mean wind speed of 8 m/s. At 182 seconds there is a 0.587 m/s, or 7.3%, discrepancy between the 60 second average wind speed and the true mean wind speed. Also, the 60 second average wind speed out performs the wind speed estimator when it comes to estimating the true mean wind speed. Both the maximum and average discrepancy are lower for the 60 second average wind speed.

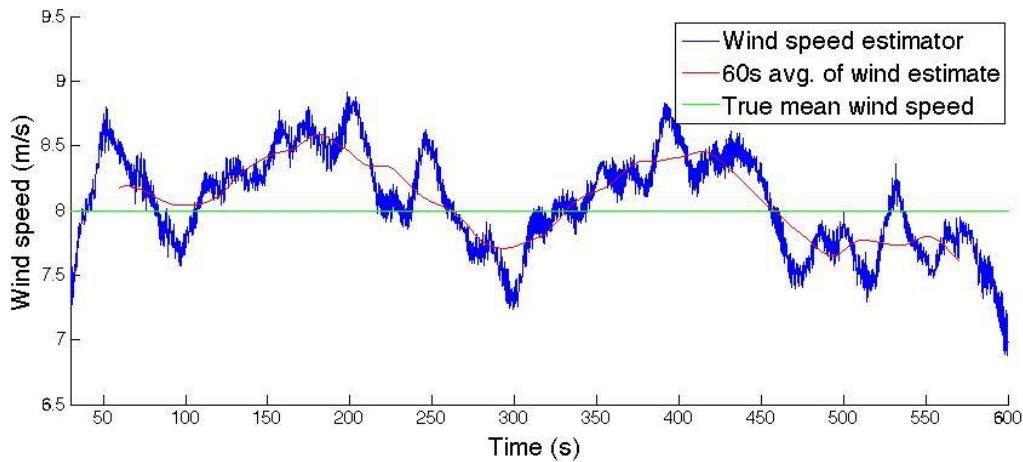


Figure 2.24. Estimating U_{mean} using a 60 s average of wind speed estimates for simulation #13.

By analyzing all of the test cases in our data set with a variety of averaging times we can create Figures 2.25 and 2.26. These figures show the maximum and average discrepancies between the various averaged wind speed estimates and the known true mean wind speeds. In Figure 2.25 we see that longer average times result in smaller mean discrepancies between the average wind speed estimate and the true mean wind speed. This is expected. Longer averaging times would likely perform even better, but 570 seconds was the longest averaging time we could use with the current data set. The highest mean discrepancies (as a % of the true mean wind speed) are at low wind speeds. For some wind speeds (16 m/s - 22 m/s) Figure 2.25 seems to show diminishing improvement for averaging times over 240 s. However, for the rest of the wind speeds Figure 2.25 does not seem to show a clear point of diminishing returns.

Figure 2.26 largely shows the same trends that we see in Figure 2.25, but the

trends are a little less clear. Figure 2.26 shows a few unexpected results. For example, in 18 m/s wind Figure 2.26 an averaging time of 240 s gives a larger maximum discrepancy than any of the shorter or longer averaging times. I suspect that if we analyzed a lot more data some of these unexpected results would no longer be present.

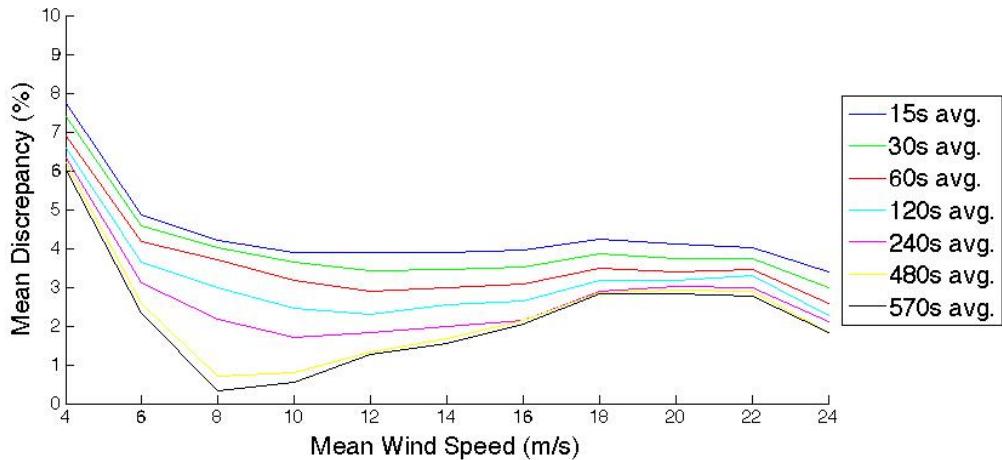


Figure 2.25. Mean discrepancy between true U_{mean} and estimated U_{mean} for a variety of averaging times.

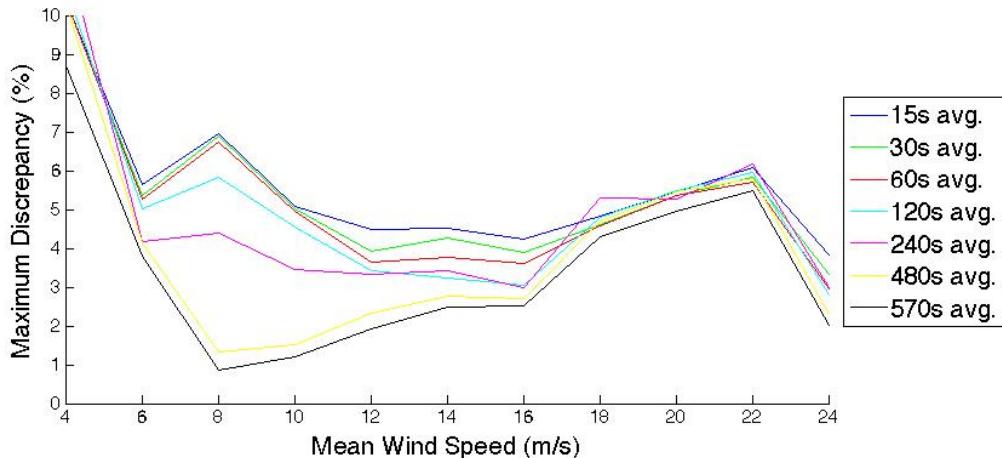


Figure 2.26. Max discrepancy between true U_{mean} and estimated U_{mean} for a variety of averaging times.

Though Figures 2.25 and 2.26 imply that the averaging time should be as long as possible, there is a practical reasons why a very long averaging time would be undesirable. When we simulate wind fields using TurbSim the mean wind speed

remains constant, but research has shown that in real world conditions the mean wind speed does not remain constant indefinitely. In real world conditions the mean wind speed may be slowly changing.[24] If the mean wind speed is changing, we can minimize the impact on our wind speed averaging by keeping the averaging time short and by centering the averaging window on the time of interest. If we choose to center our averaging window on the time of interest, our averaging time is further constrained by the amount of time it will take wind speed fluctuations to propagate from the upwind turbine to the downwind turbine. For example, if the mean wind speed is 20 m/s and the downwind turbine is 1260 m ($10 \times$ the NREL 5-MW rotor diameter) behind the upwind turbine, it will take about 63 seconds for the wind speed fluctuations to travel from the upwind turbine to the downwind turbine. In that scenario, the averaging time must be less than 126 seconds.

2.5.1 A few thoughts on the limitations of Taylor's Frozen Turbulence Hypothesis

Though Taylor's hypothesis is often invoked in investigations of atmospheric turbulence, experimental results have shown that it is not universally accurate. In particular, Taylor's hypothesis is more accurate over short distances than over long distances and it is more accurate for low frequency wind speed fluctuations than high frequency fluctuations.[25, 26]

2.6 Other Potentially Useful Turbine Sensors

The methods discussed in previous sections aren't the only ways to use a turbine as a sensor for feed forward control. A typical turbine has other sensors that are potentially useful for feed forward control. This section briefly discusses two of those sensors, the wind vane and the rotor speed sensor.

Utility scale turbines are equipped with wind vane's that measure wind direction. Wind direction measurements are not used in this dissertation. The feed forward control schemes developed in Chapter 3 and Chapter 4 are simulated in a two turbine system where wind direction is treated as a known quantity. However, in

a real world application of these feed forward control systems, wind direction measurements would be critical. In a wind farm each turbine would have many other turbines it could share information with. Wind direction would tell each turbine which other turbines it should send information to and which turbines it should receive information from. In many wind farms, wind direction changes seasonally or over the course of each day. As wind direction changes, wind vane readings would help the wind farm adapt and adjust how turbines share data for feed forward control.

Section 2.4 shows that rotor speed can be used with blade pitch and knowledge of turbine dynamics to estimate wind speed. However, rotor speed also has the potential to be useful on its own. In region 3 control, the NREL 5-MW turbine attempts to track a constant rotor speed of 12.1 RPM. Any deviation from 12.1 RPM indicates a change in wind speed. Larger and/or more abrupt changes in wind speed result in larger deviations from 12.1 RPM. Since large abrupt changes in wind speed result in large structural loads on the turbine, there is a correlation between how much the rotor deviates from 12.1 RPM and how much gust induced structural loading the turbine experiences. Monitoring rotor speed could potentially be used as a simple way to identify the most damaging gusts. For example, a feed forward control system could consider anything causing a rotor speed over 13.31 RPM (10% higher than the rated speed of 12.1 RPM) an extreme gust event. Whenever evidence of an extreme gust event is observed, the feed forward control system could take some kind of action to prepare downwind turbines for the gust.

2.7 Summary and Conclusions

This chapter investigates ways to use a turbine as a sensor. The focus is on gathering data that could be useful for feed forward control. To ensure low cost, we only consider techniques that rely on sensors a modern utility scale turbine would already have.

Section 2.2 describes a set of 66 turbulent wind fields. Those wind fields are

used in subsequent sections to evaluate sensor techniques.

Section 2.3 discusses wind speed estimates based on nacelle top anemometer readings. First the limitations of anemometer measurements are discussed. Next, wind speed measurements at the rotor center are compared to the average wind speed over the rotor area. The discrepancies between those measurements are analyzed for the 66 turbulent wind fields described in Section 2.2.

Section 2.4 discusses wind speed estimates based on an inverted dynamic model of the turbine. The governing dynamic equations are derived. Those dynamic equations are then combined with WT_Perf simulations of the NREL 5-MW turbine to develop a data sets that can be interpolated to estimate wind speed from tip speed ratio and collective blade pitch. Next, wind speed estimates based on turbine dynamics are compared to the average wind speed over the rotor area. The discrepancies between those measurements are analyzed for the 66 turbulent wind fields described in Section 2.2. This wind speed estimation technique is found to provide more accurate wind speed measurements than the anemometer. The turbine dynamics based estimator has smaller average discrepancies and smaller maximum discrepancies with the average wind speed over the rotor area. Section 2.4.2 discusses the effect of yaw misalignment on turbine dynamics based wind speed estimation and how that effect could be compensated for.

Section 2.5 Discussed convection velocity and how to estimate it from wind speed estimates at a single location.

Section 2.6 discusses other turbine sensors, such as wind vanes or rotor speed sensors, that could be useful for feed forward control.

Chapter 3

Feed Forward Optimum Pitch Control

3.1 Introduction

As discussed in Section 1.2, the closed loop control system of a typical utility scale wind turbine manipulates generator torque and collective blade pitch in order to control the power generation and rotor speed of the turbine. At low and medium wind speeds the blade pitch remains fixed at 0° while generator torque is manipulated to achieve the rotor speed that maximizes electricity generation. At high wind speeds the collective blade pitch is manipulated to manage loading on the wind turbine as well as to maintain constant rotor speed and power generation. For a given wind speed there is a desired rotor speed (Ω_{Rotor}), collective blade pitch (θ), and generator torque (T_{Gen}). Figure 3.1 shows these desired values for the NREL 5-MW turbine. If the wind remained constant the control system of the NREL 5-MW turbine would bring Ω_{Rotor} , θ , and T_{Gen} to the values shown in Figure 3.1 and maintain those desired conditions. However, since wind speed often fluctuates (as shown in Figure 3.2) and utility scale turbines are large mechanical devices that are unable to instantaneously adjust to changes in wind speed, turbines are frequently chasing the desired conditions instead of operating at the desired conditions.

Figures 3.3 and 3.4 illustrate the transient response of the NREL 5-MW turbine when subjected to sudden changes in wind speed. In both cases FAST has been

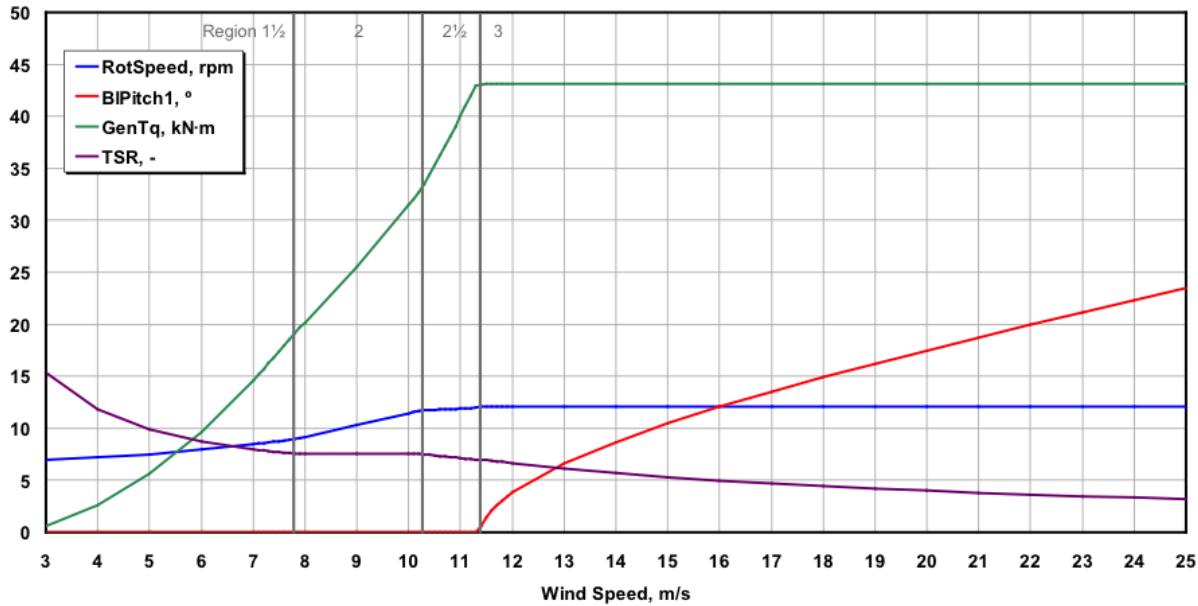


Figure 3.1. Steady state behavior of the NREL 5-MW.[2]

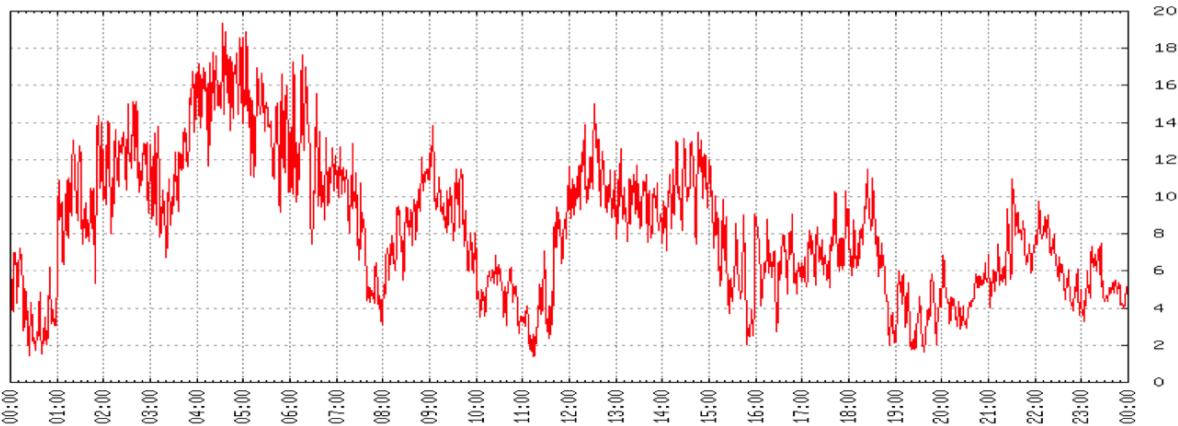


Figure 3.2. An example of wind speed fluctuations over 24 hours.[6]

used to model a NREL 5-MW turbine with the baseline controller defined in.[2] Figure 3.3 illustrates turbine behavior in region 2, wind speeds between 7.8 m/s and 10.3 m/s, while Figure 3.4 illustrates turbine behavior in region 3, wind speeds between 11.4 m/s and 25 m/s. Each figure shows wind speed, “Actual” and “Optimal” power, generator torque, and blade pitch. “Actual” power is the power generated by the turbine. “Optimal” power is the desired steady state power generation for the current wind speed (based on the NREL 5-MW power curve shown in Figure 1.6). Generator torque and blade pitch are the two actuators used to control turbine

performance.

In Figure 3.3 the turbine is subjected to a uniform incoming wind that changes between 8 m/s and 10 m/s every 20 seconds. These wind speeds are in region 2, where optimal power production corresponds to the maximum aerodynamic efficiency of the turbine. Note that the turbine takes approximately 20 seconds to reach optimal generation after each change in wind speed. This corresponds to a reduction in efficiency. The optimal power curve generates 0.6% more power (15.6 kW on average) than the actual power curve. In this region 2 case turbine performance is regulated entirely by torque control and the turbine blades remain at 0° pitch.

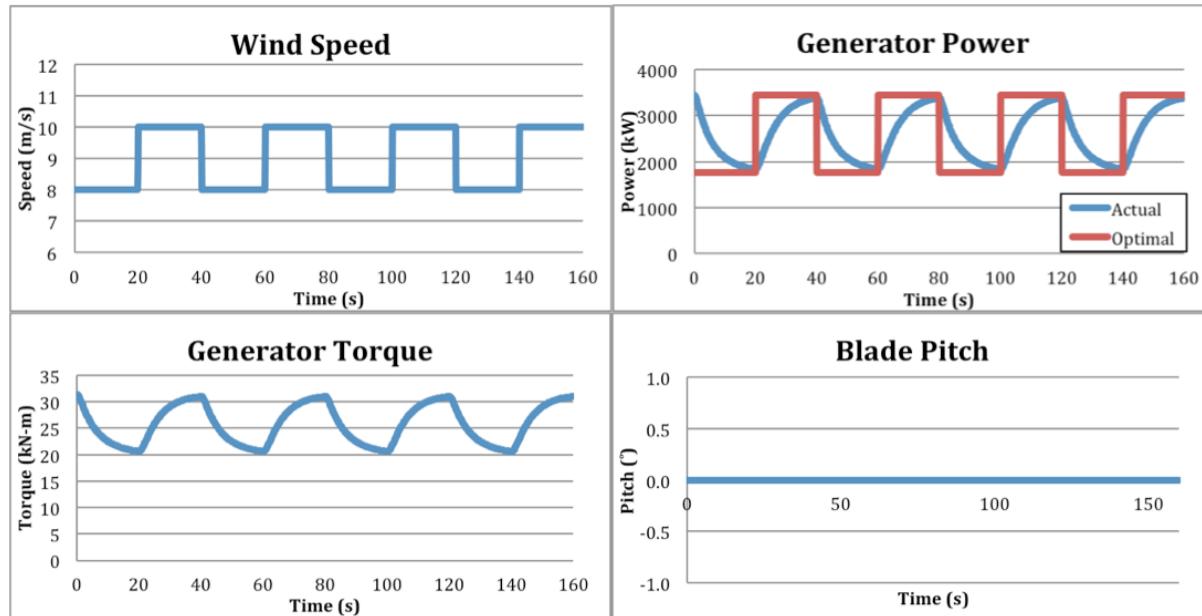


Figure 3.3. NREL 5-MW response to wind speed fluctuations in control region 2.

In Figure 3.4 the turbine is subjected to a uniform incoming wind that changes between 16 m/s and 18 m/s every 40 seconds. These wind speeds are in region 3, where optimal power production is the rated power of the turbine, 5 MW. After each change in wind speed the turbine experiences both spikes and dips in power production. The largest fluctuations in power occur in the first 10 seconds after a change in wind speed. After 25 seconds the generated power is nearly constant at 5 MW. In this case turbine performance is regulated by both torque control and pitch

control.

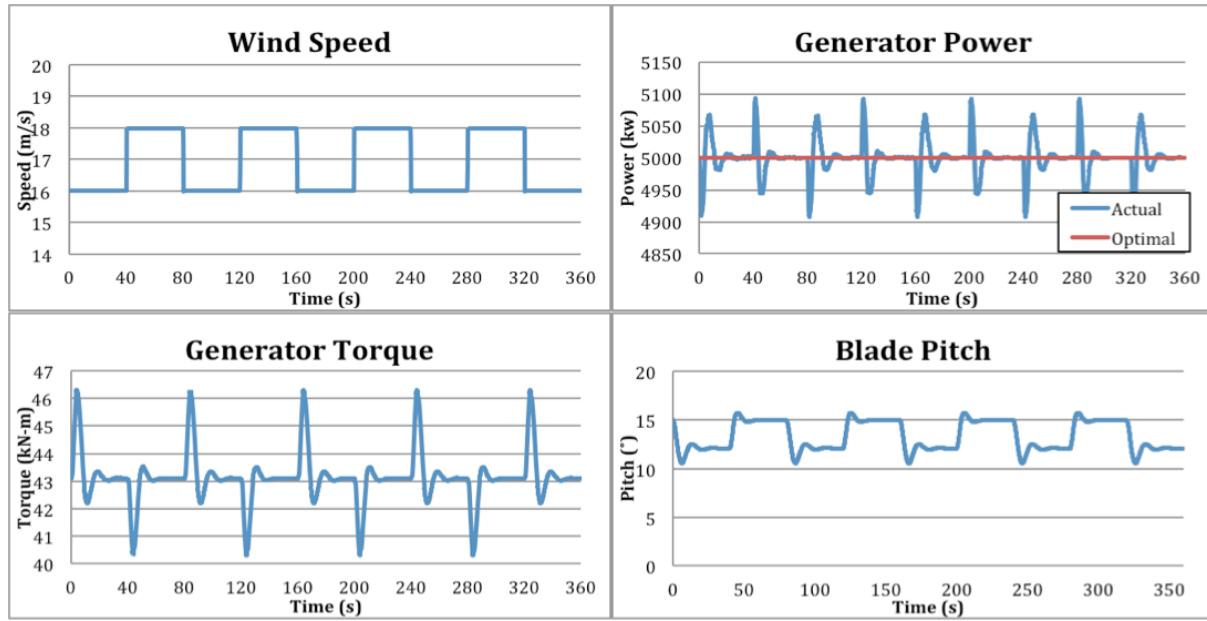


Figure 3.4. NREL 5-MW response to wind speed fluctuations in control region 3.

The previous figures show that wind turbines respond reactively to changes in wind speed. However, several researchers have shown that turbine performance can be improved if a turbine is able to pre-emptively react to changes in wind speed before they occur. Ozdemir, Seiler, and Balas showed that perfect preview knowledge of incoming wind speed could be used to increase power production or decrease gearbox loads by improving generator torque control in region 2 operation. The authors used state space optimal control techniques to control generator torque based on incoming fluctuations in wind speed. In some of their simulations they were able to achieve increases in power generation of 6% or decreases in gearbox loading of 30%. [27] Dunne, Pao, Wright, Jonkman, Kelley, and Simley have shown that perfect preview knowledge of incoming wind speed fluctuations could also be used to reduce turbine loads by improving pitch control in region 3 operation. The authors examined three methods of feed forward pitch control based on wind speed preview information: non-causal series expansion, preview control, and optimized FIR filter methods. The authors simulated the response of the feed forward controllers to both turbulent wind and sudden gusts. Each of the feed forward controllers were

capable of reducing damage equivalent loads on the turbine blades and/or turbine tower.[28] Schlipf, along with various collaborators, has shown that imperfect preview knowledge of incoming wind speed, obtained through a nacelle mounted LIDAR, can be used to improve turbine control.[4, 26, 29–31] Schlipf, and Kühn used simulations of the LIDAR and turbine to demonstrate that LIDAR based feed forward collective pitch control could reduce extreme and fatigue loads on turbine blades and the turbine tower.[31] However, Schlipf, et al. concluded that only minimal increases in power production could be achieved using LIDAR assisted generator torque control in region 2 operation.[30] Scholbrock, et al. field tested a LIDAR assisted feed forward controller on the NREL CART3 research turbine. The authors observed reductions in damage equivalent loads for the turbine blade bending moments and rotor torque.[32]

The remainder of chapter examines the feasibility of feed forward pitch control using wind speed preview information generated by using wind speed estimates from an upwind turbine. The following section describes how FAST, a single turbine simulation tool, can be used to simulate a two turbine system. Section 3.3 describes the feed forward controller. Sections 3.4 and 3.5 examine how the feed forward controller performs when subjected to wind gusts and turbulent wind. Section 3.6 examines how sensitive the feed forward controller is to errors in feed forward wind speed estimate data and the corresponding implications.

3.2 Using FAST to Model a Two Turbine System

As discussed in Section 2.2, FAST is a medium fidelity wind turbine simulation tool. FAST models both the aerodynamics and structural dynamics of a single turbine. FAST can model a turbine’s response to either uniform incoming wind or, with the help of TurbSim, statistically accurate turbulent wind. Though FAST only models one turbine at a time it can be used to model a multi turbine system if a few assumptions are made. Figure 3.5 shows the two turbine system being simulated. The wind is blowing from left to right, the terrain is flat, and the downwind turbine

is slightly offset from the upwind turbine. When simulating this two turbine system in FAST the following assumptions are made:

- Taylor's hypothesis is valid. The wind speed fluctuations experienced by the upwind turbine will propagate downwind at some convection velocity without changing. Therefore, the downwind turbine will experience the same series of wind speed fluctuations as the upwind turbine but at a later time.
- The slight offset of the downwind turbine will be large enough to keep the downwind turbine out of the upwind turbine's wake.
- The slight offset of the downwind turbine will be small enough that wind speed fluctuations will hit both the upwind and downwind turbines.
- Because the terrain is flat, complex terrain effects do not need to be considered.



Figure 3.5. Illustration of the two turbine system modeled in this chapter.

Figure 3.6 is a block diagram illustrating how FAST can be used to simulate the two turbine system shown in Figure 3.5. First a simulated wind field is generated. This can be a simple wind field, such as uniform incoming wind, or a turbulent wind field generated with TurbSim. Next, FAST is used to simulate how a turbine with a conventional control system will respond to the wind field. This first FAST simulation simulates the upwind turbine in our two turbine system. Once the first FAST simulation is complete, the simulation results are post-processed using the wind speed estimation techniques described in 2.4. In real world applications, data

from the upwind turbine would be continually processed and passed to the downwind turbine. However, since we are simulating the upwind and downwind turbine separately, it is easier to process all of the upwind turbine data at the same time. Finally, FAST is used to simulate the downwind turbine. In this second simulation the turbine experiences the same wind field, but the turbine has feed forward pitch control and has access to wind speed preview data that was generated by post processing the results of the upwind turbine simulation. By comparing the simulation results from the first (upwind) and second (downwind) FAST simulations we can see how the feed forward pitch controller has affected turbine performance.

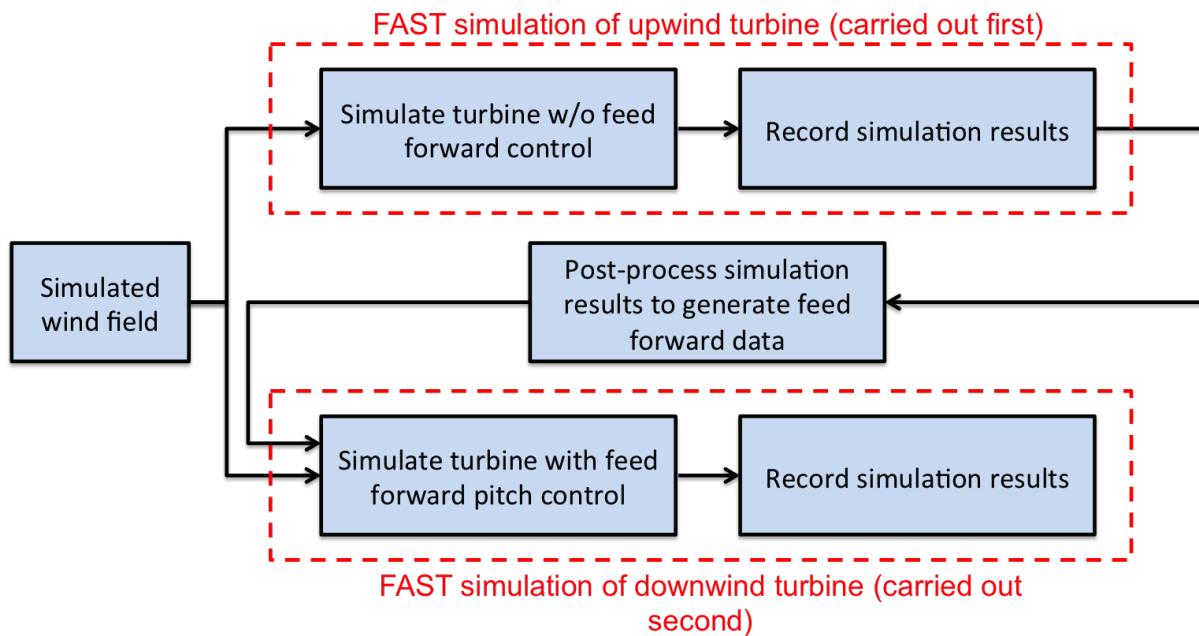


Figure 3.6. The process used to simulate a two turbine system with FAST.

3.3 Controller Design

3.3.1 Implementing Feed Forward Control

Figure 3.7 is a simple block diagram illustrating how the control system, turbine, and wind interact for a typical utility scale turbine. The behavior of the turbine is affected by the wind as well as the collective pitch (θ) and generator torque (T_{Gen}) commands sent by the closed loop controller. The closed loop controller monitors the

turbine rotor speed (Ω_{Rotor}), the only feedback signal used in a typical utility scale turbine controller, to determine if the turbine is operating at a desired steady state operating point. If the turbine is not at a desired operating point the controller manipulates collective blade pitch (θ) and generator torque (T_{Gen}) to bring the turbine to a desired operating point. In region 3 (high wind speeds) the controller tries to keep the turbine rotor speed (Ω_{Rotor}) and power generation constant. In region 2 (low and medium wind speeds) the controller tries to maintain the rotor speed that will maximize power generation. From a control standpoint the wind is a disturbance, it is an uncontrolled input that affects the behavior of the turbine. Generally, changes in the wind cause the turbine to move away from desired operating points while the closed loop controller acts to bring the turbine back to desired operating points.

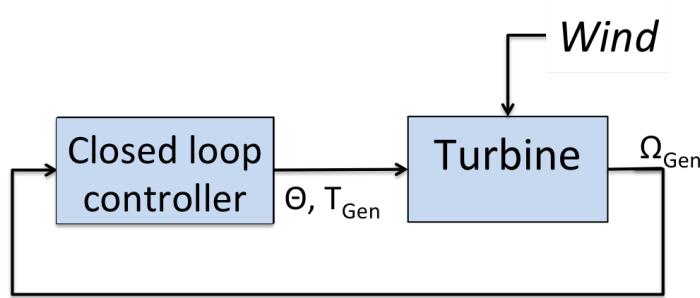


Figure 3.7. Simple block diagram of turbine control w/o feed forward control.

Figure 3.8 is a simple block diagram illustrating the method we have chosen to incorporate feed forward control into the control system of a turbine. You can see that the feedback control loop shown in Figure 3.7 remains intact, but a feed forward controller, which issues supplemental collective pitch (θ_{ff}) and generator torque (T_{ff}) commands has been added. The closed loop controller shown in Figure 3.8 is identical to the one in Figure 3.7. This is not the only way to implement feed forward control, but this method does have some advantages. It is fairly simple in that it does not require us to re-design the entire control system. In addition, if the turbine were to lose the feed forward signal for any reason it would simple begin operating as if it were using a standard closed loop turbine control system.

In fact this feed forward control implementation allows the turbine to smoothly transition between standard closed loop control and closed loop with feed forward control. This trait is desirable for a variety of situations. For example, if the wind changes direction the turbine may find that it no longer has an upwind turbine to use as a sensor.

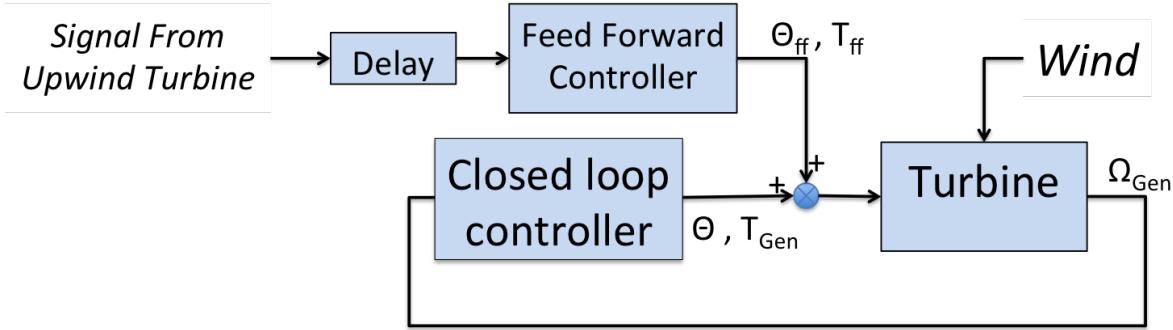


Figure 3.8. Simple block diagram of turbine with feed forward control.

3.3.2 Closed Loop Controller

The closed loop controller is modeled in Simulink and coupled to FAST as described in the FAST user's guide[33], and the full Simulink model of the closed loop controller can be found in appendix C. The controller has all the properties of the NREL 5-MW controller defined by NREL in [2] plus a model of the collective blade pitch actuator. Figure 3.9 illustrates the controller. The NREL 5-MW controller defined in [2] consists of a measurement filter, a torque controller, and a pitch controller. The measurement filter is a recursive, single pole low pass filter with a corner frequency of 0.25 Hz. The purpose of this low pass filter is to mitigate high frequency excitations of the control system. The torque controller uses a torque schedule, basing generator torque on the rotational speed of the turbine. The pitch controller is a non-linear PI (proportional and integral) controller where the proportional and integral gains are a function of blade pitch.

Since this chapter is concerned with feed forward pitch control, the turbine will be operating in control region 3 for all simulations. The NREL 5-MW turbine operates in control region 3 when wind speeds are higher than the turbine's rated wind

speed of 11.4 m/s. In control region 3 the pitch controller varies collective blade pitch in an attempt to maintain a constant rotational rotor speed (ω_{Rotor}) of 12.1 RPM. When the rotor is spinning at 12.1 RPM the generator supplies the rated generator torque (43,093.55 Nm), but when the rotor deviates from 12.1 RPM the torque controller varies torque in an effort to maintain a constant power generation of 5 MW.

The controller shown in Figure 3.9 also includes a model of the collective pitch actuator. Though the pitch actuator is not part of the control system it must be modeled in Simulink because current versions of FAST do not model pitch actuator dynamics. If pitch actuator dynamics are not modeled, the closed loop controller is able to instantly change blade pitch. This is physically unrealistic and in some circumstances it leads to unrealistic control system behavior, such as high frequency fluctuations in pitch or unrealistic reactions to feed forward pitch control.

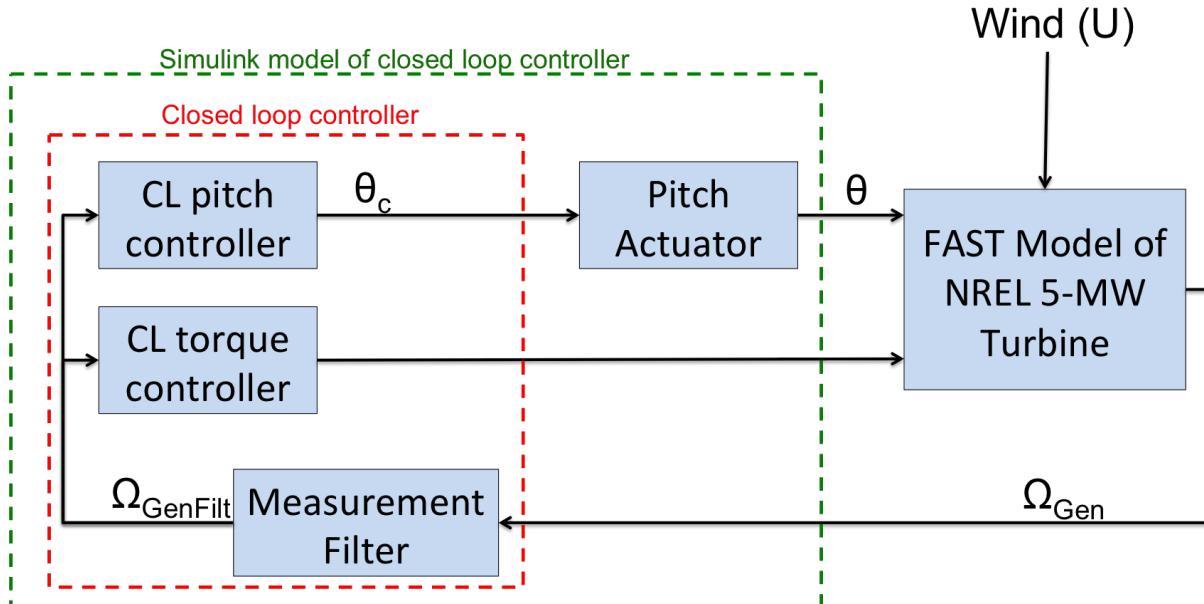


Figure 3.9. Block diagram of the NREL 5-MW closed loop control system, as modeled in this dissertation.

The pitch actuator is modeled as a second order system with dynamics described by equation 3.1, where θ is the actual collective blade pitch, θ_c is the desired blade pitch requested by the controller, ξ is the damping ratio of the pitch actuator, and

ω_o is the undamped natural frequency of the pitch actuator. Jonkman does define a NREL 5-MW blade pitch actuator in [2], but Jonkman has admitted that the pitch actuator defined in [2] is very fast and is unrealistic.[34] Instead we use $\xi = 0.7$ and $\omega_o = 1\text{Hz}$, which have been used to model NREL 5-MW pitch actuators in previous research on feed forward turbine pitch control.[28, 35]

$$\ddot{\theta} + 2\xi\omega_o\dot{\theta} + \omega_o^2\theta = \omega_o^2\theta_c \quad (3.1)$$

3.3.3 Feed Forward Controller

The feed forward controller is based on the dynamic feed forward pitch controller proposed by Shlipf in.[26] Figure 3.10 illustrates how the feed forward controller interfaces with the closed loop controller described in the previous section. As described in Section 3.3.1, the wind is a disturbance, an uncontrolled input that affects system behavior. Typically, changes in the wind cause the turbine to deviate from the desired operating conditions while the closed loop controller acts to bring the turbine back to the desired operating conditions. Ideally we would like the feed forward controller to cancel out the effect of the disturbance so the turbine never deviates from the desired operating conditions to begin with and the closed loop controller would not need to act.

As discussed in the previous section, the goal of region 3 pitch control is to maintain a constant rotor speed of 12.1 RPM. This constant rotor speed is maintained when the aerodynamic torque applied to the rotor by the wind is equal to the torque applied to the rotor by the generator and there is no net torque to accelerate or decelerate the rotor. If the rotor is spinning at 12.1 RPM and the wind speed is anywhere between the rated wind speed (11.4 m/s) and the cutout wind speed (25 m/s) of the turbine there is a pitch angle (θ) that will cause an aerodynamic torque on the rotor that perfectly matches the torque applied to the rotor by the generator. We'll call this pitch angle the steady state pitch (θ_{ss}). Though the steady state pitch can't be calculated it can estimated through simulation. The relationship between steady state pitch and wind speed is shown in Figure 3.11.

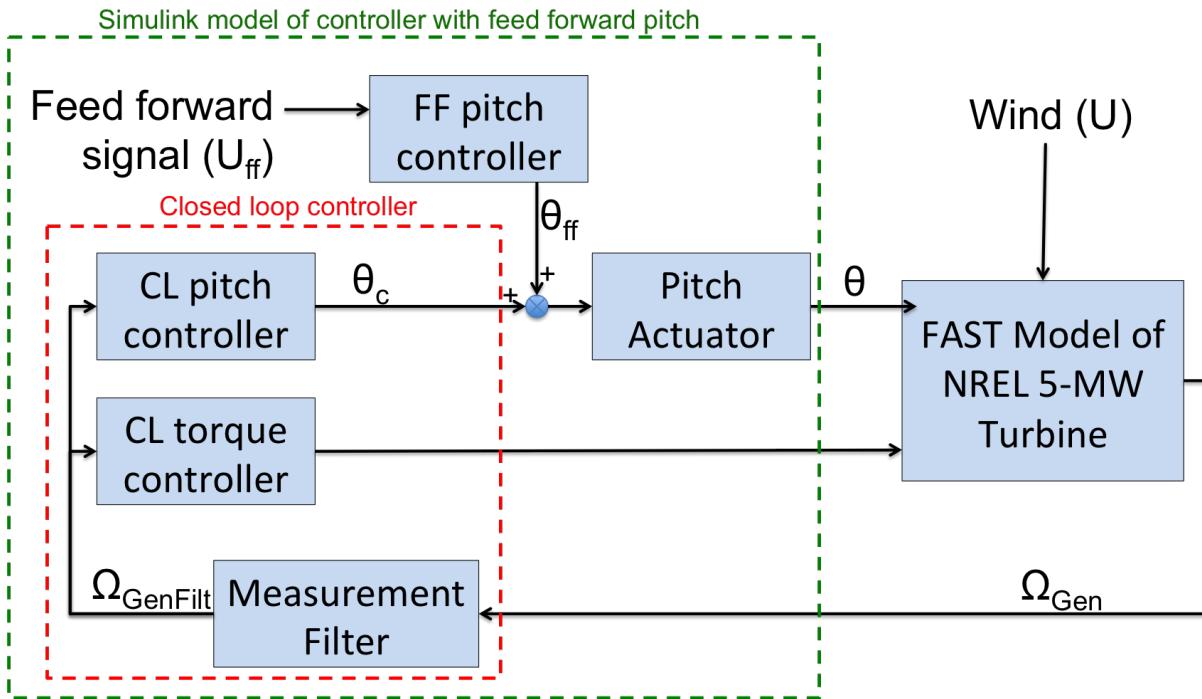


Figure 3.10. Block diagram of the NREL 5-MW controller with feed forward and closed loop control.

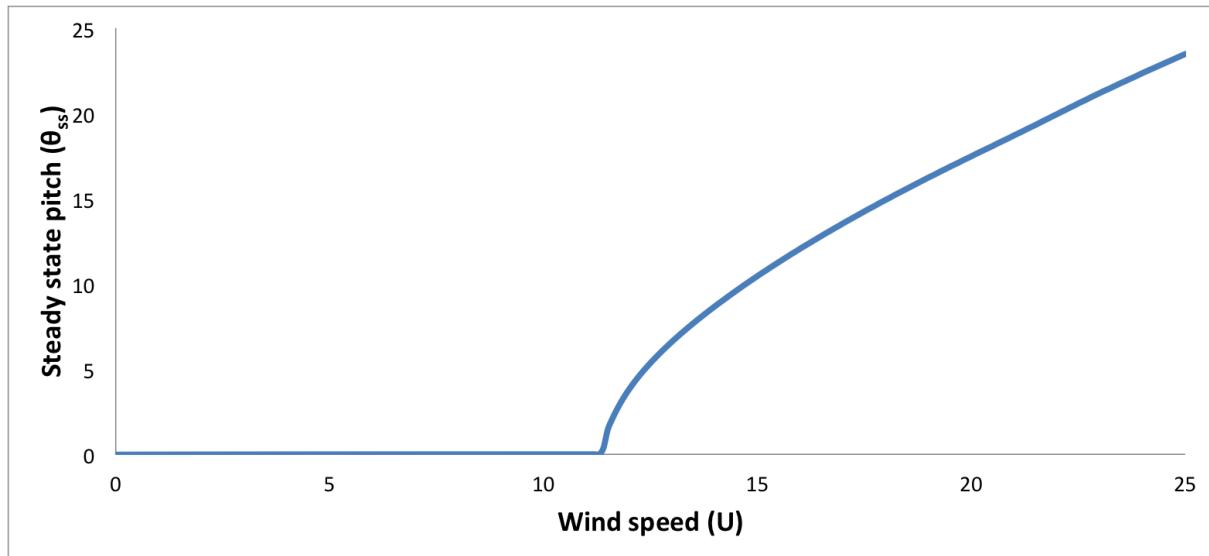


Figure 3.11. Steady state pitch (θ_s), in deg, as a function of incoming wind speed, in m/s.

If the collective blade pitch of the turbine (θ) always matches the steady state pitch (θ_{ss}) for the incoming wind speed (U) then the turbine will never deviate from the rated rotor speed of 12.1 RPM. If the feed forward signal U_{ff} is an accurate

estimate of the incoming wind speed (U) then it is possible to design a feed forward controller that causes the collective blade pitch to always match the corresponding steady state pitch. Since all pitch commands pass through the pitch actuators we can only ensure the correct feed forward pitch signal if we first pass the feed forward signal U_{ff} through a function that inverts the dynamics of the blade pitch actuator. Therefore, the pitch commands supplied by the feed forward pitch controller (θ_{ff}) is given by the equations 3.2 and 3.3. θ_{ss} is the steady state pitch function that was experimentally determined and is illustrated in Figure 3.11. U_{ffd} is the dynamic feed forward velocity, which accounts for and cancels out the dynamics of the pitch actuator.

$$\theta_{ff} = \theta_{ss}(U_{ffd}) \quad (3.2)$$

$$U_{ffd} = (\ddot{U} + 2\xi\omega_o\dot{U} + \omega_o^2 U)/\omega_o^2 \quad (3.3)$$

If the feed forward wind speed estimate U_{ff} is perfectly accurate then the feed forward signal generated using equations 3.2 and 3.3 will completely compensate for fluctuations in wind speed and the closed loop controller will never issue a pitch command. However, we know that in practice U_{ff} will not be a perfect estimate of incoming wind speed and the closed loop controller will act whenever the feed forward pitch controller does not perfectly compensate for fluctuations in wind speed.

3.4 Performance With Perfect Wind Speed Knowledge

Now that the feed forward control system has been defined the performance of this feed forward control system can be examined. In the following sections we examine the performance of the feed forward controller when it has perfect knowledge of the incoming wind speed. Two wind cases are examined. The first case is a large gust. These occur infrequently, but can be very damaging to a turbine. The second case is a turbulent wind inflow, which is typical of what a turbine experiences in

day to day operation. In any real world implementation of this control system the feed forward controller will have imperfect knowledge of the incoming wind. As discussed in Chapter 2, a wind speed estimator provides imperfect estimates of both the wind speed (see Section 2.4.1) and convection velocity (see Section 2.5). In addition, we know that Taylor's frozen turbulence hypothesis is not entirely accurate in practice and the wind speed fluctuations experienced by the downwind turbine will not be identical to the wind speed fluctuations experienced by the upwind turbine. However, simulating the feed forward control system with perfect knowledge of the incoming wind speed can provide valuable insight into the behavior of the feed forward control system, and can demonstrate the performance improvements that might be possible when the feed forward controller described in Section 3.3 is operated in ideal conditions.

3.4.1 Gust Response

Figure 3.12 shows an Extreme Operating Gust (EOG) for an NREL 5-MW turbine operating in 16 m/s wind according to IEC 61400-1.[36] Though extreme operating gusts occur infrequently, they can be very damaging to a wind turbine when they do occur. To examine the system's response to this extreme operating gust, both the upwind turbine and downwind turbine are subjected to a uniform wind inflow (i.e. the wind has no turbulence and all points on the turbine rotor are subjected to the same wind speed) with the magnitude shown in Figure 3.12. Since we are assuming the feed forward controller has perfect knowledge of the incoming wind speed, the wind speed information in Figure 3.12 is also used as the feed forward wind speed estimate.

Figures 3.13 through 3.17 illustrate how the upwind and downwind turbines respond to the 16 m/s extreme operating gust, while Table 3.1 is a summary of some of the performance metrics that will be discussed in the following paragraph. By comparing the response of the upwind turbine, which does not have feed forward pitch control, to the response of the downwind turbine, which does have feed forward pitch control, we can see how feed forward pitch control affects turbine performance.

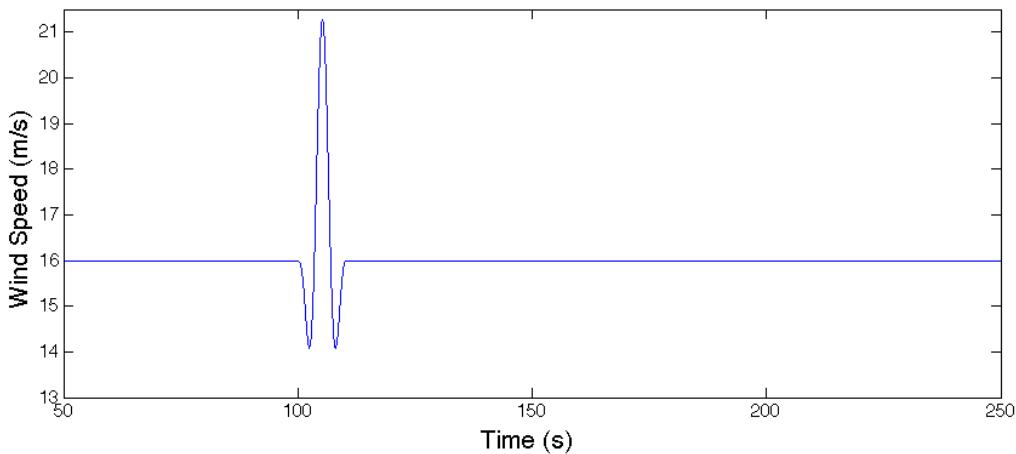


Figure 3.12. Extreme Operating Gust (EOG) in 16 m/s wind.

In Figure 3.13 it can be observed that the turbine with feed forward pitch control (the downwind turbine) responds to the gust sooner than the turbine without feed forward control (the upwind turbine). This is unsurprising. As discussed in Section 3.1, a turbine using only closed loop control will respond reactively to changes in turbine behavior, which are caused by changes in wind speed. On the other hand, a turbine with feed forward control can respond to changes in wind speed before they affect turbine behavior. In fact, a perfect feed forward controller would prevent the wind speed changes from affecting turbine behavior at all. We can also see in Figure 3.13 that the turbine with feed forward control has higher maximum pitch angle when responding to the gust, and that the blade motion has less transient behavior after the gust passes.

Figure 3.14 shows that deviations from the rated rotor speed of 12.1 RPM are greatly reduced with feed forward control. The turbine with feed forward control has a maximum rotor speed of 12.26 RPM, which is only 1.3% over the rated speed. Without feed forward control, the rotor speed reaches a peak value of 14.38 RPM, which is 18.8% over the rated rotor speed. A 18.8% rotor overspeed can be problematic as it may cause an emergency shut down of the turbine. According to Tobias Wehrhan of Oak Engineering LLC the threshold for an overspeed shut down is usually 15-20% over the rated rotor speed, depending on the turbine (personal

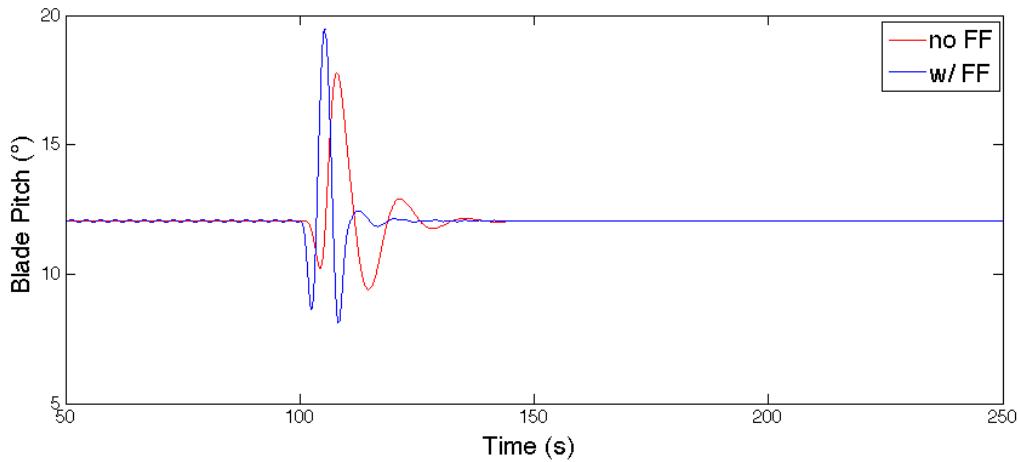


Figure 3.13. Blade pitch caused by 16 m/s EOG when feed forward signal is based on true wind speed.

communication, March 19, 2015). Once an overspeed shut down has been initiated, the turbine stops generating power until it can be safely restarted. This can lead to a significant loss of generated energy and therefore a loss of revenue for the turbine operator.

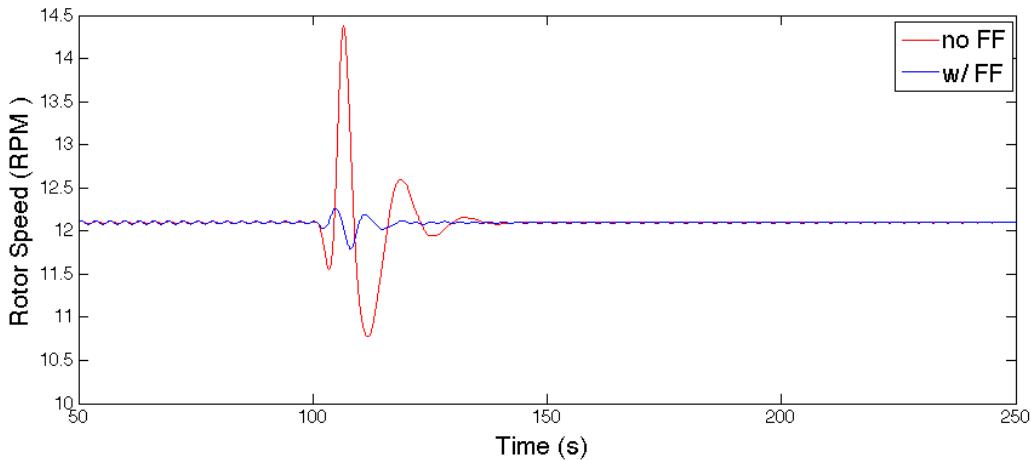


Figure 3.14. Rotor speed caused by 16 m/s EOG when feed forward signal is based on true wind speed.

Figure 3.15 shows that deviations from the rated power are greatly reduced with feed forward control. It is desirable to limit spikes in power. Power spikes can cause an emergency shut down of the turbine if it drives the DC-bus voltage above safe levels, which is typically 15-18% above the nominal DC-bus voltage (Tobias

Wehrhan, personal communication, March 19, 2015). FAST does not model the DC-bus voltage, so these results don't show how close the turbine without feed forward control comes to triggering an over voltage shut-down. Table 3.1 shows the total energy generated in each simulation. We see that the performance improvements from feed forward control are achieved without a cost to energy generation. The turbine with feed forward control actually generates slightly more energy than the turbine without feed forward control.

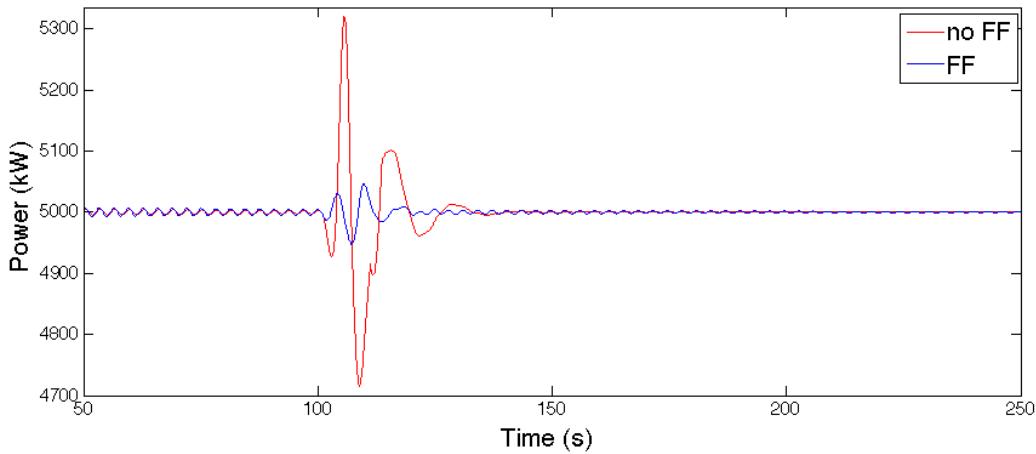


Figure 3.15. Power generation caused by 16 m/s EOG when feed forward signal is based on true wind speed.

Excessive structural loading can cause turbine components to immediately fail, or can cause excessive wear and tear that shortens the life of turbine components. When turbine components fail the turbine must be shut down and repaired. If we can reduce the frequency of shut downs and repairs we can increase the revenue (by increasing energy generation) and decrease the cost to the turbine operator. There are many turbine components that can experience damage from gust loading. In this section we have chosen to examine two loads, the fore-aft bending moment at the base of the tower (Figure 3.16) and Blade root bending moment (BRM) at the root of one of the turbine blades (Figure 3.17). We see similar behavior in both Figures. The turbine with feed forward control experiences lower loads during the extreme operating gust, while experiencing nearly identical loads at all other times. The feed forward controller reduces the maximum tower bending moment from 100,706

kNm to 52,913 kNm, a reduction of 47%, and reduces the maximum BRM from 12,176 kNm to 7,496 kNm, a reduction of 38%.

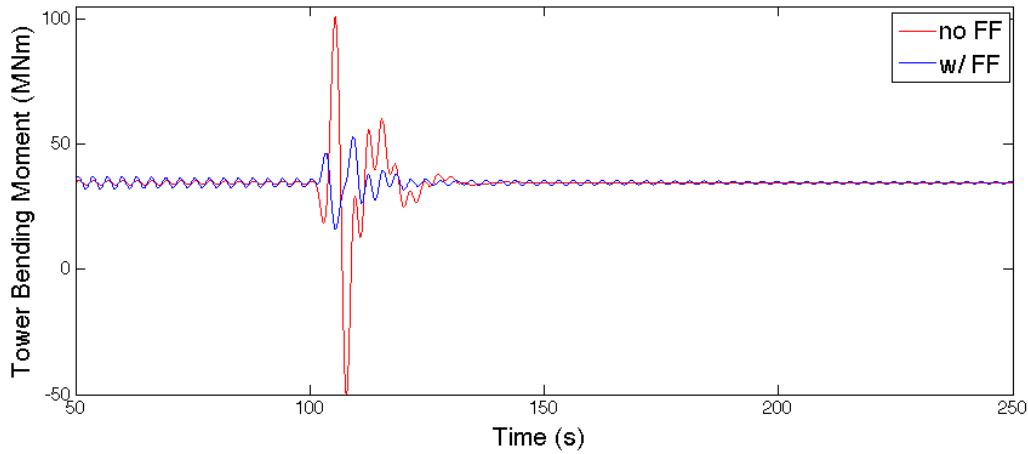


Figure 3.16. Tower base fore-aft moment caused by 16 m/s EOG when feed forward signal is based on true wind speed.

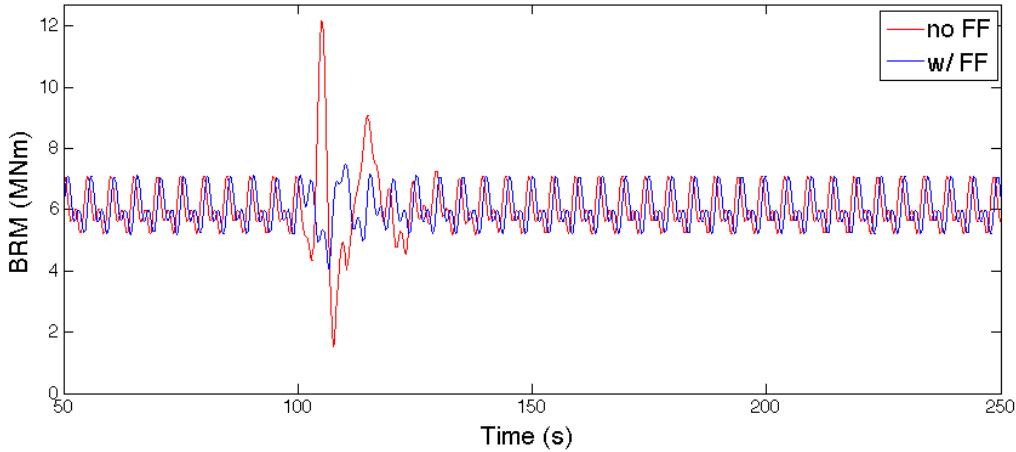


Figure 3.17. Blade root bending moment (BRM) caused by 16 m/s EOG when feed forward signal is based on true wind speed.

Table 3.1 summarizes the performance metrics that have been discussed in this subsection. We see that feed forward control has significantly reduced structural loading of the turbine without decreasing energy generation. This can potentially lead to a reduction in the repair and maintenance costs of the turbine without a reduction in revenue from the turbine. The turbine without feed forward control experienced a 18.8% rotor overspeed, which may be enough to induce an emergency

Turbine Control	Max Tower Base Bending Moment (MNm)	Max Blade Root Bending Moment (MNm)	Max Rotor Overspeed (%)	Energy Generated (kWh)
Closed Loop	100.71	12.18	18.8	833.62
Closed Loop With Feed Forward	52.91	7.50	1.3	834.23

Table 3.1. EOG response with feed forward signal based on true wind speed.

shutdown of the turbine. Overspeed shut down faults are not modeled in these simulations, but if the turbine without feed forward control experienced an overspeed shut down then it would generate significantly less energy than the turbine with feed forward control. In that scenario the turbine with feed forward control would have lower repair and maintenance costs while having higher revenue from energy generation.

3.4.2 Turbulent Wind Response

In this subsection we examine how our two turbine system reacts to turbulent wind. Like the previous subsection, both the upwind turbine and downwind turbine are subjected to the same wind speed fluctuations. The downwind turbine is equipped with the feed forward controller described in Subsection 3.3.3 and is assumed to have perfect knowledge of the incoming wind. The feed forward controller relies on knowledge of the incoming wind speed and the derivatives of the incoming wind speed. However, in a turbulent wind field the wind speed is not constant over the swept area of the rotor so there is no single measurement that can be considered the true incoming wind speed. In this subsection we will treat the average wind speed over the swept area of the rotor as the wind speed. Because this rotor average wind speed has some high frequency fluctuations it is passed through a single pole low-pass filter with a corner frequency of 0.5 Hz to generate the feed forward wind speed.

Figure 3.18 shows the rotor average wind speed and feed forward wind speed for 200 seconds of the turbulent wind field used in this subsection. The complete wind field is 600 seconds long and was generated by TurbSim using a 16 m/s mean wind speed and the GPLLJ turbulence model, which is based on experimental data gathered in Lamar, Colorado. This was one of the 66 test cases used in Chapter 2 and more details about this test case can be found in Section 2.2. Figures 3.19 through 3.23 illustrate how the upwind and downwind turbines respond to the turbulent wind field. All figures in this subsection have the same axes as the corresponding figures in Subsection 3.4.1. This helps illustrate the differences in magnitude between the wind turbine response to an extreme operating gust and the wind turbine response to a turbulent wind field with the same mean wind speed.

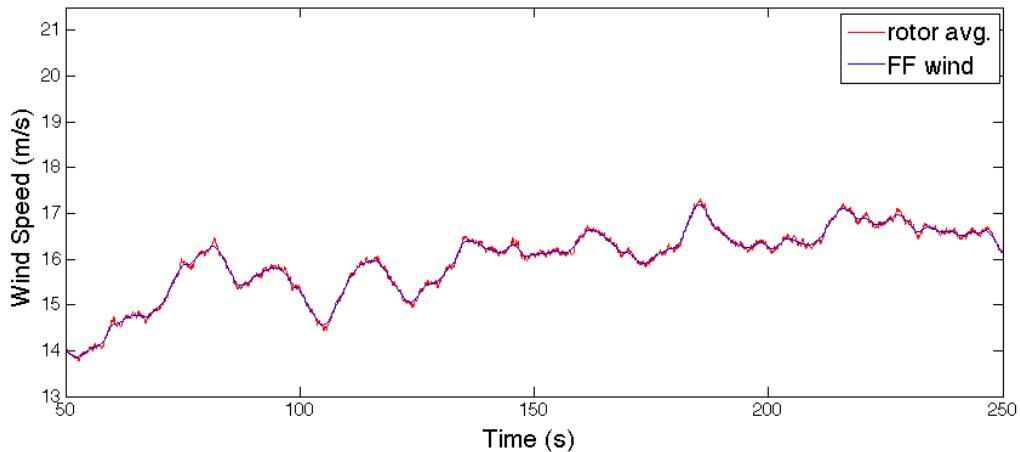


Figure 3.18. Rotor average wind speed and feed forward wind speed for 16 m/s turbulent wind.

In Figure 3.19 we see the blade pitch response of each turbine. We see that the blade pitch response of each turbine follows a similar pattern of peaks and valleys. However, for the turbine with feed forward pitch control the peaks and valleys occurs a few seconds earlier. This is expected because both turbines are reacting to the same fluctuations in wind speed but the turbine with feed forward control can react to those wind speed fluctuations before they affect turbine behavior.

Figures 3.20 and 3.21 show that the feed forward controller reduces deviations in both rotor speed and power. With feed forward control the rotor speed stays within

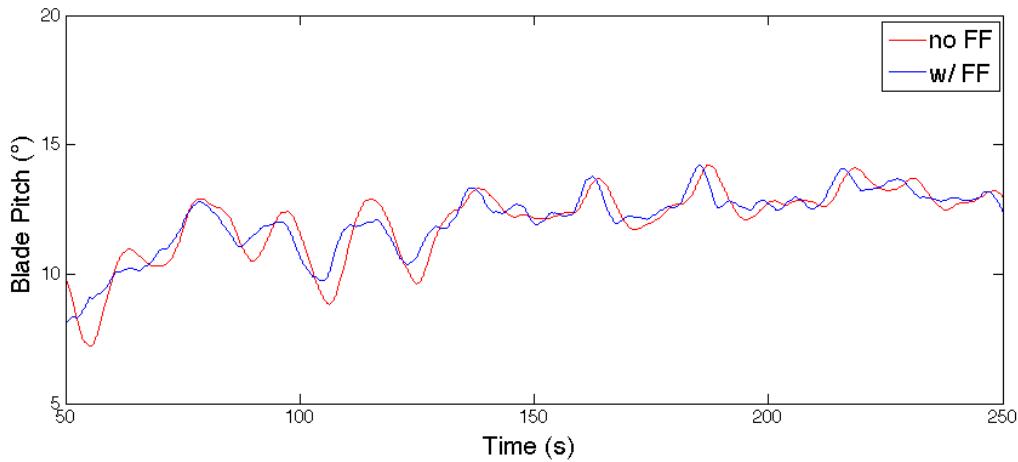


Figure 3.19. Blade pitch caused by 16 m/s turbulent wind when feed forward signal is based on true wind speed.

1.5% of the rated rotor speed and the power stays within 0.76% of the rated power. Without feed forward control the rotor speed deviates as much as 4.7% from the rated speed and the power deviates as much as 1.2% from the rated power. Though the turbine with feed forward control performs better than the turbine without feed forward control it is important to note that neither turbine has a spike in rotor speed or power that could trigger an emergency shut down.

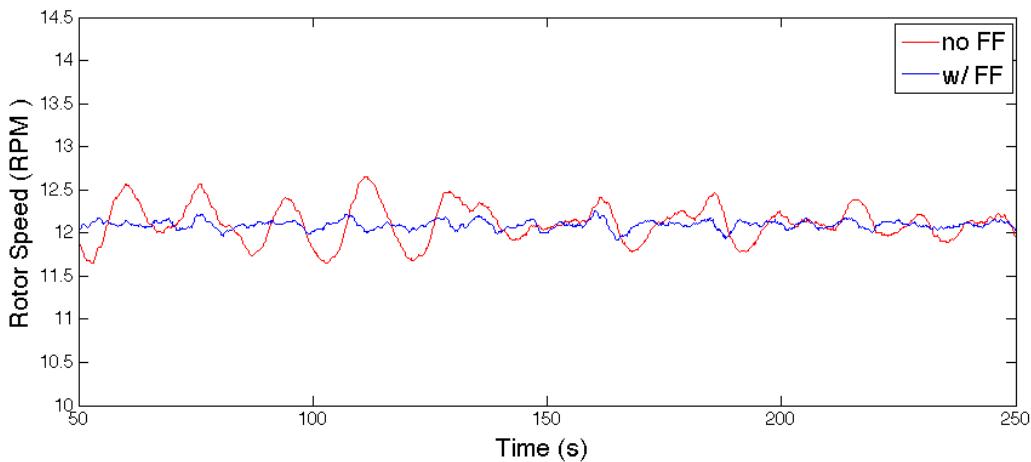


Figure 3.20. Rotor speed caused by 16 m/s turbulent wind when when feed forward signal is based on true wind speed.

Figures 3.22 and 3.23 show the tower fore-aft bending moment and the blade root bending moment. In the previous subsection, when we examined the turbines'

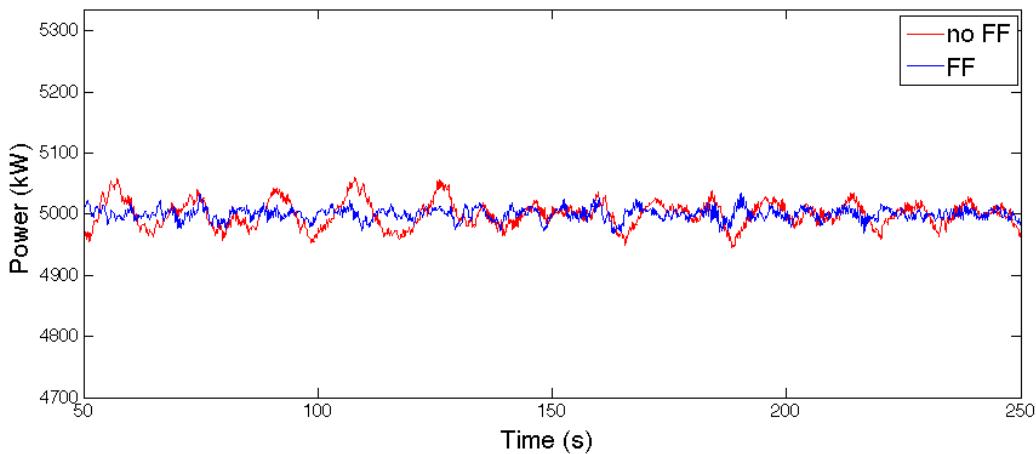


Figure 3.21. Power generation caused by 16 m/s turbulent wind when feed forward signal is based on true wind speed.

responses to an extreme operating gust, it was immediately obvious that feed forward control significantly reduced loads. In Figures 3.22 and 3.23 it is not obvious that feed forward control is reducing loads. Though there are many moments in the simulation when feed forward control reduces loading, there are also many times when feed forward control causes higher loading. To determine which control system has better performance we must statistically analyze the loading.

In the previous subsection we analyzed loading by measuring and comparing the maximum loads experienced by the turbines. For these simulation results comparing the maximum loads doesn't make sense. In the previous subsection the turbine experienced a single large spike in loading when the extreme operating gust hit the turbines. In this subsection the turbine experiences many loading cycles that all have similar amplitudes.

Table 3.2 shows the mean and standard deviation of several turbine performance metrics. We see that feed forward control does not affect the mean values, but does decrease the variability of the performance metrics. Feed forward control reduces the standard deviation of the power by more than 50%.

Reduced variability in power generation can be beneficial for grid integration of wind turbines. The standard deviation of the tower base bending moment and the blade root bending moment are reduced by 29% and 2% respectively.

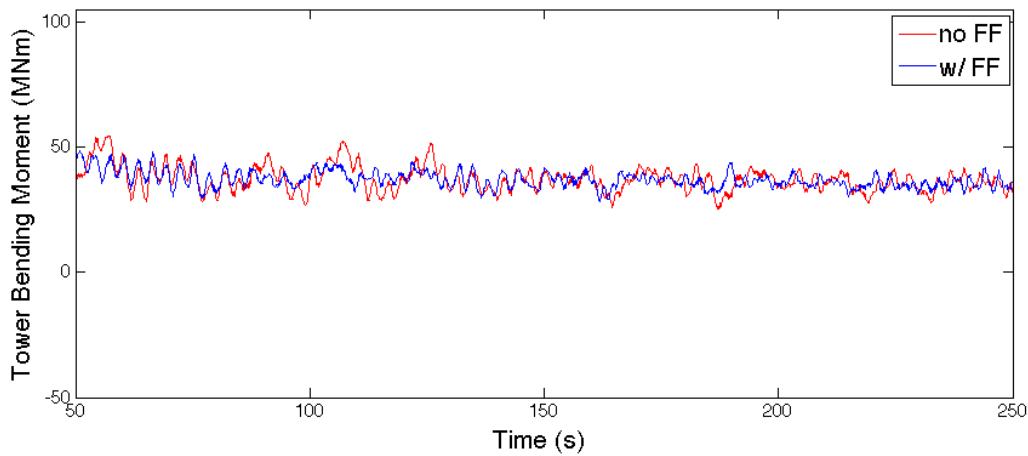


Figure 3.22. Tower base fore-aft moment caused by 16 m/s turbulent wind when feed forward signal is based on true wind speed.

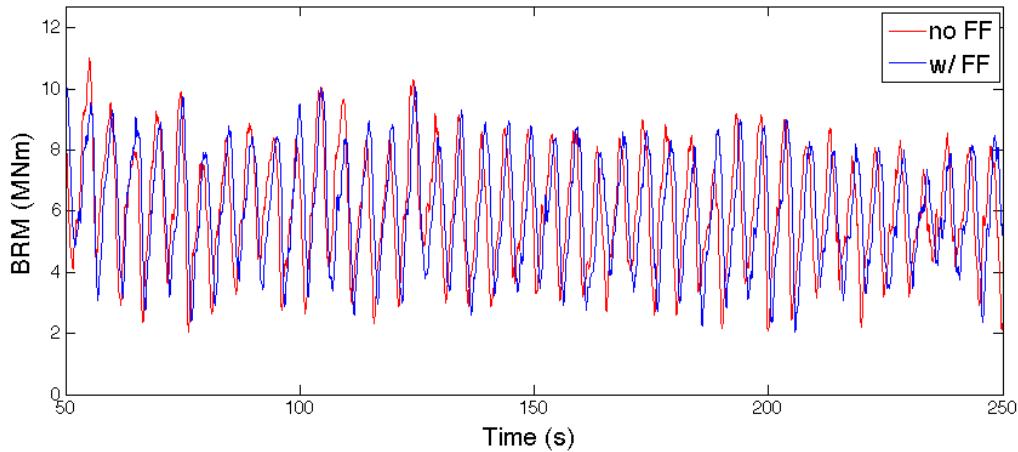


Figure 3.23. Blade root bending moment (BRM) caused by 16 m/s turbulent wind when feed forward signal is based on true wind speed.

Though a reduction in the standard deviation of the structural loads is noteworthy, it is more important to understand how the changes in structural loads will affect the wear and tear on turbine components. To estimate the wear and tear inflicted on the tower base and the blade root we can calculate the short term damage equivalent loads (DEL). An explanation of damage equivalent loads and how they were calculated in this dissertation can be found in Appendix D.

Table 3.3 shows that for this simulation of 16 m/s turbulent wind, feed forward control reduces the damage equivalent loads for both tower base bending moment

Turbine Control	Tower Base Bending Moment (MNm)		Blade Root Bending Moment (MNm)		Rotor Speed (RPM)		Power (kW)	
	Mean	σ	Mean	σ	Mean	σ	Mean	σ
Closed Loop	36.60	4.72	5.98	1.83	12.1	0.208	5,000	21.1
Closed Loop With Feed Forward	36.60	3.34	5.96	1.79	12.1	0.0526	5,000	10.3

Table 3.2. 16 m/s turbulent wind response when feed forward signal is based on true wind speed.

Turbine Control	Tower Base Bending Moment		Blade Root bending Moment	
	DEL (MNm)	Change in DEL	DEL (MNm)	Change in DEL
Closed Loop	45.96	-	10.52	-
Closed Loop With Feed Forward	35.57	-22.6%	9.98	-5.1%

Table 3.3. Short term damage equivalent loads (DEL) for 570 s simulation of turbine in 16 m/s turbulent wind when feed forward signal is based on true wind speed.

and blade root bending moment. This indicates that the turbine with feed forward control experiences less wear and tear on its components, which may lead to less repair and maintenance costs over the life of the turbine.

3.5 Performance With Wind Speed Estimate From Upwind Turbine

The previous sections demonstrated how the feed forward pitch controller affects performance when it has perfect knowledge of the incoming wind. However, in practice the feed forward controller will not have perfect knowledge of the incoming

wind. The feed forward controller will have an estimate of the incoming wind that is generated by using the upwind turbine as a sensor. In the following subsections we examine the performance of the feed forward controller when it has only an estimate of the incoming wind speed. The incoming wind speed estimate is based on the dynamic behavior of the upwind turbine and is generated using the process described in Section 2.4. The simulations described below assume Taylor's frozen turbulence, and assume that we will be able to perfectly determine the convection velocity. We know that neither of those assumptions will be entirely accurate in practice, but the simulations described in this section bring us a step closer to a realistic implementation of feed forward pitch control than the simulations described in the previous section.

3.5.1 Gust Response

Figure 3.24 shows an extreme operating gust for an NREL 5-MW turbine operating in 16 m/s wind according to IEC 61400-1 [36] and the feed forward wind speed estimate generated by the upwind turbine. This is the same extreme operating gust simulated in Section 3.4.1. The feed forward wind speed estimate does not perfectly capture the fluctuations in wind speed. In particular, the feed forward wind speed estimate has smaller peaks and valleys and contains some transient behavior after the extreme operating gust has passed.

Figure 3.25 shows the pitch response of the turbines. As we saw in previous simulations, the turbine with feed forward pitch control reacts to the extreme operating gust sooner than the turbine without feed forward pitch control. One noteworthy feature is the large dip in blade pitch for the turbine with feed forward pitch control. The blade pitch briefly drops to 5.4° at approximately 110 seconds. This is potentially dangerous behavior for the turbine with feed forward pitch control. For a given wind speed, decreasing the blade pitch results in an increase in the wind energy captured by the rotor. This leads to increased structural loads and increased rotor speed. If the blade pitch suddenly drops too low, as may be the case here, it can potentially lead to spikes in structural loads or rotor speed.

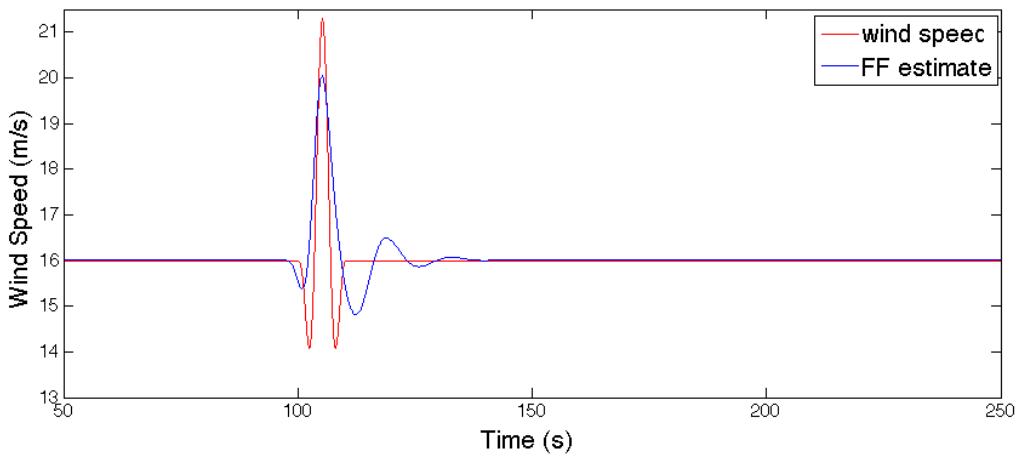


Figure 3.24. Extreme operating gust (EOG) and feed forward wind speed estimate based on gust.

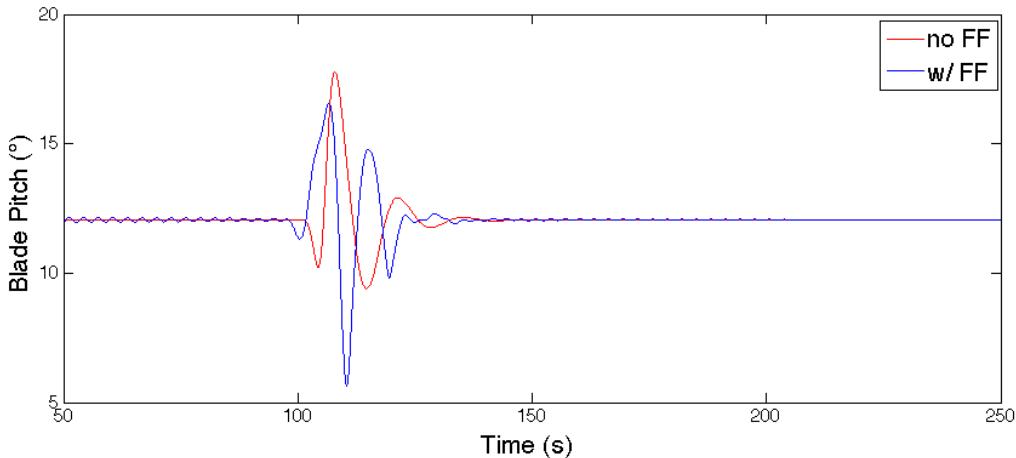


Figure 3.25. Blade pitch caused by 16 m/s EOG when feed forward signal is based on wind speed estimate.

Figure 3.26 shows that feed forward control significantly reduces deviations from the rated rotor speed of 12.1 RPM. The turbine without feed forward pitch control has a maximum rotor speed 18.8% above the rated speed. As discussed before, an overspeed of this magnitude could cause an emergency shutdown of the turbine. The turbine with feed forward pitch control has a maximum rotor speed 6.4% above the rated speed. For the turbine without feed forward control the maximum rotor speed occurs immediately after the peak of the extreme operating gust. The turbine with feed forward pitch control also has a rotor overspeed immediately after

the peak of the extreme operating gust, but the maximum rotor speed occurs approximately 10 seconds later, immediately after the sudden dip in blade pitch that was observed in Figure 3.25. This suggests that the largest rotor overspeed experienced turbine with feed forward pitch control was caused by the control action of the feed forward controller and not by the extreme operating gust.

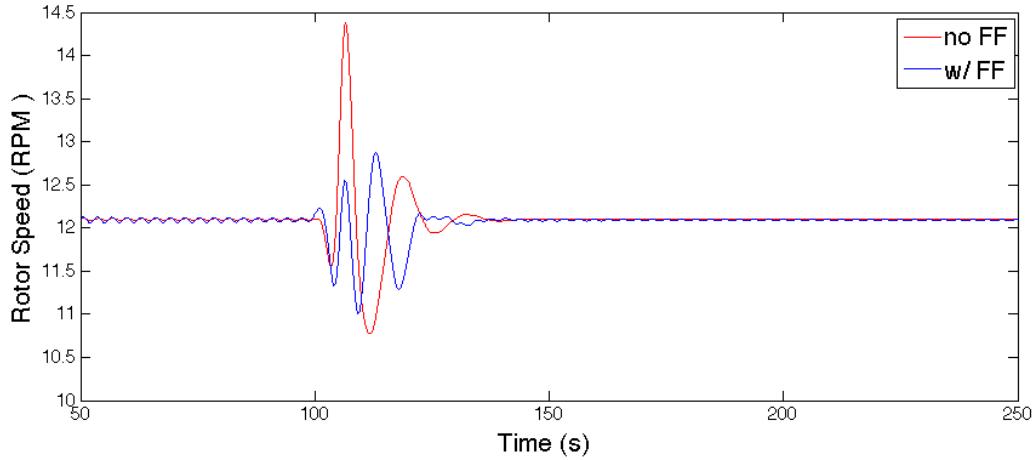


Figure 3.26. Rotor speed caused by 16 m/s EOG when feed forward signal is based on wind speed estimate.

In the remaining figures we see similar behavior to what we see in Figure 3.27. Feed forward pitch control improves performance by reducing deviations from rated power (Figure 3.27), maximum tower fore-aft bending moments (Figure 3.28), and maximum blade root bending moments (Figure 3.29). The turbine with feed forward pitch control experiences a peak in power production and structural loads immediately after the peak of the extreme operating gust. The turbine with feed forward pitch control then experiences an even larger peak in power production and structural loads immediately after the sudden dip in blade pitch that was observed in Figure 3.25.

Table 3.4 summarizes the performance metrics for this simulation. We see that feed forward pitch control reduces structural loading of the turbine and deviations from the rated rotor speed. The use of feed forward pitch control does decrease energy capture by 0.29 kWh, but this is a very small decrease. If the wind plant sells energy for 8 cents per kWh the use of feed forward pitch control would result

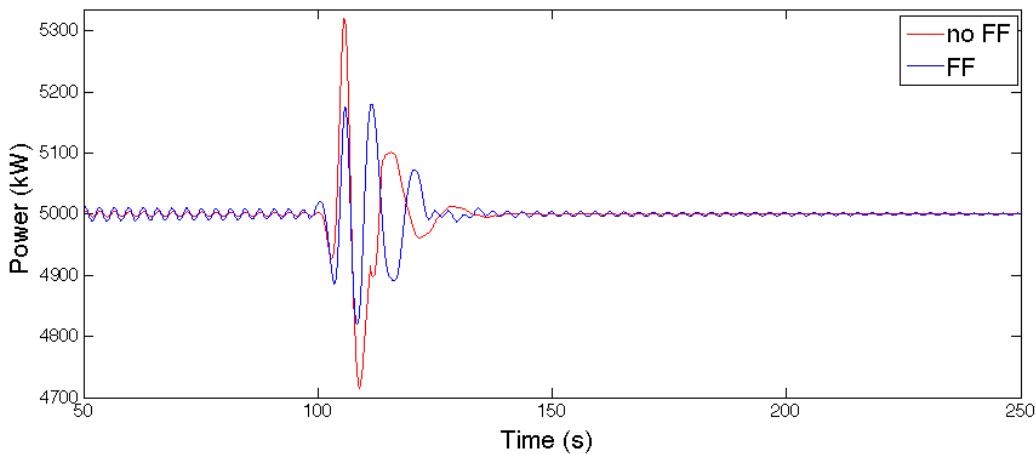


Figure 3.27. Power generation caused by 16 m/s EOG when feed forward signal is based on wind speed estimate.

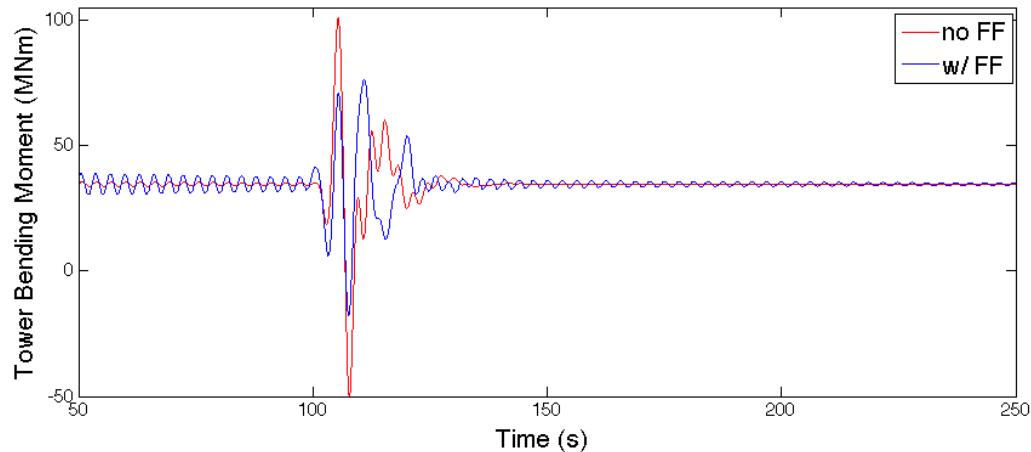


Figure 3.28. Tower base fore-aft moment caused by 16 m/s EOG when feed forward signal is based on wind speed estimate.

in a 2 cent decrease in revenue. Though feed forward pitch control does improve the turbine performance in this simulation, it is also worth noting that the feed forward pitch controller used in this simulation does not perform as well as a feed forward pitch controller with perfect knowledge of incoming wind speed (Table 3.1).

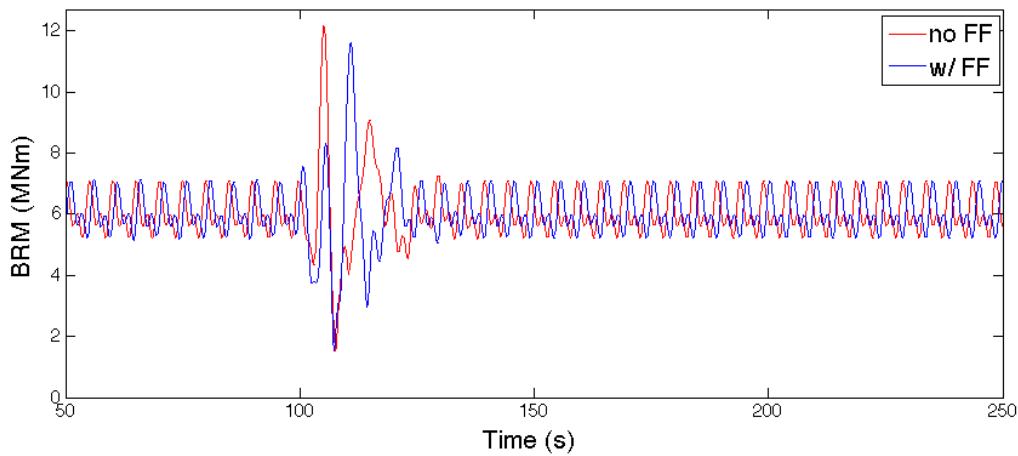


Figure 3.29. Blade root bending moment (BRM) caused by 16 m/s EOG when feed forward signal is based on wind speed estimate.

Turbine Control	Max Tower Base Bending Moment (MNm)	Max Blade Root Bending Moment (MNm)	Max Rotor Overspeed (%)	Energy Generated (kWh)
Closed Loop	100.71	12.18	18.8	833.62
Closed Loop With Feed Forward	76.11	11.60	6.4	833.33

Table 3.4. EOG response with feed forward signal based on wind speed estimate.

3.5.2 Turbulent Wind Response

Figure 3.30 shows the rotor average wind speed and feed forward wind speed for 200 seconds of the turbulent wind field used in this subsection. This subsection uses the same turbulent wind field used in Subsection 3.4.2. Though the feed forward wind speed, estimated through the rotor dynamics of the upwind turbine, does not perfectly match the rotor average wind speed, it does capture most of the peaks and valleys of the wind speed fluctuations.

Figures 3.31 through 3.35 show the dynamic behavior of a turbine with and without feed forward pitch control. We see in these figures that feed forward control reduces deviations in blade pitch (Figure 3.31), rotor speed (Figure 3.32), and power production (Figure 3.33). However, feed forward control does not reduce these

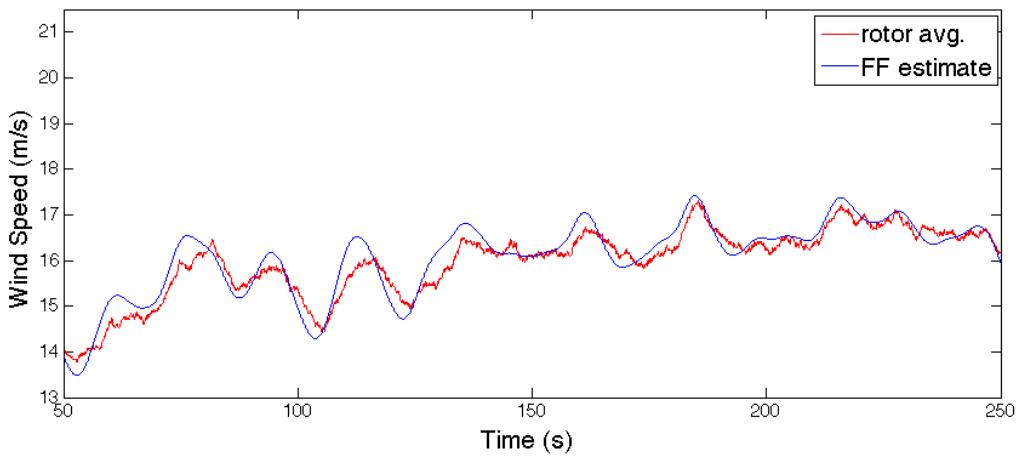


Figure 3.30. Rotor average wind speed and feed forward wind speed estimate of turbulent 16 m/s wind.

deviations as much as we saw in section 3.4.2, where the feed forward signal was based on the true rotor average wind speed.

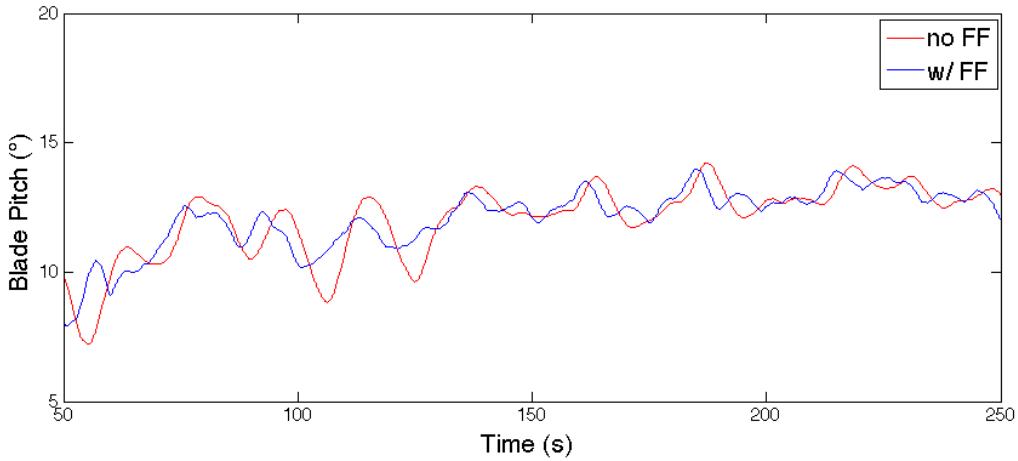


Figure 3.31. Blade pitch caused by 16 m/s turbulent wind when feed forward signal is based on wind speed estimate.

In Figures 3.34 and 3.35 it is difficult to determine how feed forward control affects the structural loading of the turbine components. However, Table 3.6 shows that feed forward control does in fact reduce the damage equivalent loads of the tower base bending moment (17.4% reduction) and the blade root bending moment (5.0% reduction). Note that these reductions in damage equivalent loads are not as large as the reductions observed when the feed forward wind speed was the true

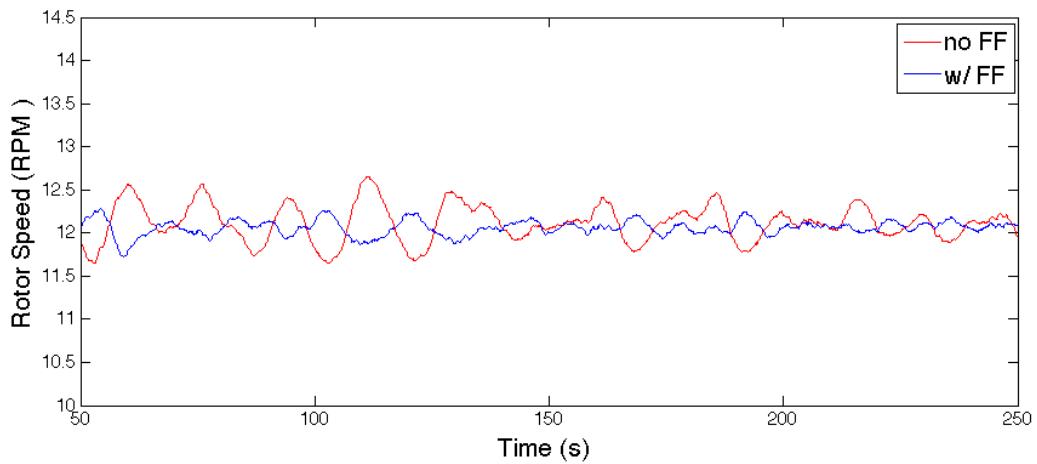


Figure 3.32. Rotor speed caused by 16 m/s turbulent wind when feed forward signal is based on wind speed estimate.

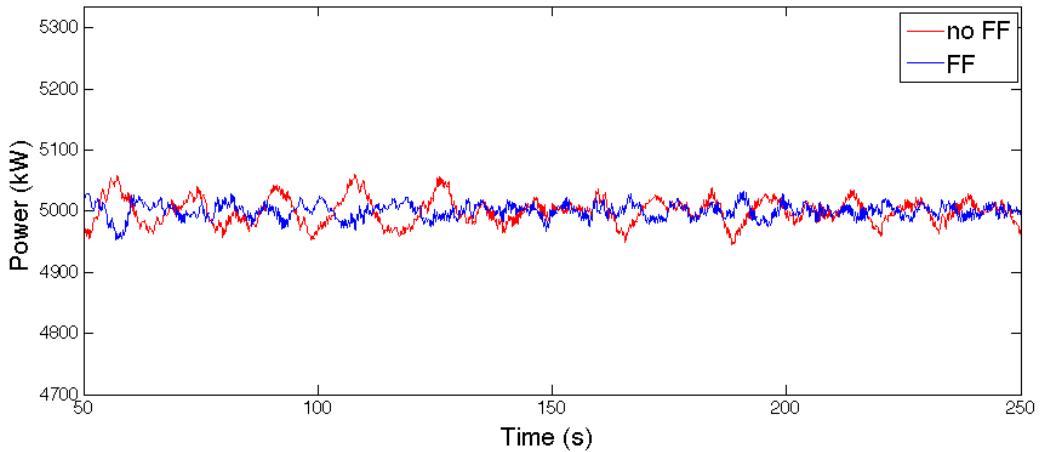


Figure 3.33. Power generation caused by 16 m/s turbulent wind when feed forward signal is based on wind speed estimate.

rotor average wind speed (Subsection 3.4.2).

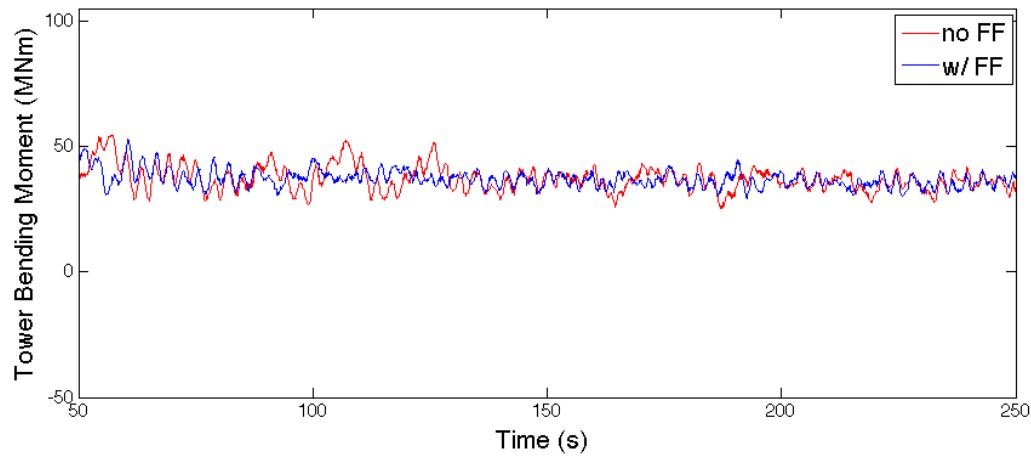


Figure 3.34. Tower base fore-aft moment caused by 16 m/s turbulent wind when feed forward signal is based on wind speed estimate.

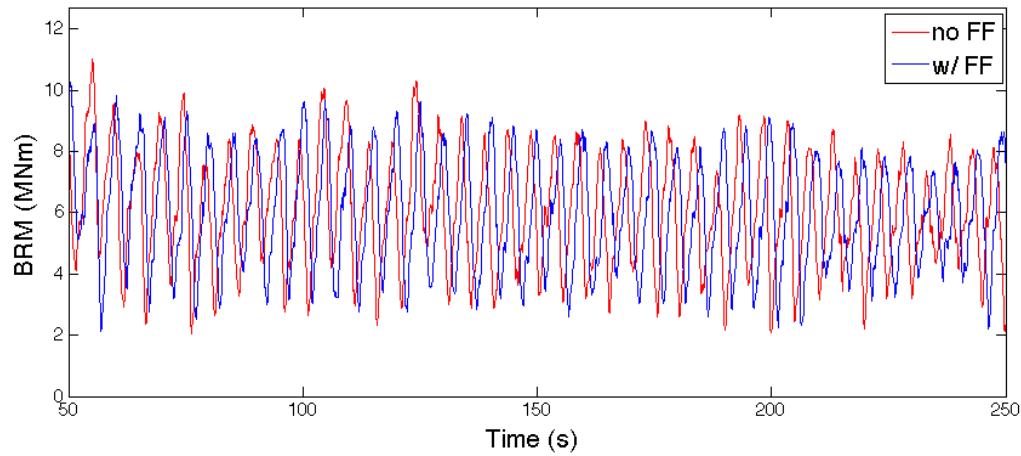


Figure 3.35. Blade root bending moment (BRM) caused by 16 m/s turbulent wind when feed forward signal is based on wind speed estimate.

Turbine Control	Tower Base Bending Moment (MNm)		Blade Root Bending Moment (MNm)		Rotor Speed (RPM)		Power (kW)	
	Mean	σ	Mean	σ	Mean	σ	Mean	σ
Closed Loop	36.60	4.72	5.98	1.83	12.1	0.208	5,000	21.1
Closed Loop With Feed Forward	36.60	3.48	5.97	1.79	12.1	0.081	5,000	11.8

Table 3.5. 16 m/s turbulent wind response when feed forward signal is based on wind speed estimate.

Turbine Control	Tower Base Bending Moment		Blade Root Bending Moment	
	DEL	Change (MNm)	DEL	Change (MNm)
Closed Loop	45.96	-	10.52	-
Closed Loop With Feed Forward	37.94	-17.4%	9.99	-5.0%

Table 3.6. Short term damage equivalent loads for 570 s simulation of turbine in 16 m/s turbulent wind when feed forward signal is based on wind speed estimate.

3.6 Sensitivity to Errors in Feed Forward Data Timing

In the previous section we observed that feed forward pitch control based on wind speed estimates from an upwind turbine can improve performance and reduce structural loading provided Taylor's frozen turbulence holds and we are able to perfectly estimate the convection speed of the wind. However, experiments have shown that Taylor's frozen turbulence hypothesis is never completely accurate and it is impossible to perfectly estimate the convection velocity. In this section we bring our system one step closer to reality by considering the effect of errors in the convection velocity estimate.

In Section 2.5 we discussed a method for estimating the convection velocity by calculating the mean wind speed estimate over some period of time. For the 66 test cases examined in Section 2.5 we found that the accuracy of the convection velocity estimate varied with both the true mean wind speed, and the time period over which the wind speed estimates were averaged. For wind speeds above the rated wind speed (the conditions we are considering for feed forward pitch control) the mean discrepancy between the true and estimated convection velocity ranged from 2% to 4%.

By making a few reasonable assumptions about our system, we can get a rough estimate of the convection velocity error we might expect for the 16 m/s turbulent wind field our system was subjected to in Sections 3.4.2 and 3.5.2. We can also determine how that convection velocity error would affect the timing of the feed forward wind speed estimates received by the downwind turbine. If we assume a 60 second averaging time, we might expect the error in our convection velocity estimate to average approximately 3.2% (Figure 3.26), or 0.5 m/s. If we assume the downwind turbine is 10 rotor diameters, or 1260 meters, behind the upwind turbine a 3.2% error in the convection velocity estimate will cause an error in the timing of the feed forward of approximately 2.5 seconds.

We can simulate the effect of an error in the convection velocity estimate by mod-

ifying the turbulent wind response simulation from Section 3.5.2. Instead of using perfectly timed feed forward wind speed estimates, the feed forward wind speed estimates is shifted in time. To determine how errors in the convection velocity estimate affect system performance we can calculate the resulting damage equivalent loads (DEL) on the tower bending moment and blade root moment.

Figure 3.36 illustrates how timing errors in the feed forward data affect the damage equivalent load of the tower fore-aft bending moment. A negative timing error indicates the feed forward data is ahead of the turbulent wind field. In other words, the system has overestimated the convection velocity and as a result the turbulent wind fluctuations are arriving at the downwind turbine later than the system expects them to.

The red line indicates the tower base DEL for the upwind turbine. Since the upwind turbine does not depend on feed forward data, the tower base DEL for the upwind turbine is constant across all simulations. The blue line indicates the tower base DEL for the downwind turbine. We see from the figure that a small negative error in the convection velocity estimate actually improves performance. The minimum damage equivalent load corresponds to a timing error of approximately -0.5 seconds. We also see that if the timing error is less than -2.8 seconds or greater than 1.2 seconds the turbine with feed forward control performs worse than the turbine without feed forward control. This is troubling because earlier in this section we estimated that our timing error for this wind speed would be in the neighborhood of ± 2.5 seconds.

Figure 3.37 illustrates how timing errors in the feed forward data affect the damage equivalent load of the blade root bending moment. Again, a negative timing error indicates the feed forward data is ahead of the turbulent wind field, the red line indicates the DEL of the upwind turbine, and the blue line indicates the DEL of he downwind turbine. We see that a small negative error improves performance. Like the tower base DEL, the blade root DEL is smallest when the timing error is -0.5 seconds. We also see that if the timing error is less than -4.1 seconds or

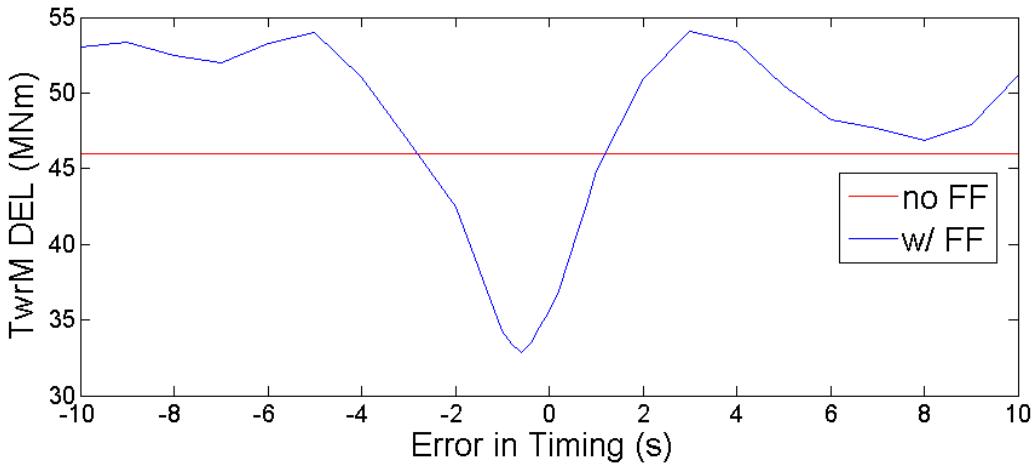


Figure 3.36. Tower base DEL for various timing errors in feed forward wind speed estimates.

greater than 2.3 seconds the turbine with feed forward control performs worse than the turbine without feed forward control. Blade root DEL is improved over a wider range of timing errors than we saw for tower base DEL, however our estimated timing error of ± 2.5 seconds would not guarantee that the downwind turbine with feed forward pitch control would perform better than a turbine with conventional closed loop control. Figure 3.37 also shows that BRM DEL is reduced for timing errors between 5.5 s and 10 s. This is likely due to the feed forward pitch action coincidentally lining up with some of the fluctuations in wind speed. In other words, there are instances where the downwind turbine is reacting to a gust so late that it accidentally mitigates the effect of a later gust. A large error in convection speed would not reliably and consistently reduce BRM DEL.

Timing errors do not affect the average power production. Each simulation had an average power production of 5,000 kW. However, timing errors do affect the standard deviation of the power production. As shown in Figure 3.38, feed forward pitch control reduces the standard deviation of the power production when the timing error is small, but feed forward control increases the standard deviation of the power production for timing errors greater than +2.4 seconds or less than -3.6 seconds. A small standard deviation is desirable as it indicates the power output of the turbine remains closer to the desired power output. Again we see a region of improved

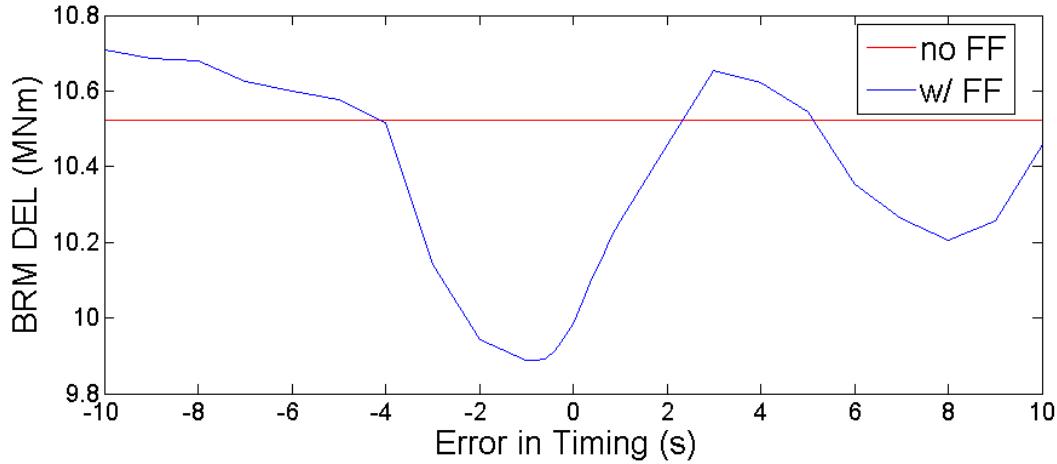


Figure 3.37. Blade root DEL for various timing errors in feed forward wind speed estimates.

performance for large timing errors (+6.4 to +9.6 s), but this improvement is coincidental.

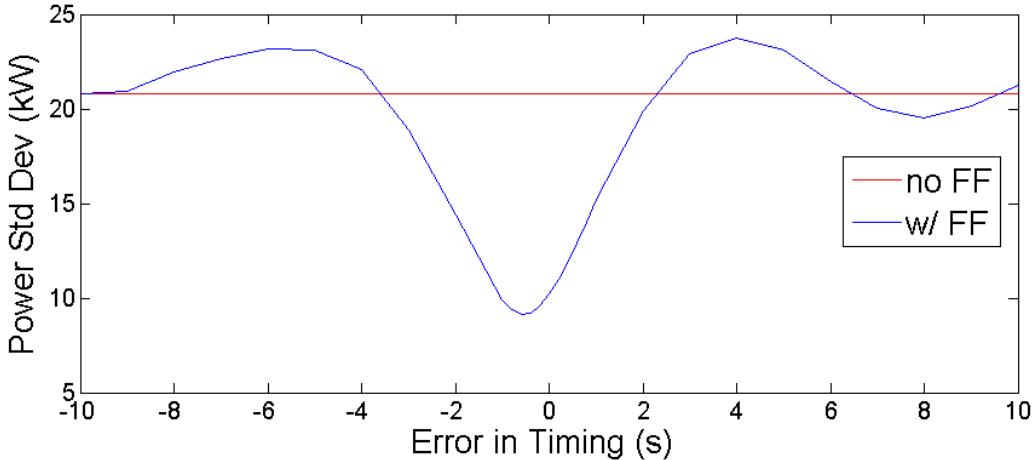


Figure 3.38. Standard deviation of power production for various timing errors in feed forward wind speed estimates.

All of the data presented in this section tell the same story. The feed forward pitch control system is sensitive to timing errors. Though the feed forward control system improves performance when timing errors are small it can degrade performance when significantly large timing errors are present. Based on the convection velocity estimate study carried out in Section 2.5, we can not guarantee timing errors will be small enough to ensure feed forward control will improve turbine per-

formance. A control system that is just as likely to degrade performance as improve it is not particularly useful. As such, the results in this section show that in practice the feed forward pitch control scheme proposed in this chapter is not practical. At this point it is not worthwhile to carry out additional investigations into other real world complications the system would face, such as inaccuracies of the Taylor's frozen turbulence hypothesis.

3.7 Summary and Conclusions

In this chapter a feed forward optimal pitch control scheme is introduced and investigated. In this control scheme an upwind is used as a wind speed sensor. Wind speed measurements are passed from the upwind turbine to a feed forward controller on a downwind turbine. These wind speed measurements give the feed forward controller information about the wind speed fluctuations the downwind turbine will experience in the future. The feed forward controller determines the optimal pitch corresponding to the incoming wind speed fluctuations and issues supplementary pitch commands to the existing feedback control loop of the downwind turbine. These supplementary pitch commands enable the downwind turbine to preemptively respond to incoming wind speed fluctuations potentially improving energy capture and reducing damage to the turbine.

The control scheme is investigated through three rounds of simulation using the NREL FAST turbine simulation tool. The initial round of simulations uses a simplified idealistic model of the system, while the subsequent rounds systematically incorporate complicating factors that a real world system would encounter.

The first round of simulations (Section 3.4) assumes that the feed forward controller has perfect measurements of the incoming wind speed fluctuations, that the convection velocity has been estimated perfectly, and that the wind speed fluctuations experienced by the downwind turbine will be identical to the wind speed fluctuations experienced by the upwind turbine. Results show that in these conditions feed forward control significantly improves turbine performance with respect

to both extreme operating gusts and turbulent wind.

The second round of simulations assumes that the convection velocity has been estimated perfectly, and that the wind speed fluctuations experienced by the downwind turbine will be identical to the wind speed fluctuations experienced by the upwind turbine. It does not assume that the feed forward controller has perfect measurements of the incoming wind speed fluctuations. Instead, the wind speed fluctuations are estimated using the dynamics of the upwind turbine. The results from the second round of simulations show smaller performance improvements from the use of feed forward optimal pitch control (compared to the first round of simulations). However, in these conditions feed forward control still significantly improves turbine performance.

The third round of simulations assume that the wind speed fluctuations experienced by the downwind turbine will be identical to the wind speed fluctuations experienced by the upwind turbine. Wind speed fluctuations are estimated using the dynamics of the upwind turbine and a variety of timing errors are introduced to simulate the effect of errors in estimating the convection velocity. Results from this round of simulations show that the feed forward optimal pitch control system is very sensitive to timing errors. If the timing of the data is off by more than a few seconds the feed forward control system can cause a turbine to perform more poorly than a turbine without feed forward control.

The feed forward optimal pitch control scheme presented in this chapter seemed promising through the first two rounds of simulations. However, the system's high sensitivity to timing errors, and therefore the systems high sensitivity to errors in the estimation of the convection velocity, ultimately make this control scheme impractical in real world applications. In order to effectively use an upwind turbine as a sensor to improve the performance of a downwind turbine we must investigate alternate control schemes that are less sensitive to timing errors.

Chapter 4

Turbine Derating Based on Feed Forward Signal From Upwind Turbine

4.1 Introduction

The overarching objective of this dissertation is to explore techniques for using an upwind turbine as a feed forward sensor to improve the control of downwind turbines. The previous chapter investigated a control scheme in which wind speed measurements from the upwind turbine were used to influence the blade pitch control of the downwind turbine. Initial simulations seemed promising, with the downwind turbine tracking the optimal pitch angle more closely, experiencing lower structural loading, and experiencing smaller rotor overspeeds. However, further simulations showed the control scheme was sensitive to timing errors in the feed forward data, and was likely impractical as a result.

This chapter investigates another method for improving the control of a downwind turbine using an upwind turbine as a sensor. The control scheme investigated in this chapter has been designed to overcome the shortcomings of the control system investigated in the previous chapter while maintaining many of the same performance goals. The following sections will show that this feed forward control scheme can be very insensitive to timing errors in the feed forward data, that

the control system can potentially reduce wear and tear on turbine components by reducing loads caused by large gusts of wind, and that the control system can potentially increase energy capture by reducing overspeed shutdowns in response to large gusts of wind.

The feed forward control scheme investigated in this chapter relies on selective turbine derating to reduce structural loads and rotor overspeeds. Turbine derating, sometimes referred to as turbine curtailment, is when the power produced by a turbine is intentionally reduced below the nominal operating conditions of the turbine. When a turbine is derated, the reduction in power production is generally accompanied by a reduction in structural loads as the turbine extracts less energy from the wind. There are a variety of turbine derating strategies, several of which are investigated by Deshpande and Peters [37], but they all involve reducing either the generator speed and/or the generator torque. This is not surprising since the power generated by the turbine is approximately equal to the generator speed times the generator torque.

Turbine derating can be done for a variety of reasons, and can be done at the request of an electrical utility, or can be done by the wind plant operator in an attempt to improve plant performance. The derating requirements imposed by utilities vary, but to date, most utility requested turbine derating occurs due to lack of sufficient transmission capacity on the utility grid, or in response to high wind generation at times of low electrical load.[38] A utility may also ask a wind plant to derate turbines to help utility grid frequency regulation.[39, 40] A wind plant operator may derate a turbine to reduce structural loads and increase the longevity of turbine components.[41] In applications where wind turbine maintenance is difficult to schedule, such as offshore wind farms, a turbine that experiences damage could be derated and operated at a diminished capacity until a convenient time can be found for inspection and repairs.[42, 43] Derating can be used to implement a "soft cut out" scheme in which the turbine will be derated in very high wind speeds instead of being completely shut down once the cut out wind speed is reached. [44]

If detailed knowledge of the incoming wind speed is known, then a turbine can be dynamically derated to optimize power capture without exceeding structural loading limits.[44–46] In some cases, derating some of the turbines in a wind plant can increase the total power production of the plant.[47]

The remainder of this chapter describes and investigates a control scheme that uses turbine derating in an attempt to mitigate the detrimental effects of a large wind gust propagating through a wind farm. The following section describes the derating scheme used by the feed forward controller. Section 4.3 investigates the relationship between derating, power generation, structural loads, and rotor overspeed when a turbine is subjected to a large wind gust. Section 4.4 investigates the transition between a rated and derated state. Section 4.5 describes the feed forward control system design. Section 4.6 examines the performance of the feed forward control system when subjected to a large gust of wind.

4.2 Turbine Derating

The power generated by a turbine is given by $P_{gen} = T_{gen}\Omega_{gen}\eta_{gen}$ where T_{gen} is the generator torque, Ω_{gen} is the generator speed, and η_{gen} is the efficiency of the generator. For a given wind speed, a turbine can be derated by reducing Ω_{gen} and/or η_{gen} below the normal operating values. Derating a turbine typically causes a reduction in both the power generation and structural loads of the turbine. However, reducing structural loads is not always a goal of turbine derating. Therefore, the load reduction capabilities of derating are not always thoroughly examined in literature. Several derating methods have been proposed in literature. In this section three of those derating methods are investigated to determine which method is best suited for our feed forward controller.

Method 1 is based on “derating strategy C” proposed by Deshpande and Peters in “Wind turbine controller design considerations for improved wind farm level curtailment tracking.”[37] In this method the rated generator speed remains unchanged and the turbine is derated by reducing the rated generator torque. The

torque-speed curve of the generator torque controller is scaled to accomodate this change. Method 2 is based on the derating strategy proposed by Frost, Goebel, and Obrecht in “Integrating Structural Health Management with Contingency Control for Wind Turbines.”[48] In this method the torque-speed curve of the torque controller remains unchanged. The turbine is derated by changing the point on the torque speed curve at which the turbine switches to a region 3 control strategy. Essentially this method derates the turbine by reducing both the rated generator torque and the rated generator speed. Method 3 is based on the derating method proposed by Petrovic and Bottasso in “Wind Turbine Envelope Riding.”[46] In this method the rated generator torque remains unchanged while the turbine is derated by reducing the rated generator speed. To accommodate the reduction in the rated generator speed, the torque-speed curve of the generator torque controller is scaled and a minimum pitch angle is imposed on the blade pitch controller.

To evaluate the three derating methods, all three derating methods are simulated for a pair of test cases. In test case one the turbine has been derated by 30% and is operating in steady 16 m/s when it experiences an extreme operating gust. In test case two the turbine has been derated by 30% and is operating in steady 12 m/s wind when it experiences an extreme operating gust. In both test cases the extreme operating gust is defined according to IEC 61400-1 [36] for a class 1 turbine in category A turbulence. Figures 4.1 through 4.4 illustrate how the derating strategies affect the power generation, tower base fore-aft bending moment, blade root bending moment, and rotor speed. As expected, derating the turbine by 30% results in a 30% reduction in power production for all three derating strategies. The deviations in power production caused by the extreme operating gust are slightly larger for derating method 3, but otherwise all three derating strategies have a very similar effect on power production. In Figure 4.2 we see that all three derating strategies have a similar effect on tower base fore-aft bending moment before and after the extreme operating gust, with a 30% derating corresponding to an average reduction of 31%. However, the derating methods have differing effects on the peak

tower base fore-aft bending moments induced by the extreme operating gust. Derating method 1 has a peak moment of 88.4 MNm, a 13.5% reduction compared to a non derated turbine, while derating method 2 has a peak moment of 85.2 MNm, a 16.6% reduction. Derating method 3 performs significantly better with a peak moment of 65.8 MNm, a 35.6% reduction. Figure 4.3 tells a similar story. All three derating methods have a similar effect on blade root moments when the turbine is in constant wind, but they have differing effects when the turbine is experiencing an extreme operating gust. Once again, derating method 3 out performs the other two methods with respect to reducing peak structural loads. Derating method 1 reduces the peak blade root moment by 17.5%, method 2 reduces the peak blade root moment by 17.0%, and method 3 reduces the peak blade root moment by 29.5%. Figure 4.4 shows that the three derating strategies have very different effects on rotor speed. It is not surprising that the derating method based on reducing the rated torque (method 1) has the least effect on rotor speed, the derating method based on reducing the rated generator speed (method 3) has the greatest effect on rotor speed, and the derating method based on reducing both the rated speed and rated torque (method 2) falls somewhere in between. With derating method 1 the turbine sees a peak rotor speed of 14.18 RPM, only a 1.1% reduction from the non-derated turbine. That peak rotor speed is 17.2% over the turbine's rated rotor speed of 21.1 RPM. With derating method 2 the turbine sees a peak rotor speed of 13.57 RPM, a 5.4% reduction, which corresponds to a peak overspeed of 12.1%. With derating method 3 the turbine sees a peak rotor speed of 10.40 RPM, a 27.5% reduction.

Test case 2, where a turbine is derated by 30% and is operating in steady 12 m/s wind when it experiences an extreme operating gust, shows similar trends to test case 1. Plots of the power generation, tower base fore-aft bending moment, blade root bending moment, and rotor speed for test case two are not shown here. However, the performance observed in both test cases has been quantified and summarized in Table 4.2 and Table 4.2. The tables show that derating method 3 is significantly better than methods 1 and 2 at reducing both the structural loads and

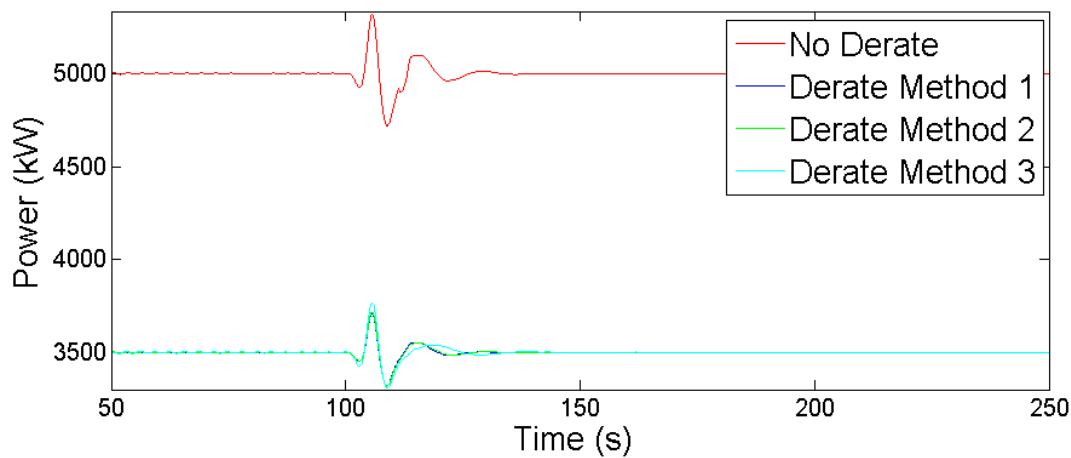


Figure 4.1. Power generated during 16 m/s EOG for 30% derated turbine.

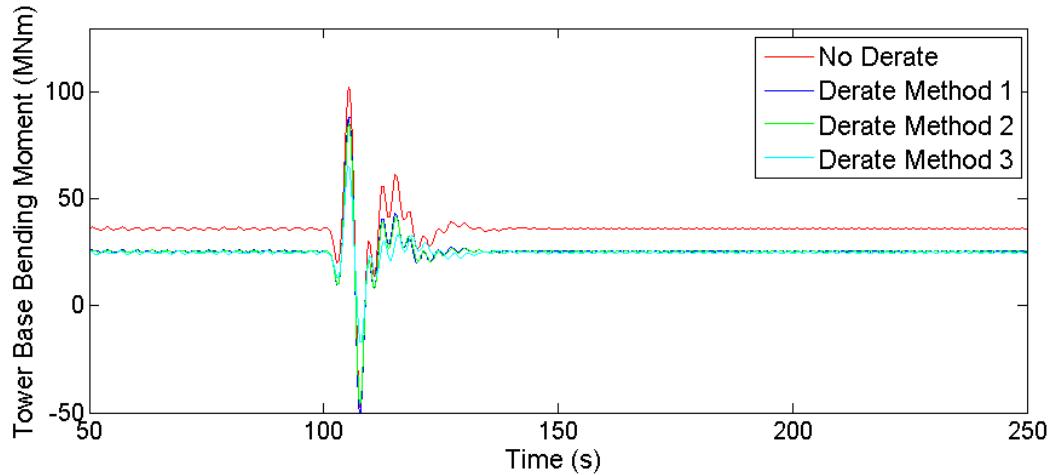


Figure 4.2. Tower fore-aft bending moment for 30% derated turbine during 16 m/s EOG.

the rotor overspeeds caused by an extreme operating gust. Since derating method 3 does the best job of mitigating the negative effects of a large gust of wind, it is best suited for our feed forward controller.

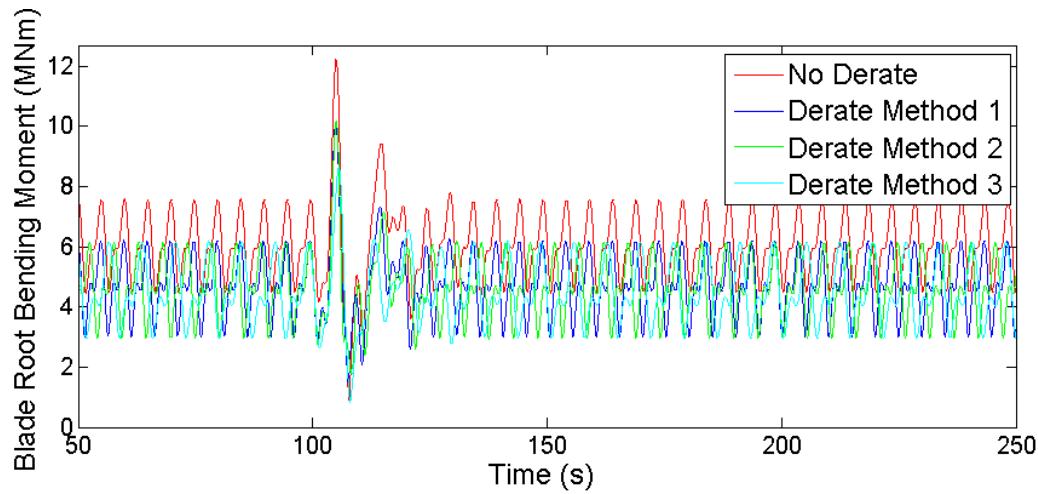


Figure 4.3. Blade root moment for 30% derated turbine during 16 m/s EOG.

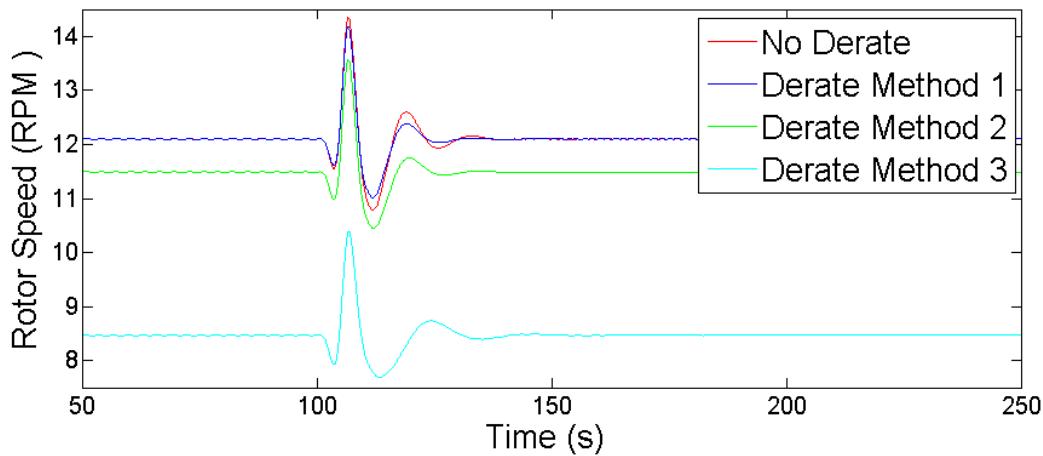


Figure 4.4. Rotor speed for 30% derated turbine during 16 m/s EOG.

	Max Tower Base Bending Moment		Max Blade Root Bending Moment		Max Rotor Speed	
	(MNm)	Decrease	(MNm)	Decrease	(RPM)	Decrease
No Derating	102.2	-	12.25	-	14.35	-
Derating Method 1	88.40	13.5%	10.11	17.5%	14.18	1.1%
Derating Method 2	85.19	16.6%	10.17	17.0%	13.57	5.4%
Derating Method 3	65.81	35.6%	8.64	29.5%	10.40	27.5%

Table 4.1. Effect of derating methods for test case 1: 16 m/s EOG with 30% derating.

	Max Tower Base Bending Moment		Max Blade Root Bending Moment		Max Rotor Speed	
	(MNm)	Decrease	(MNm)	Decrease	(RPM)	Decrease
No Derating	124.7	-	11.84	-	14.07	-
Derating Method 1	91.36	26.7%	11.84	26.1%	13.77	2.1%
Derating Method 2	89.71	28.1%	11.78	26.5%	13.17	6.4%
Derating Method 3	70.20	43.7%	8.30	48.2%	10.38	26.2%

Table 4.2. Effect of derating methods for test case 2: 12 m/s EOG with 30% derating.

All further simulations described in this chapter use derating method 3. When using method 3, a request to derate the turbine results in three changes to the turbine's closed loop controller:

- The rated generator speed (Ω_{GenR}) is reduced.
- The generator torque (T_{Gen}) vs. speed (Ω_{Gen}) curve is scaled such that the rated torque occurs at the new Ω_{GenR} .
- A minimum blade pitch constraint is imposed.

The first two items are sufficient to derate a turbine in region 3 control. The rated generator speed is scaled according to equation 4.1 where Ω_{GenR} is the rated generator speed under normal operation, and η_{DR} is the derate factor. The derate factor indicates the amount of derating as a fraction of 1. For example, $\eta_{DR} = 0.25$ would derate the turbine by 25%. The rated generator torque is not affected by derating. However, the curve must be compressed along the Ω_{Gen} axis such that the rated torque occurs at the new rated speed ($\Omega_{GenR_Derated}$). This requires a change in rated speed (as shown in Equation 4.1) as well as the change in generator torque constant (k_{Gen}) shown in equation 4.2. Appendix C, Figure C.15 contains the Matlab code used for torque control calculations.

$$\Omega_{GenR_Derated} = (1 - \eta_{DR})\Omega_{GenR} \quad (4.1)$$

$$k_{Gen_Derated} = \frac{k_{Gen}}{(1 - \eta_{DR})^2} \quad (4.2)$$

The adjustments in equations 4.1 and 4.2 do not provide sufficient derating when the turbine is operating in region 2 control. In other words, if the turbine is operating in region 2 control and a 20% derate command ($\eta_{DR} = 0.2$) is issued, adjusting Ω_{GenR} and k_{Gen} does not reduce power generation by 20%. This problem can be fixed by imposing a minimum pitch constraint. Normally in region 2 control the turbine blade pitch would be fixed at 0°. Increasing blade pitch reduces energy capture,

which further derates the turbine. Figure 4.5 shows the relationship between derating factor and the blade pitch required for a turbine in region 2 control to achieve that derating. This relationship was based on the WTPerf data discussed in Section 2.4. Imposing a minimum blade pitch based on the relationship shown in Figure 4.5 ensures that a turbine in region 2 control will properly derate. The minimum pitch constraint also enables the turbine to smoothly transition between region 2 and region 3 control while derated.

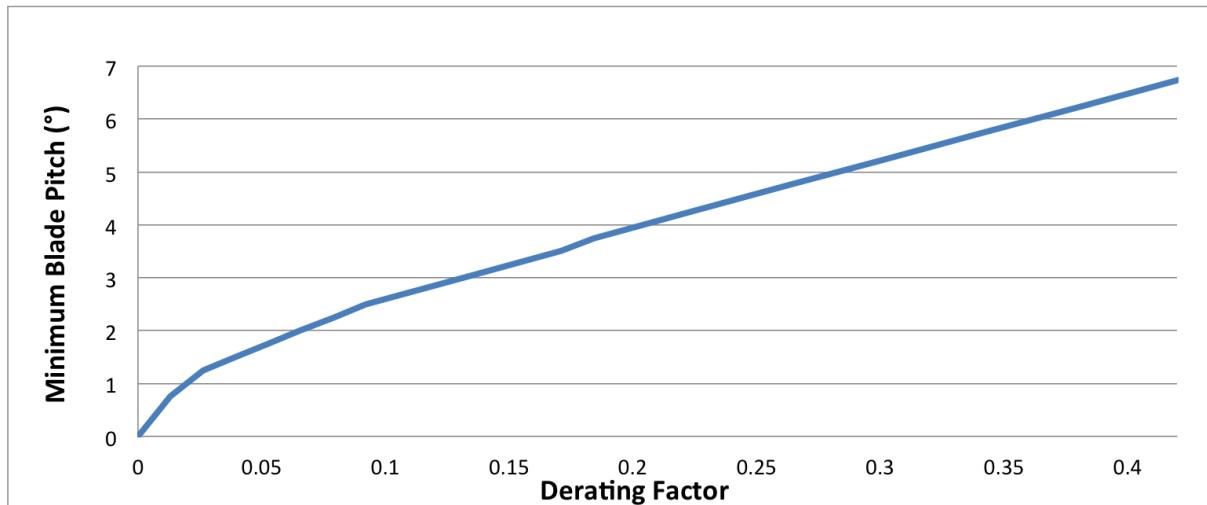


Figure 4.5. Effect of sudden rating changes on blade root bending moment.

4.3 Effect of Derating on Dynamic Turbine Response

This section investigates the relationship between derating and the dynamic turbine response. In particular we are interested in determining how derating affects the structural loading and rotor overspeed a turbine experiences when subjected to a large gust. To study these relationships a series of simulations were run subjecting a derated NREL 5-MW turbine to an extreme operating gust at a variety of mean wind speeds and a variety of derating factors. The series of simulations included mean wind speeds of 6 m/s, 8 m/s, 10 m/s, 12 m/s, 14 m/s, 16 m/s, 18 m/s, and 20 m/s, as well as derating factors of 0%, 5%, 10%, 15%, 20%, 25%, and 30%. For all simulations the turbine is operating in uniform, constant wind when it experiences

an extreme operating gust (as defined by IEC 61400-1 [36] for a class 1 turbine in category A turbulence).

Figure 4.6 shows the relationship between derating factor and average power production for several of the mean wind speeds simulated. As expected, there is a 1 to 1 relationship between the derating factor and the reduction in power generation. Each 1% of derating corresponds to a 1% decrease in power production. The other wind speeds simulated also showed this same relationship.

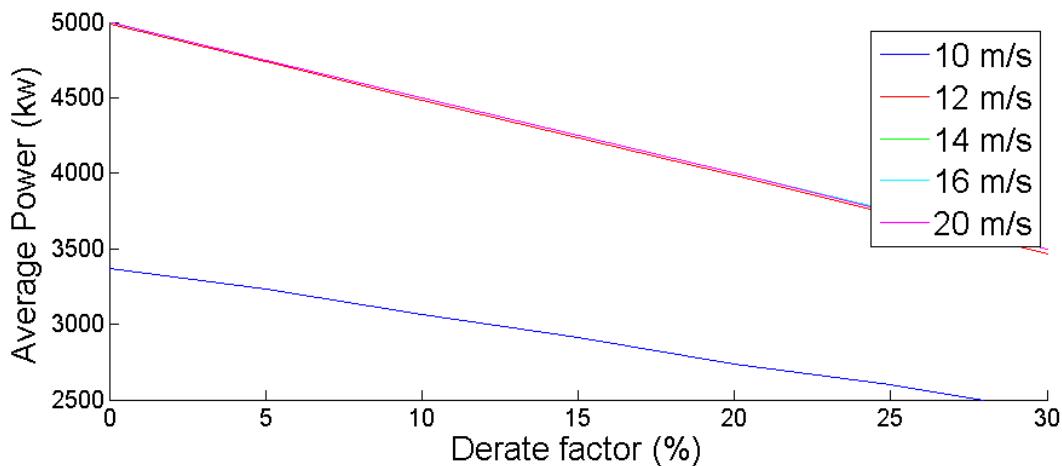


Figure 4.6. Effect of derating on avg. power for EOGs at various wind speeds.

Figure 4.7 shows the relationship between derating factor and the maximum tower base fore-aft bending moment for several of the mean wind speeds simulated. The largest tower bending moments occur in simulations where the mean wind speed is equal to the rated wind speed of 12 m/s. The maximum tower bending moment decreases slowly and steadily as the mean wind speed is increased above 12 m/s. The maximum tower bending moment also decreases steadily, but more precipitously, as the mean wind speed decreases below 12 m/s. We also notice that the relationship between derating factor and maximum tower bending moment appears nearly linear. On average, each 1% of derating corresponds to a 1.25% decrease in maximum tower base fore-aft bending moment.

Figure 4.8 shows the relationship between derating factor and the maximum blade root bending moment (BRM). For low derating factors the highest loads occur

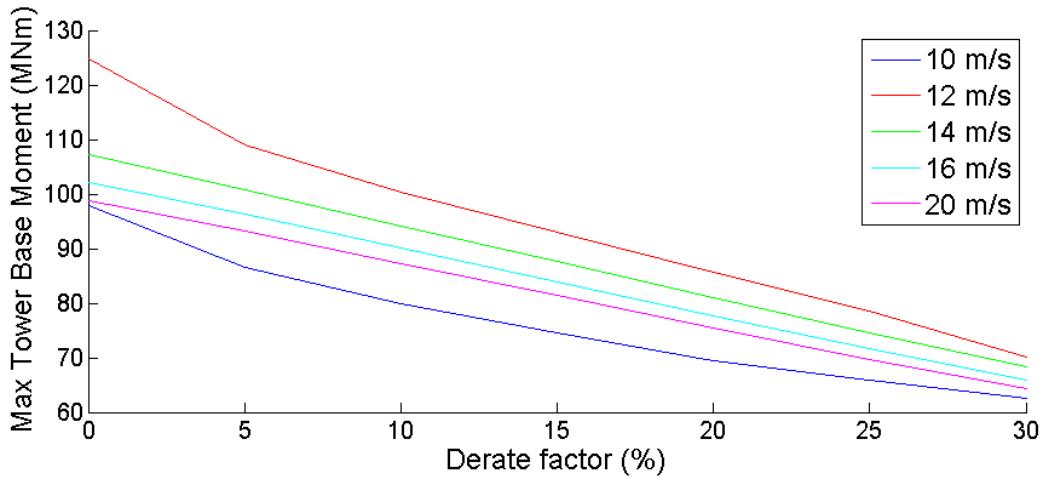


Figure 4.7. Effect of derating on max tower moment for EOGs at various wind speeds.

when the mean wind speed is equal to the rated wind speed of 12 m/s. The maximum loads gradually decrease as the mean wind speed either increases or decreases away from the rated wind speed. However, at higher derating factors all of the test cases at or above the rated wind speed (12 m/s - 20 m/s) appear to converge. The relationship between derating factor and maximum load reduction appears nearly linear, but the slope varies based on wind speed. The largest load reductions are seen in 12 m/s wind, where each 1% of derating corresponds to a 1.6% decrease in max BRM. The smallest load reductions are seen in 20 m/s wind, where each 1% of derating corresponds to a 0.6% decrease in max BRM.

Figure 4.9 shows the relationship between derating factor and the maximum rotor speed, while Figure 4.10 shows the corresponding maximum rotor overspeed. We see in these figures that the simulations with higher mean wind speeds have higher maximum rotor speeds and therefore higher maximum rotor overspeeds. On average each 1% of derating corresponds to a 0.7% decrease in max rotor speed.

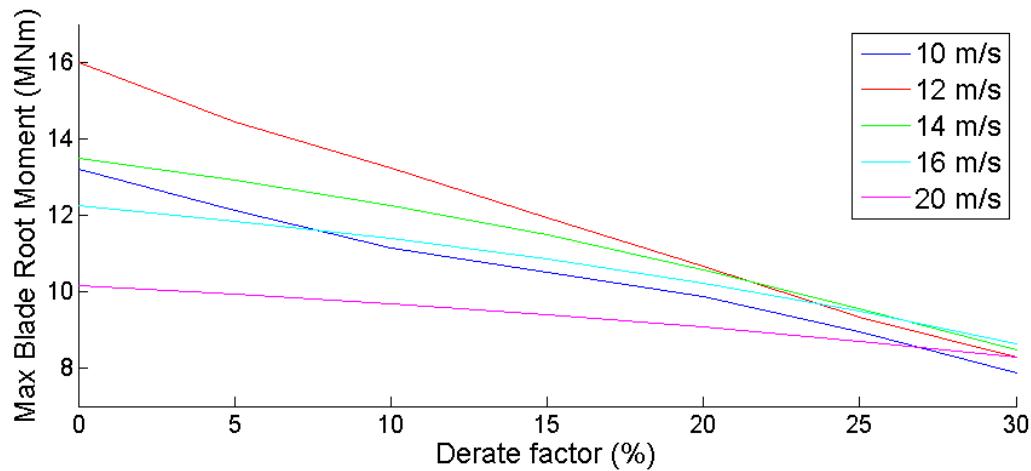


Figure 4.8. Effect of derating on max BRM for EOGs at various wind speeds.

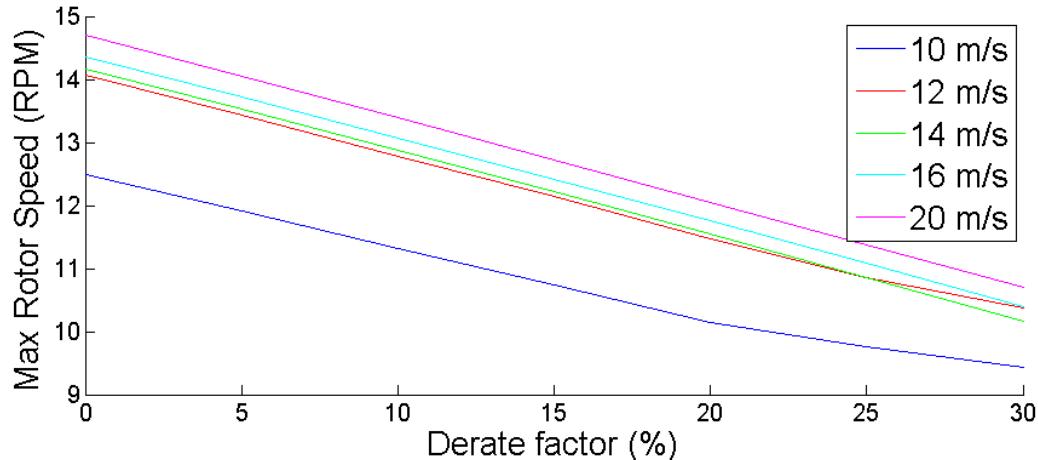


Figure 4.9. Effect of derating on max rotor speed for EOGs at various wind speeds.

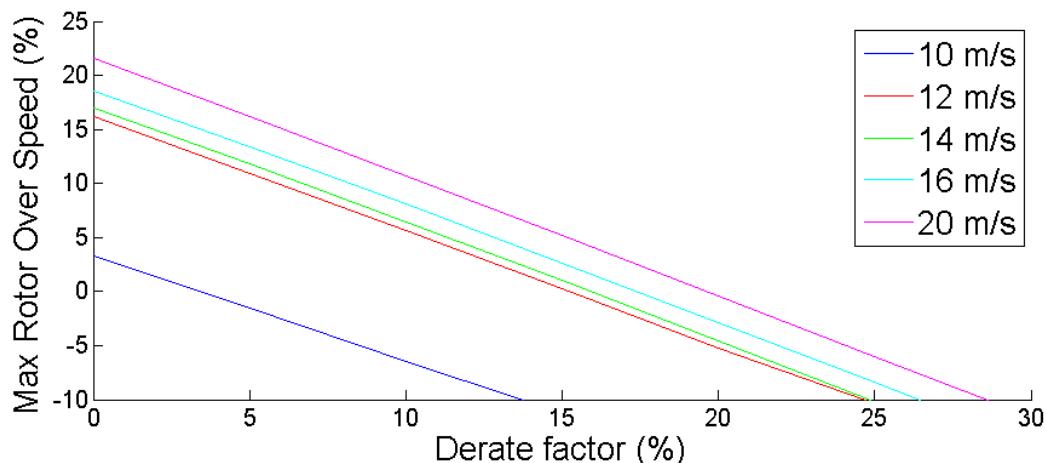


Figure 4.10. Effect of derating on max overspeed for EOGs at various wind speeds.

4.4 Transitioning Between Rated and Derated Operation

The previous subsection demonstrated that operating in a derated state can significantly reduce the structural loading and rotor overspeeds experienced by a turbine. However, the control scheme proposed in this chapter uses selective derating. In this control scheme the turbine will be derated when reductions in loading and rotor speed are necessary, but the turbine will operate normally when those reductions are not needed. To implement this control scheme we must understand how the turbine behaves while it is transitioning into and out of derated operation. This subsection examines the transient behavior of the turbine during these transitions and how undesirable transient behavior can be mitigated.

When the rating of a turbine is abruptly changed the turbine exhibits undesirable transient behavior. This transient behavior is illustrated by Figures 4.11 through 4.13. The figures show a test case in which the turbine operates in a steady, uniform 16 m/s wind. At $t = 100$ seconds the turbine is abruptly derated by 20%, then at 200 seconds the turbine is abruptly returned to its full rating. A 20% derating corresponds to a rotor speed reduction from 12.1 RPM to 9.68 RPM. In Figure 4.11 we see that when the turbine is abruptly derated the rotor undershoots the desired rotor speed by 0.94 RPM then oscillates before settling to within 2% of the desired rotor speed at approximately 130 seconds. When the turbine is abruptly returned to full rating it exhibits similar behavior, overshooting the desired rotor speed by 1.62 RPM (a 13.4% overshoot) before oscillating and settling to within 2% of the desired rotor speed at approximately 230 seconds.

The tower fore-aft bending moment also exhibits undesirable oscillations with settling times of approximately 30 seconds (Figure 4.12). These oscillations appear to have both a higher frequency component and a lower frequency component. The higher frequency oscillation (2 rad/s) corresponds to the first natural frequency of the tower. The lower frequency oscillation is caused by oscillations in the collective blade pitch. When the turbine's collective pitch controller reduces the blade

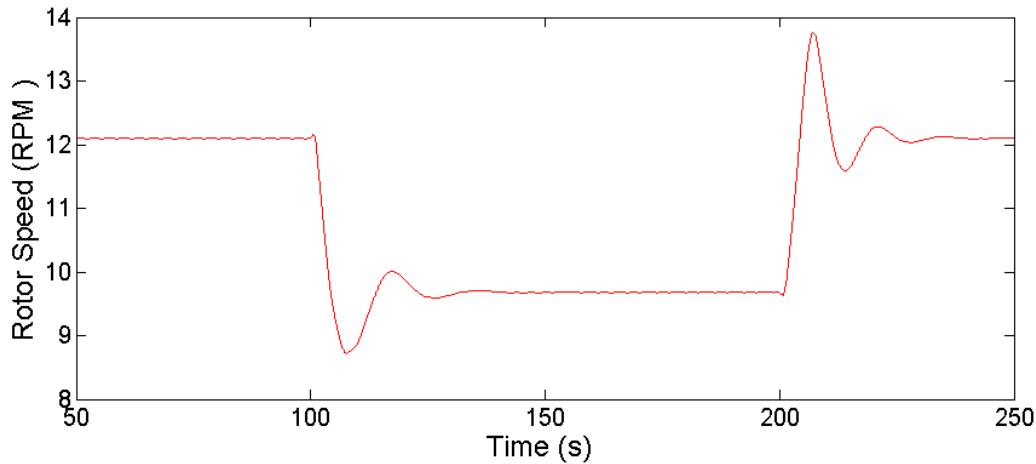


Figure 4.11. Effect of sudden rating changes on rotor speed.

pitch, in order to increase aerodynamic rotor torque and increase the rotor speed, it also increases the aerodynamic thrust on the rotor, which increases the fore-aft tower moment. Similarly, when the collective pitch controller increases blade pitch it reduces the fore-aft tower moment.

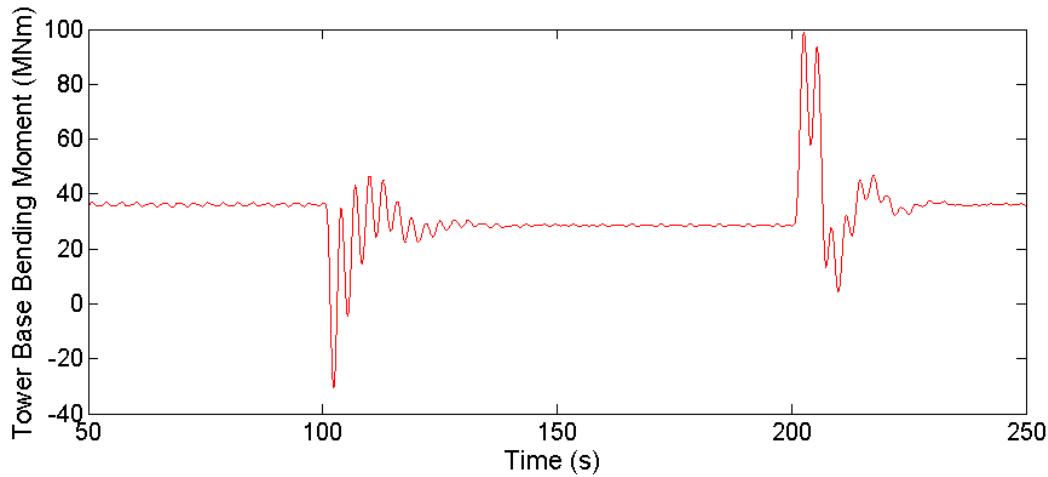


Figure 4.12. Effect of sudden rating changes on tower bending moment.

As Figure 4.13 shows, the blade root bending moment also exhibits undesirable transient behavior when the turbine abruptly transitions into or out of derated operation. However, the cyclic nature of the blade root loading makes it difficult to discern the frequency or settling time of the transient behavior. Undesirable oscillations, like those shown in Figures 4.11 through 4.13, were seen at all wind speeds.

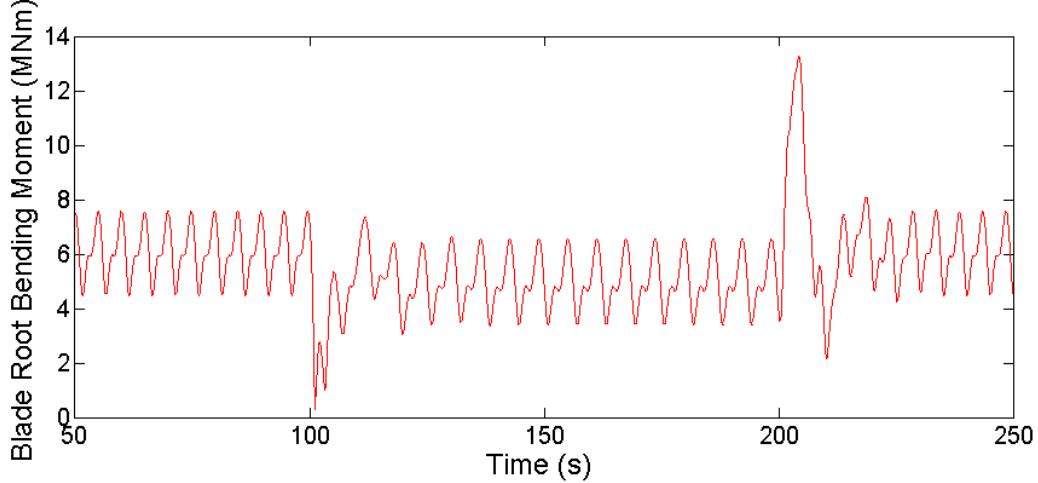


Figure 4.13. Effect of sudden rating changes on blade root bending moment.

We can reduce the excitation of undesirable oscillations by using a smoother transition into and out of derated operation. One way to smooth the transition is to pass derating commands through a low pass filter like the one described by the transfer function ($G_{filter}(S)$) shown in Equation 4.3. This low pass filter is a critically damped second order system with two poles at $-P_{filter}$. Passing derate commands through this filter turns abrupt derating commands into smoother transitions, as seen in Figure 4.14. The 98% settling time of the filtered command is determined by the value of P_{filter} and is given by $t_s = 5.8 \times P_{filter}$.

$$G_{filter}(S) = \frac{P_{filter}^2}{(S + P_{filter})^2} \quad (4.3)$$

Choosing $P_{filter} = 0.2$ gives filtered command settling time of 29.5 seconds, which is approximately equal to the settling time of the transient oscillations caused by abrupt changes in turbine rating. As Figures 4.15 through 4.17 show, using this filtered command to transition into and out of derated operation results in improved transient behavior while maintaining the same settling time. In Figure 4.15 we see that the filter eliminates the oscillations, overshoots, and undershoots in rotor speed. In Figure 4.16 we see that the filter eliminates the higher frequency oscillation. The lower frequency oscillation, caused by variations in the collective pitch angle, are still present but have reduced in magnitude. In Figure 4.17 we see that

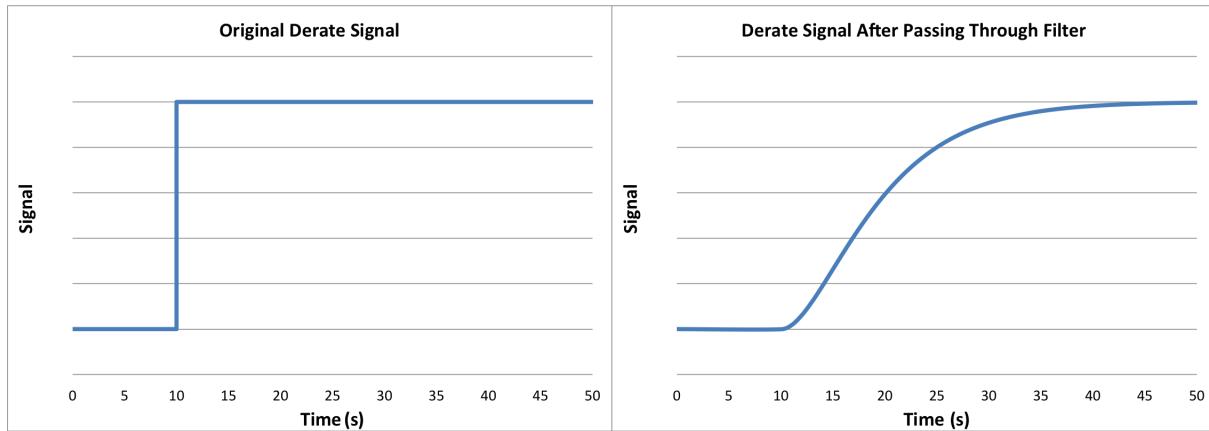


Figure 4.14. Smoothing effect of the derating input filter when $P_{filter} = 0.2$.

the filter also improves transient behavior in the blade root bending moment.

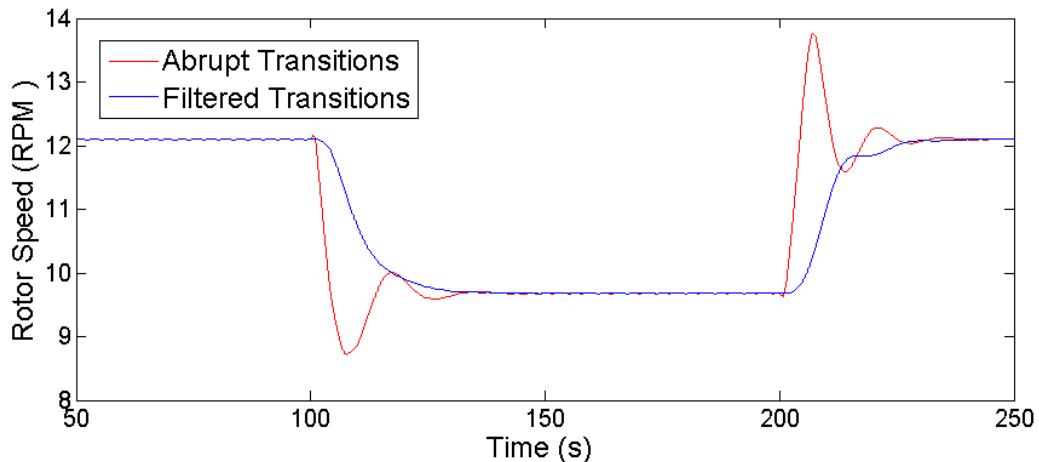


Figure 4.15. Improvements in rotor behavior from filtered derate commands.

In general, lengthening the settling time of the filter reduces undesirable transient oscillations and reduces the magnitude of any overshoots or peak loads associated with transitioning into or out of derated operation. Figures 4.18 through 4.20 show the relationship between filter settling time and the maximum overshoots observed in the derating transition simulations. In practice the maximum acceptable settling time would be limited by the time it takes a gust of wind to travel from the upwind turbine to the downwind turbine. This time will depend on both the wind speed and the layout of the wind farm, however, some simple calculations can be made to help choose a reasonable filter settling time. The NREL 5-MW turbine

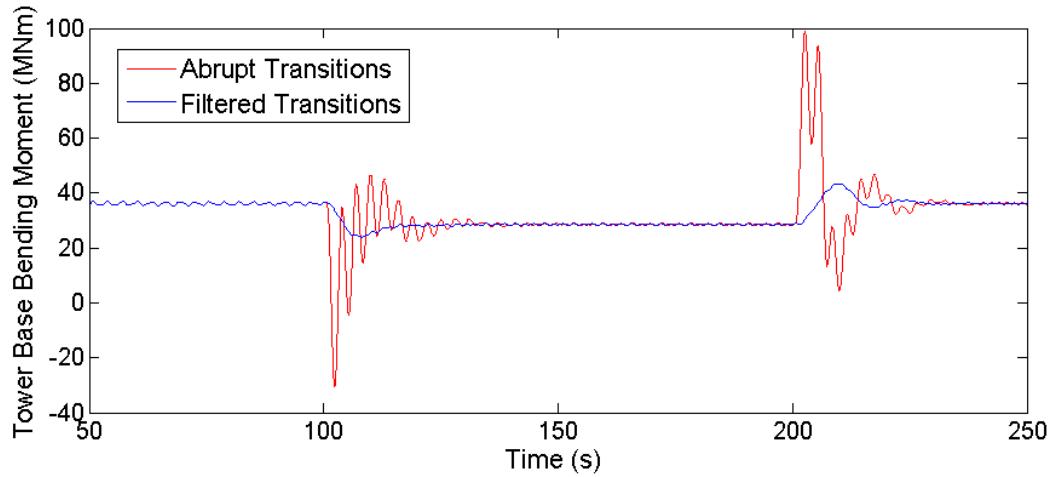


Figure 4.16. Improvements in tower bending moment from filtered derate commands.

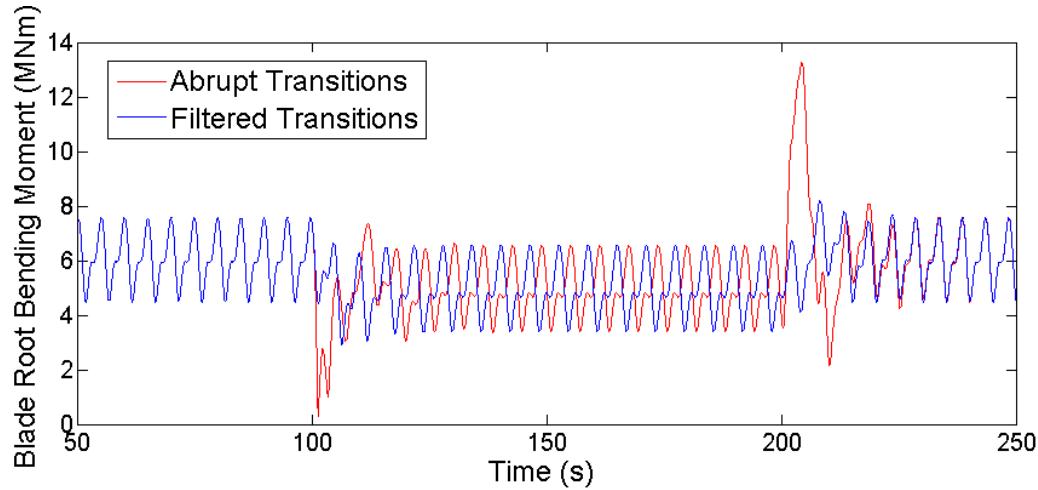


Figure 4.17. Improvements in blade root moment from filtered derate commands.

has a rotor diameter of 126 meters. If the turbines are spaced 10 rotor diameters apart then in 12 m/s wind, the wind speed at which loads are highest and this control scheme would likely be most beneficial, a gust would travel between turbines in approximately 105 seconds. In 25 m/s wind, the highest wind speed in which the NREL 5-MW turbine will operate at all, the gust would travel between turbines in approximately 50 seconds. If the turbines are spaced 5 rotor diameters apart then a gust would travel between turbines in approximately 53 seconds for 12 m/s wind and in approximately 25 seconds for 25 m/s wind.

A low pass filter, given by equation 4.3 with $P_{filter} = 0.2$, will be used to smooth

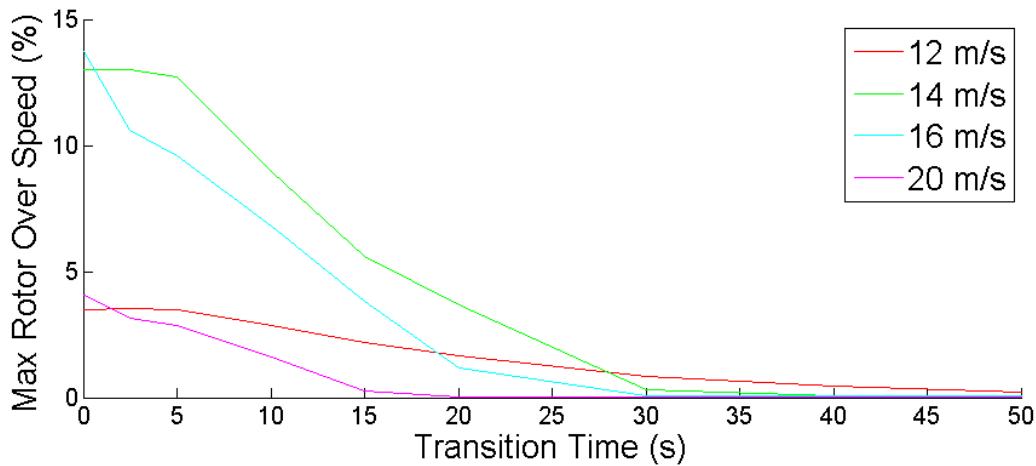


Figure 4.18. Effect of input filter settling time on rotor overspeed.

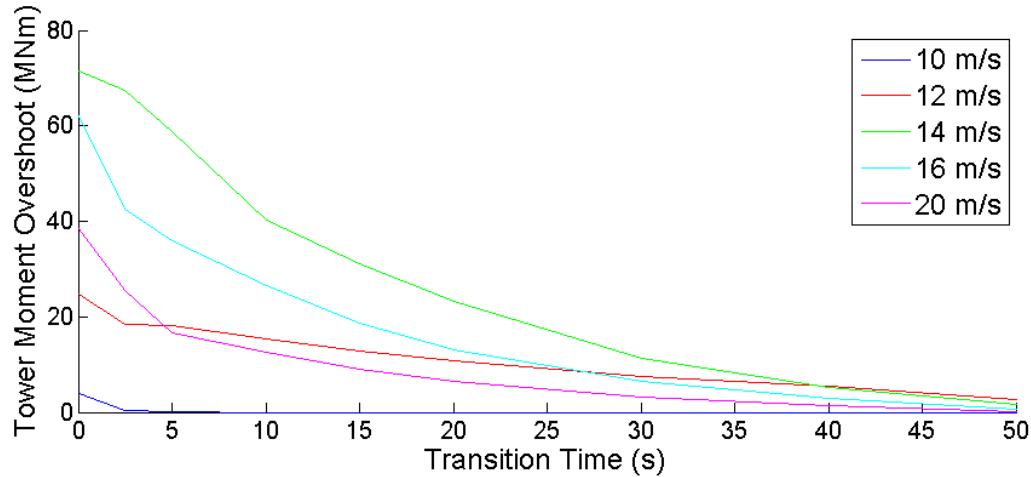


Figure 4.19. Effect of input filter settling time on peak tower bending moment.

the turbine transitions into and out of derated operation. $P_{filter} = 0.2$ corresponds to a filter settling time of 29.5 seconds. Though this filter could potentially cause problems for tightly packed turbines in very high wind speeds a filter settling time of 29.5 seconds would be practical for almost all operating conditions of the NREL 5-MW turbine. As seen above, this filter also gives very good transient behavior when the turbine transitions into and out of derated operation.

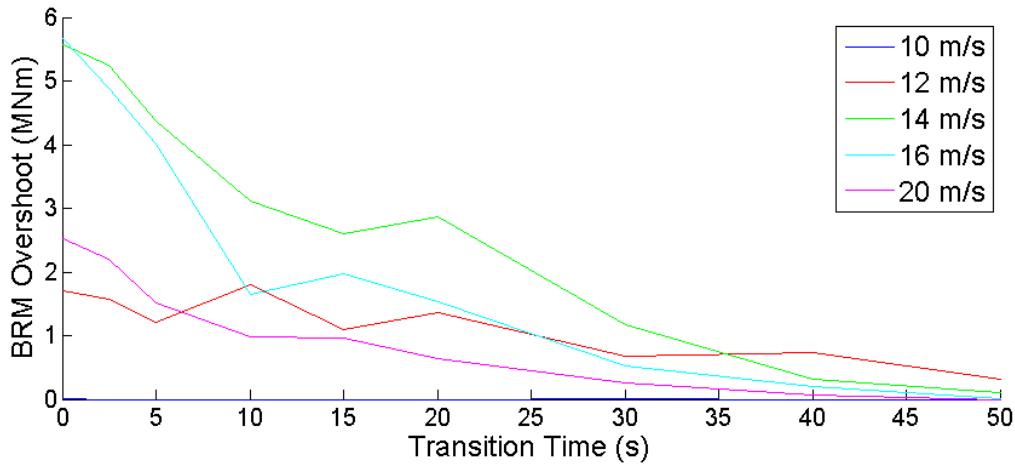


Figure 4.20. Effect of input filter settling time on peak blade root moment.

4.5 Control System Design

As Figure 4.21 illustrates, a plant level controller will monitor the behavior of the upwind turbine and derate the downwind turbine when necessary. To determine when derating is necessary, the plant level controller will monitor the rotor speed of the upwind turbine. If the upwind turbine rotor speed exceeds some pre-defined threshold the plant level controller will interpret that as an extreme gust event and derate the downwind turbine. This threshold should be chosen based on the range of rotor speeds a turbine experiences during nominal operation, and how aggressive a plant operator chooses to be with this selective derating strategy. Therefore, in practice this threshold may be site specific.

For the simulations carried out in the following sections the threshold to initiate derating is set to 13.31 RPM (a 10% overshoot). Section 2.2 describes a set of 66 simulations of the NREL 5-MW turbine in turbulent wind. That data set contains 7 hours of simulated turbine operation in wind speeds between 12 m/s and 22 m/s, wind speeds for which the turbine is operating in region 3 control and attempting to track the rated rotor speed of 12.1 RPM. In that 7 hours of data the rotor speed exceeds 5% overshoot only 0.44% of the time and never exceeds 6.76% overshoot. A 10% threshold to initiate derating should be sufficient to identify extreme gust events and roughly splits the difference between what the turbine sees during

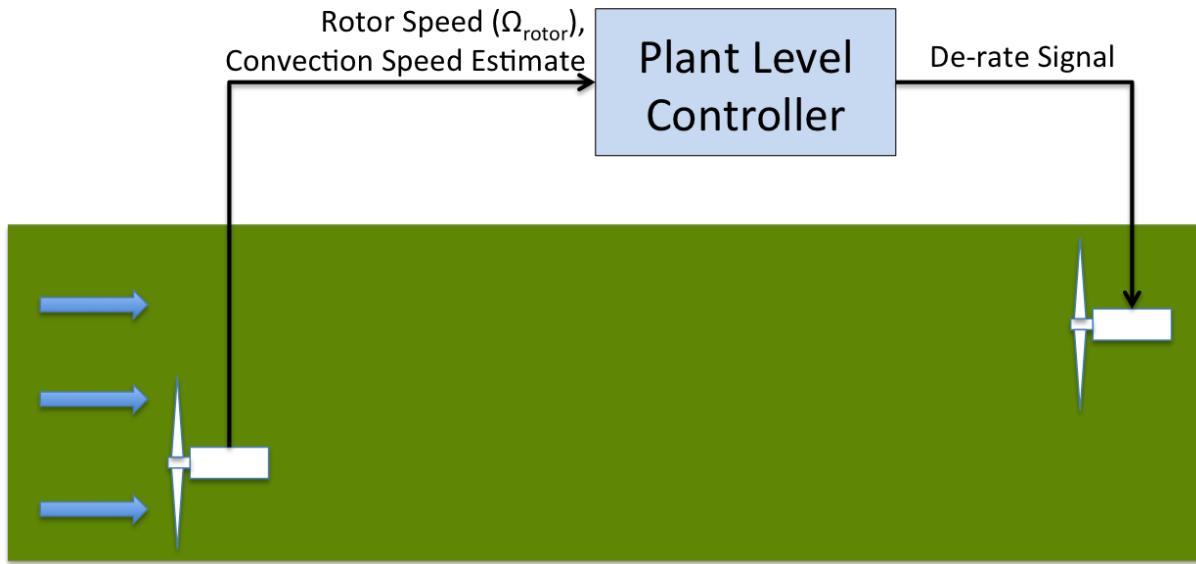


Figure 4.21. Control system overview.

normal operation and a 15% overspeed, which could potentially cause a overspeed shutdown. When derating is initiated, the downwind turbine will be derated by the maximum rotor overspeed observed in the extreme gust event minus 5%.

As shown in section 4.3, derating the downwind turbine will reduce the structural loads and rotor overspeed caused by the extreme gust event. As a result the downwind turbine will experience reduced structural damage and avoid a possible overspeed shutdown when the extreme gust arrives. Once the extreme gust has passed the downwind turbine the downwind turbine will be returned to full rated operation. Several methods can be used to determine when the downwind turbine should be derated ($T_{startDerate}$) and when it is safe for the downwind turbine to return to full rated operation ($T_{endDerate}$). Two of those methods are examined in the following paragraphs.

One method of determining when to derate the downwind turbine is to use an estimate of the gust's convection speed. The time it takes for a gust of wind to travel from the upwind turbine to the downwind turbine is given by $t = D/U_{conv}$, where D is the downwind distance between the turbines and the convection speed U_{conv} is the rate at which wind speed fluctuations propagate downwind. As discussed in

Section 2.5, the convection speed can be estimated using blade pitch, rotor speed, generator torque, and known dynamics of the turbine.

Large rotor overspeeds occur when the turbine is operating at or above the rated wind speed. For the NREL 5-MW turbine that means that large overspeeds will occur when the wind speed is between 12 and 25 m/s. Section 2.5 analyzed 7 hours of simulated turbine operation in those wind conditions. With a 60 second averaging time, the mean error in convection speed estimates was approximately 3.2% and the largest error seen in a convection velocity estimate was approximately 5.6%. To improve this control scheme's robustness to errors in convection velocity estimation, we can base our derate timing on a larger error than was seen in those results. If the plant level controller begins and ends the derate signal based on equations 4.4 and 4.5 then the control system will accommodate extreme wind events traveling up to 20% faster or 20% slower than the convection velocity estimate. In these equations t_{Event} is the time at which the upwind turbine experiences an unacceptably large overspeed, t_{conv} is the estimated time it will take for the gust of wind to travel from the upwind turbine to the downwind turbine, and t_s is the time required for the turbine to smoothly transition into derated operation. This system could experience problems if the upwind turbine experienced an overspeed shutdown, as convection velocity estimates would no longer be available.

$$t_{DR_start} = t_{Event} + 0.8 \times t_{conv} - t_s \quad (4.4)$$

$$t_{DR_end} = t_{Event} + 1.2 \times t_{conv} \quad (4.5)$$

Another method of determining when to derate the downwind turbine is to develop worst case scenarios based on the operating conditions of the turbine. For the NREL 5-MW turbine we have determined that large rotor overspeeds will occur in wind speeds between 12 m/s and 25 m/s. If the downwind turbine is 10 rotor diameters (1260 m) behind the upwind turbine and the convection speed is somewhere between 12 and 25 m/s it would take a gust of wind between 105 and 50.4 seconds to

travel from the upwind turbine to the downwind turbine. If the turbine is derated from 50.4 seconds to 105 seconds after the upwind turbine experiences an unacceptable overspeed, the downwind turbine will be in derated operation when the gust arrives no matter what the wind speed is. To add more safety margin we can base our worst case scenario on an even wider range of convection velocities. If the transition to derated operation is based on a 30 m/s convection time and the transition out of derated operation is based on an 8 m/s wind, the derate signal will be based on equations 4.6 and 4.7. In these equations 42 seconds is the time required for a gust traveling 30 m/s to reach the downwind turbine and 157.5 seconds is the time required for a gust traveling 8 m/s to reach the downwind turbine. A similar calculation can be done for other turbines and/or other turbine to turbine spacing. Though this method derates the downwind turbine for longer than is strictly necessary it is simple, robust, and does not require an estimate of the convection velocity.

$$t_{DR_{start}} = t_{Event} + 42\text{seconds} - t_s \quad (4.6)$$

$$t_{DR_{end}} = t_{Event} + 157.5\text{seconds} \quad (4.7)$$

4.6 System Performance

The following sections examine the performance of the selective derating feed forward control system. A two turbine system is simulated in FAST using the method described in Section 3.2. The upwind turbine, with a conventional closed loop turbine controller, is simulated first. The dynamic response of that turbine is recorded and post processed to generate feed forward control signals. This post processing simulates the plant level controller. Finally the downwind turbine, with feed forward control, is simulated. By comparing the dynamic behavior of both simulations we determine the effect of the feed forward control system.

4.6.1 Response to IEC Extreme Operating Gust

The first simulation case is a uniform wind field with an extreme operating gust (EOG). The extreme operating gust has been defined according to IEC 61400-1 [36] for an NREL 5-MW turbine operating in 16 m/s wind. This is the same test case used to analyze the performance of feed forward optimal pitch control in Sections 3.4.1 and 3.5.1, however the EOG has been shifted to occur 200 seconds after the start of the simulation, as shown in Figure 4.22. This shift will ensure that transient derating behavior will not overlap with any of the transient turbine start up behavior at the beginning of the simulation.

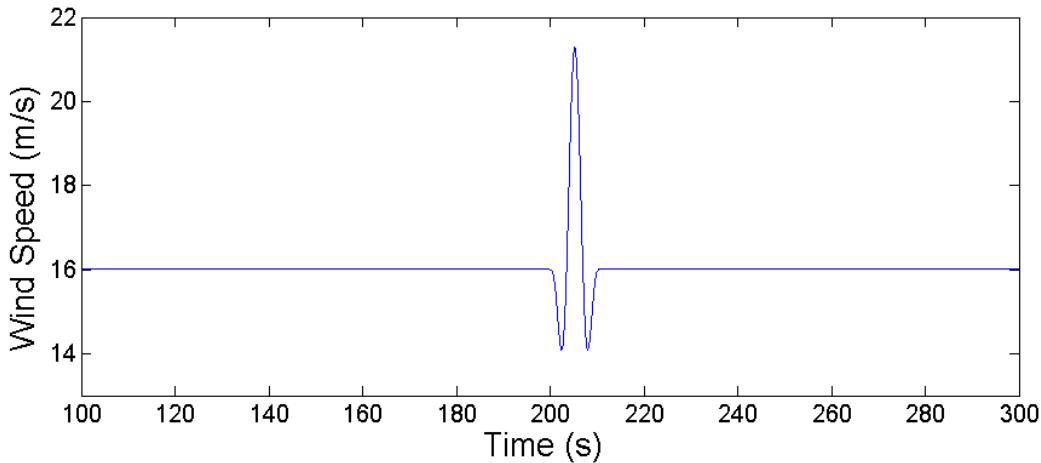


Figure 4.22. Extreme operating gust (EOG).

Figures 4.23 through 4.26 show how feed forward selective derating control affects the rotor speed, tower base fore-aft bending moment, blade root moment, and power generation of the downwind turbine. Table 4.6.1 quantifies and summarizes several important performance metrics. Two methods were used to time the derating of the downwind turbine. As discussed in Section 4.5, the first method uses an estimate of the gust convection speed to determine when to derate the turbine. The second, more conservative, method derates the downwind turbine over a longer period of time, but does not require an estimate of the convection speed. Both methods have an identical effect on rotor overspeed and peak structural loads, but they affect power generation differently.

Figure 4.23 shows that the upwind turbine (or a downwind turbine without feed forward control) experiences an 18.76 % rotor overspeed due to the EOG, potentially enough to cause an emergency shutdown of the turbine, but the feed forward controller reduces the rotor overspeed in the downwind turbine to 4.96%. In Figures 4.24 and 4.25 we see that both the upwind and downwind turbines experience large structural loads due to the EOG, but the loads experienced by the downwind turbine are smaller. The feed forward controller has reduced the maximum tower base fore-aft bending moment by 21.9% and the maximum blade root bending moment by 14.7%.

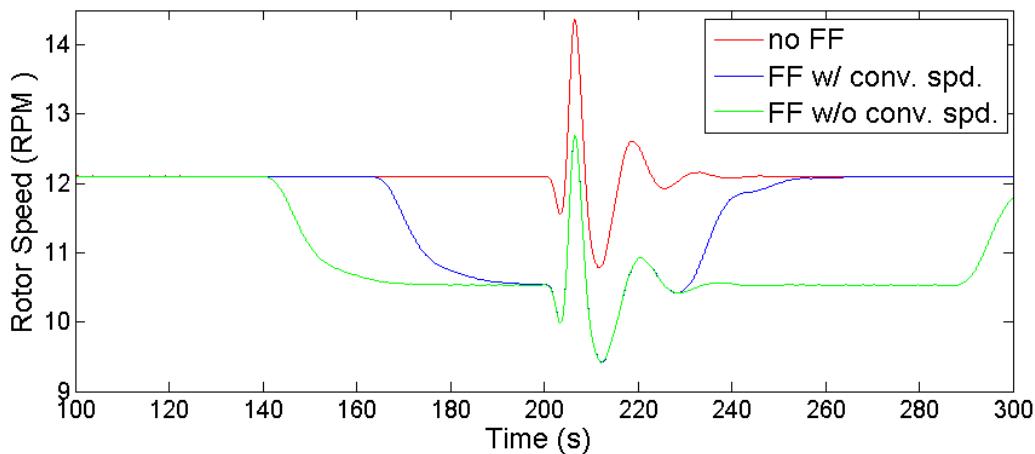


Figure 4.23. Effect of feed forward derating control on rotor speed caused by a 16 m/s EOG.

These figures also illustrate that the system is robust to differences in the convection speed or errors in convection speed estimates. As discussed in section 3.2, the simulation method used here assumes Taylor's frozen turbulence hypothesis, which implies that the true convection speed is equal to the mean wind speed. This assumption causes the EOG to reach the downwind turbine at exactly 200 seconds. However, if the EOG convects faster or slower than the mean wind speed the arrival time will shift. Similarly, if there is an error in the convection velocity estimate the window of derated operation will shift. As we can see from Figures 4.23 through 4.25 small shifts will not affect the performance of the feed forward controller because the downwind turbine will still be in derated operation when the EOG arrives.

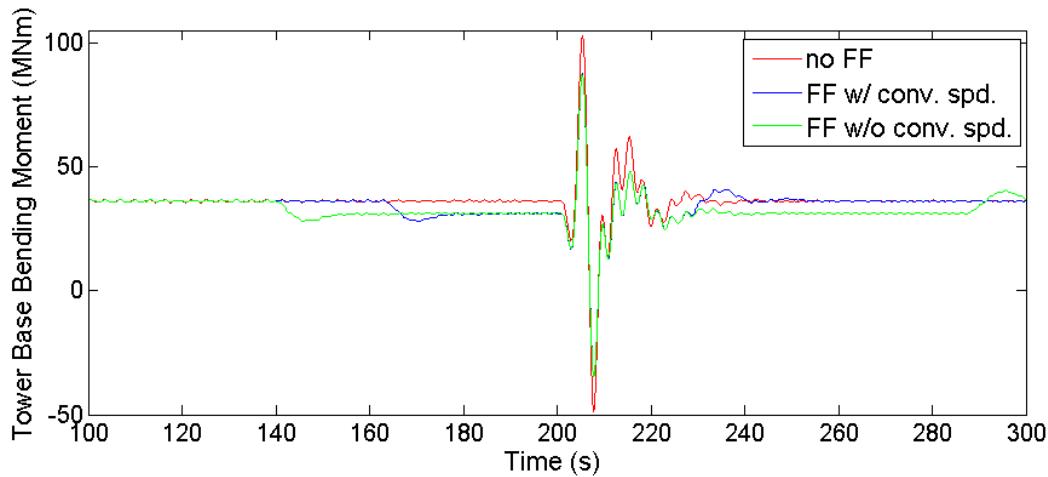


Figure 4.24. Effect of feed forward derating control on tower base fore-aft moment caused by a 16 m/s EOG.

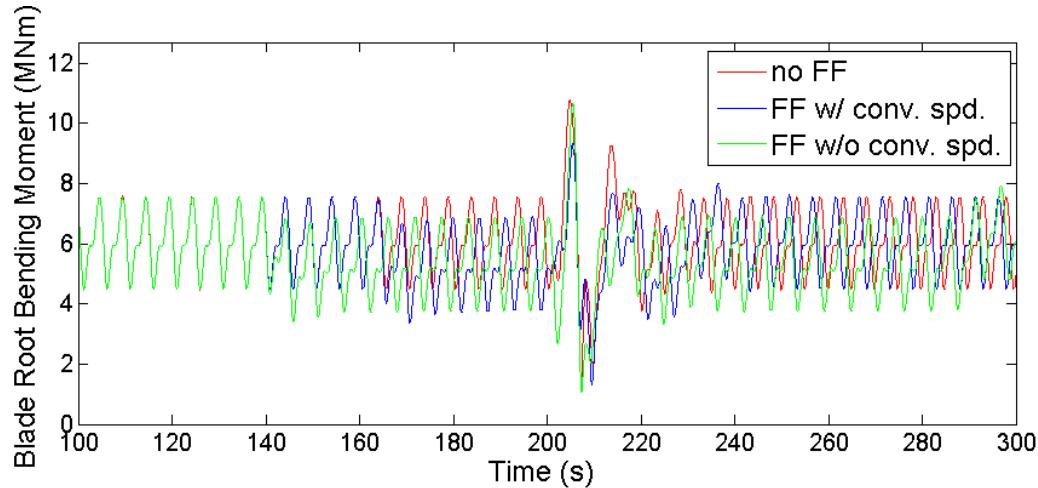


Figure 4.25. Effect of feed forward derating control on BRM caused by a 16 m/s EOG.

Figure 4.25 shows power generation. As expected, derating the downwind turbine results in reduced power generation. When the derate timing is based on a convection velocity estimate it results in a loss of 11.5 KWh. If energy is sold at 8 cents per kWh that results in \$0.92 of lost revenue. When a convection velocity estimate is not used for derate timing the downwind turbine remains derated longer. This results in a loss of 26.48 kWh and lost revenue of approximately \$2.12. These simulations appear to show that turbines with feed forward selective derating control will generate less energy, however in real world operation this derating

strategy may result in a significant increase in energy generation. The reason for this discrepancy is that these simulations do not model emergency rotor overspeed shutdowns. The turbine without feed forward control experiences a rotor overspeed of 18.76%. If this caused a 10 minute emergency shutdown it would result in a loss of approximately 833 kWh and lost revenue of approximately \$67, significantly more than the lost revenue caused by briefly derating the turbine.

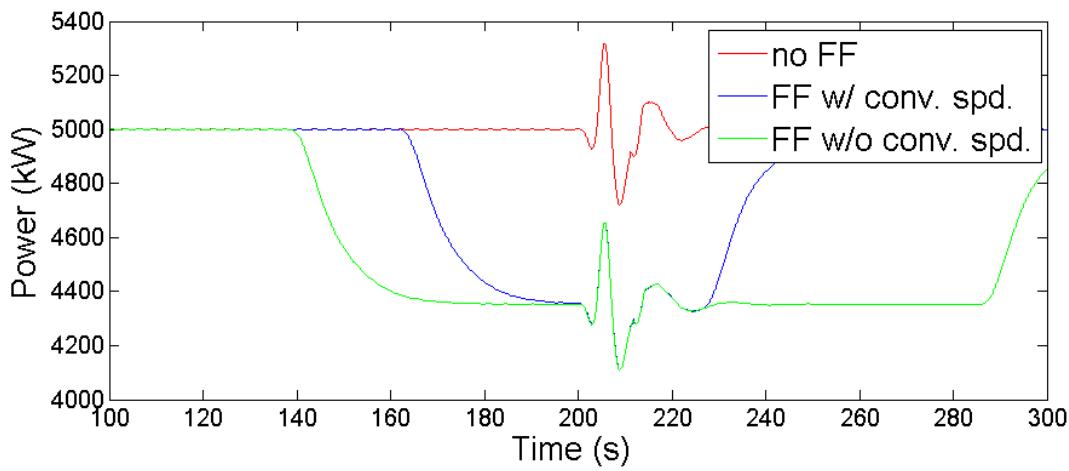


Figure 4.26. Effect of feed forward derating control on power generation caused by a 16 m/s EOG.

	Max Tower Base Moment (MNm)	Decrease	Max Blade Root Moment (MNm)	Decrease	Max Overspeed (%)	Energy Gen. (kWh)
No FF Control	102.7	-	10.77	-	18.76	831.8
FF w/ Conv. Speed	87.46	14.9%	9.43	12.4%	4.96	820.2
FF w/o Conv. Speed	87.34	15.0%	10.66	1.0%	4.96	805.3

Table 4.3. Effect of FF Control on dynamic response to 16 m/s EOG.

4.6.2 Response to Turbulent Wind With Large Gust

The Extreme Operating Gust test case provides insight into how selective derating can reduce loading and overspeeds. However, constant wind speeds like those before and after the EOG in the previous section, are rare in the real world. When a turbine experiences a large gust it is most likely part of a turbulent wind field. In this section we examine the effect of selective derating feed forward control on a two turbine system experiencing turbulent wind with a large gust. As discussed in Section 4.5, the NREL 5-MW turbine is vulnerable to large gust induced overspeeds when operating in region 3 control, which occurs in wind speeds between 12 m/s and 25 m/s. As Section 4.3 illustrates, structural loading and overspeeds vary across that range of wind speeds. In an effort to capture these variations, turbulent wind speeds with three different mean wind speeds were simulated. Figures 4.27 through 4.31 illustrate the 16 m/s turbulent wind test case, while Tables 4.4 through 4.6 quantify and summarize performance metrics for all test cases. For the sake of space, the 12 m/s and 20 m/s turbulent wind test cases are not illustrated with figures. However, the behavior seen in the 12 m/s and 20 m/s test cases is similar to what is seen in the 16 m/s case.

Figure 4.27 shows the wind speed for 200 seconds of the 16 m/s turbulent test case. As the figure shows, the turbines are subjected turbulent fluctuations in wind speed throughout the simulation with a large gust at 200 seconds. The turbulent fluctuations were generated by TurbSim and are statistically realistic, however these wind speeds are applied uniformly over the whole rotor. Due to the limitations of FAST and TurbSim it was not possible to simulate full field turbulence with a large gust, so the spatial variations in wind speed that would be observed across the rotor in a real turbulent wind field could not be captured in this simulation.

Figures 4.28 through 4.31 show how feed forward derating control affects the rotor speed, tower base fore-aft bending moment, blade root moment, and power generation of the downwind turbine for 16 m/s turbulent wind with a large gust. As in Section 4.6.1 two methods were used to time the derating of the downwind

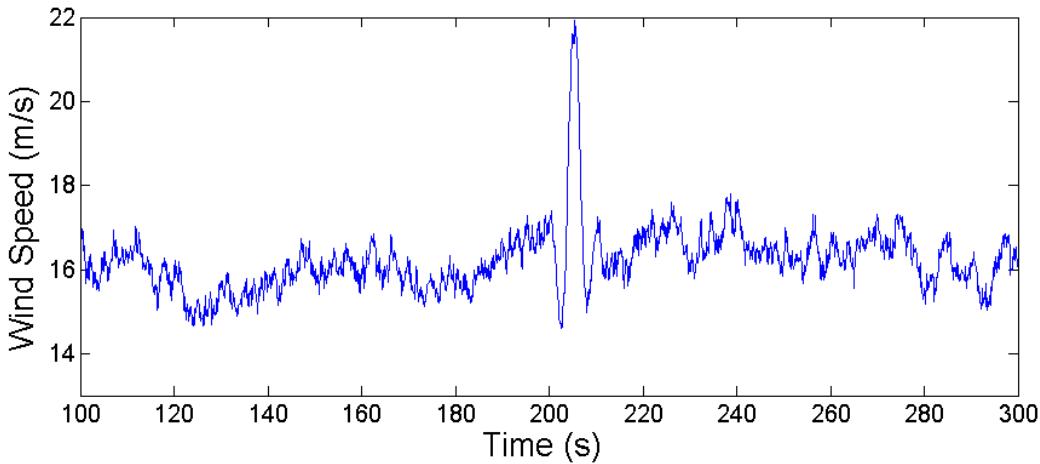


Figure 4.27. Turbulent wind with 16 m/s mean wind speed and large gust.

turbine.

Figure 4.28 shows that the upwind turbine experiences many small overspeeds throughout the simulation and experiences a large overspeed of 17.77% due to the large gust. This large overspeed could potentially cause an emergency overspeed shutdown of the upwind turbine. Selective derating feed forward control reduces the maximum overspeed experienced by the downwind turbine to approximately 5%, which is in the normal range of operation for this turbine. In Figure 4.29 we see that both the upwind and downwind turbines experience large tower base fore-aft bending moments due to the large gust, but the loads experienced by the downwind turbine are smaller. The feed forward controller has reduced the maximum tower base fore-aft bending moment by approximately 11%. As in the previous section, we see that this system would be robust to changes in the arrival time of the gust.

Figure 4.30 shows that derating the downwind turbine for the 16 m/s test case has resulted in a higher peak blade root moment (BRM). Though this result is initially surprising, it can be explained when the factors contributing to the peak load are examined more closely. We can see from Figure 4.30 that BRM has a large cyclical component. These cyclic changes in BRM have a frequency of 12 RPM, which corresponds to the rotational speed of the rotor, and are caused primarily by gravitational loads on the turbine blade. Section 4.3 showed that derating reduces BRM,

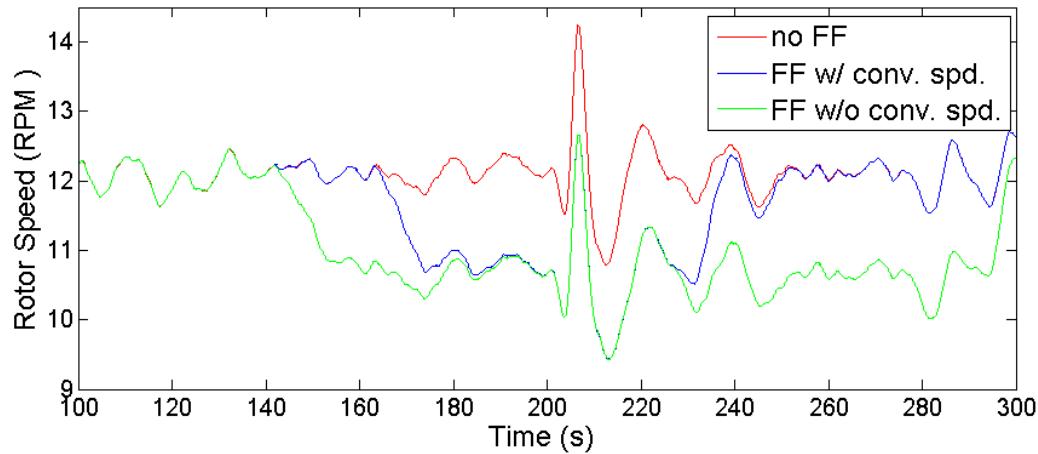


Figure 4.28. Effect of feed forward derating control on rotor speed caused by 16 m/s turbulence with large gust.

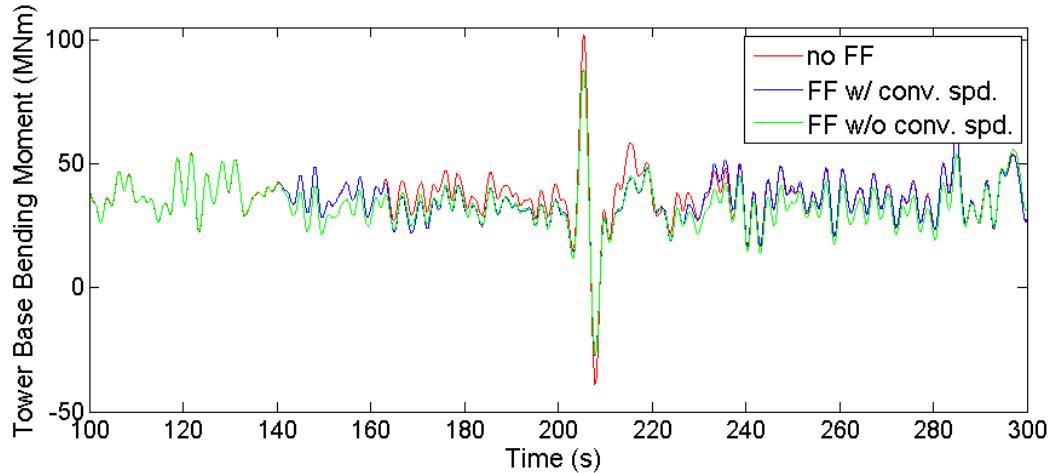


Figure 4.29. Effect of feed forward derating control on tower base fore-aft moment caused by 16 m/s turbulence with large gust.

and we see from Figure 4.30 that derating reduces BRM prior to the arrival of the gust. However, the gust induced loads interact with the cyclic gravitational loads. For the upwind turbine (without derating) the gust arrives at an opportune time. A downswing in the cyclic gravitational loading largely cancels out the gust induced BRM and the effect of the gust is almost unnoticeable. For the downwind turbine, the derating process has reduced the rotational speed of the rotor. As a result, the downwind turbine is at a different and less opportune point in its rotation when the gust arrives. In real world applications the timing of a large gust would be random.

Because derating reduces the BRM induced by the gust, we would expect derating to reduce peak BRM more often than not. However, as this result illustrates, derating does not always result in reduced peak BRM loads.

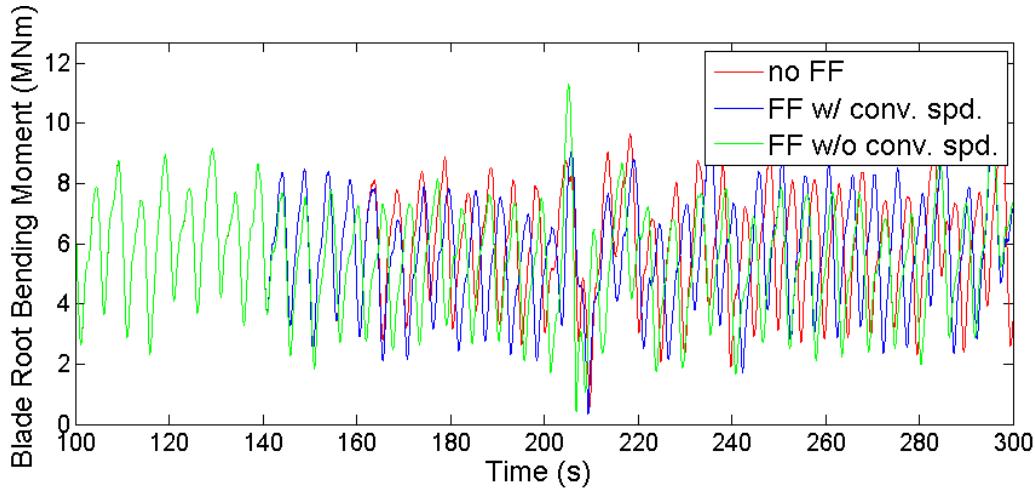


Figure 4.30. Effect of feed forward derating control on BRM caused by 16 m/s turbulence with large gust.

Figure 4.31 shows power generation. As expected, derating results in reduced power generation. Derating based on a convection speed estimate results in 10.7 kWh of lost energy production. The more conservative strategy of derating without a convection speed estimate results in 24.9 kWh of lost energy production. However, as stated before, these derating strategies are intended to avoid emergency overspeed shut downs. An emergency overspeed shutdown would likely result in a much larger loss of energy.

Tables 4.4 through 4.6 quantify and summarize performance metrics for all three test cases: 12 m/s, 16 m/s, and 20 m/s. In an attempt to capture the overall wear and tear on the tower base and blade root, Damage Equivalent Loads (DEL) are reported instead of peak loads. Though the DEL, overspeed, and energy generation values vary between the test cases, the trends are the same. Selective feed forward derating reduces maximum overspeeds from more than 15%, which could cause an emergency overspeed shut down, to approximately 5%. Tower root bending DELs are significantly reduced for all test cases. Blade root moment DELs are reduced in some test cases, but increased in others. Energy production is reduced by tens

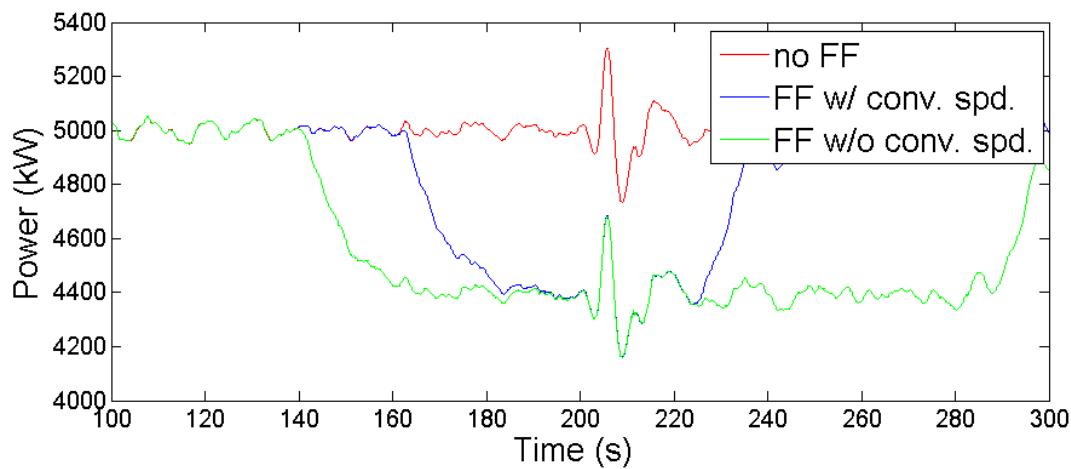


Figure 4.31. Effect of feed forward derating control on power generation caused by 16 m/s turbulence with large gust.

of kWh, which is equivalent to a few dollars of revenue.

	Tower Base DEL		Blade Root DEL		Max Overspeed	Energy Gen.
	(MNm)	Decrease	(MNm)	Decrease	(%)	(kWh)
No FF Control	78.8	-	6.74	-	15.79	812.1
FF w/ Conv. Speed	69.2	12.2 %	5.87	12.9%	4.46	800.9
FF w/o Conv. Speed	67.5	14.3%	5.16	23.2%	4.46	791.0

Table 4.4. Effect of FF Control on damage equivalent loads for a large gust in turbulent wind with 12 m/s mean wind speed.

	Tower Base DEL		Blade Root DEL		Max Overspeed (%)	Energy Gen. (kWh)
	(MNm)	Decrease	(MNm)	Decrease		
No FF Control	75.1	-	5.08	-	17.77	831.8
FF w/ Conv. Speed	67.2	10.5 %	5.14	-1.2%	5.04	821.1
FF w/o Conv. Speed	66.3	11.7%	5.44	-7.1%	4.88	806.9

Table 4.5. Effect of FF Control on damage equivalent loads for a large gust in turbulent wind with 16 m/s mean wind speed.

	Tower Base DEL		Blade Root DEL		Max Overspeed (%)	Energy Gen. (kWh)
	(MNm)	Decrease	(MNm)	Decrease		
No FF Control	91.1	-	5.08	-	22.89	831.8
FF w/ Conv. Speed	77.2	15.3 %	5.06	0.3%	5.04	818.8
FF w/o Conv. Speed	76.6	15.9%	5.20	-2.4%	5.12	798.5

Table 4.6. Effect of FF Control on damage equivalent loads for a large gust in turbulent wind with 20 m/s mean wind speed.

4.7 Conclusions.

This chapter has investigated the benefits and feasibility of derating a downwind turbine based on a feed forward signal from an upwind turbine. A survey of available turbine derating literature was carried out. Three turbine derating strategies from literature were then investigated. It was found that derating a turbine by reducing the rated rotor speed yielded the largest reductions in structural loads and rotor overspeeds. The effect of turbine derating on dynamic turbine response was

then investigated by simulating derated NREL 5-MW turbines subjected to extreme operating gusts (EOGs). It was found that rotor speed, power, blade root bending moments, and tower fore-aft bending moments were all significantly reduced by derating and that the reductions were roughly proportional to the amount of derating. The transition into and out of derated mode was also investigated. It was found that abruptly changing the rating of an NREL 5-MW turbine excited undesirable transient behavior in the turbine. However, the undesirable behavior could be avoided by using a low pass filter. A feed forward selective derating scheme based on these findings was developed. The scheme uses two possible methods to determine when to derate the downwind turbine. The first relies on a wind convection velocity estimate from the upwind turbine. The second, more conservative, method uses the range of wind speeds in which rotor overspeeds are likely to occur.

Once the feed forward selective derating strategy was designed, the performance of that system was tested in a series of FAST simulations. The FAST simulations modeled a two turbine system in several wind conditions, including both IEC Extreme Operating Gusts and turbulent wind with large gusts. The simulation results were promising. The downwind turbine experienced significant reductions in tower fore-aft moments in all simulation cases, as well as reductions in blade root bending moments in some cases. More importantly, the downwind turbine did not experience any rotor overspeeds large enough to trigger an emergency shut down. The downwind turbine experienced reductions in energy generation, however those reductions were much smaller than the reductions in energy generation that would be caused by an emergency overspeed shut down.

Though the simulations carried out in this chapter show promising system performance, more investigation is needed. A real world implementation of this system would encounter several phenomena that were not captured in these simulation. Due to the limitations of the simulation tools, neither full field turbulence nor the evolution of the wind field over time could be simulated. In addition, simulating the system with a different simulation tool could potentially lend validity to the results

shown in this chapter. In the following chapters a more sophisticated, though more computationally intensive, wind farm simulation tool will be investigated and used to study the feed forward selective derating scheme developed in this chapter.

Chapter 5

A Comparison of the NREL 5-MW Wake Characteristics Using Both SOWFA and OVERFLOW2

5.1 Introduction

Utility scale wind power is often installed as wind farms, with several turbines installed in close proximity to each other. Many established simulation tools used in the wind industry simulate single turbines.[18, 33, 49, 50] While single turbine simulation tools are valuable, they fail to capture the turbine-to-turbine interactions that are common in a wind farm. The National Renewable Energy Laboratory's (NREL) Simulator fOr Wind Farm Applications (SOWFA) is a wind farm simulation tool.[51] SOWFA uses a large eddy simulation (LES) methodology to model a flow field, while using a computationally inexpensive but well established aero-elastic simulation to model turbines in the flow field. By coupling a high fidelity flow field model to a medium fidelity turbine model SOWFA strives to accurately model wind farm aerodynamics without incurring the high computational cost of modeling a turbine with CFD techniques. SOWFA has the potential to be a valuable tool for studying wind farm phenomena such as turbine-to-turbine wake interactions[52], plant level controls[53], and complex terrain.[54] However, SOWFA is still a relatively new simulation tool and more validation of SOWFA's capabilities would

be beneficial. This chapter focuses on a comparison of the predicted NREL 5-MW wake structure and behavior using both SOWFA and a Reynolds-averaged Navier-Stokes (RANS) method, OVERFLOW2. The NREL 5-MW rotor is analyzed at several wind speeds and the aerodynamic performance predicted by SOWFA and OVERFLOW2 are compared. This chapter will also examine how SOWFA results are affected by downstream length and grid resolution of the near-wake grid.

The study documented in this chapter was carried out in collaboration with Dr Raymond Chow. Dr Chow performed the RANS simulations discussed below and provided several of the figures, as noted in the figure captions. Dr Chow also featured these RANS simulations in a pair of publications.[49, 50]

5.2 Hybrid LES Method

5.2.1 SOWFA

In SOWFA, the flow field is modeled using a large eddy simulation (LES) methodology based on the open source CFD software openFOAM.[51] The large-eddy simulation is governed by the momentum transport equation, with terms accounting for convection and the Coriolis force, shown in Equation 5.1 and the potential temperature transport equation shown in Equation 5.2. The last term in Equation 5.1 (f_i^T) accounts for other forces applied to the flow field (from the wind turbine actuator line model). Sub filter scale momentum and temperature fluxes are calculated using the Smagorinski model.[55] The filter length scale is defined as $\Delta = (\Delta x \Delta y \Delta z)^{1/3}$ where Δx , Δy , and Δz are the local cell lengths.

$$\frac{\partial \bar{u}_i}{\partial t} + \frac{\partial}{\partial x_i}(\bar{u}_i \bar{u}_j) = -2\epsilon_{i3k}\Omega_3 \bar{u}_k - \frac{\partial \bar{p}}{\partial x_i} - \frac{1}{\rho_0} \frac{\partial}{\partial x_j} \bar{p}_0(x, y) - \frac{\partial}{\partial x_j}(\tau_{ij}^D) - g \left(\frac{\bar{\theta} - \theta_0}{\theta_o} \right) \delta_{i3} + \frac{1}{\rho_0} f_i^T \quad (5.1)$$

$$\frac{\partial \bar{\theta}}{\partial t} + \frac{\partial}{\partial x_j}(\bar{u}_j \bar{\theta}) = - \frac{\partial}{\partial x_j}(q_j) \quad (5.2)$$

5.2.2 Actuator Line Model

The LES model is 2-way-coupled to the aeroelastic turbine model using actuator line models of the turbine blades.[56] Because of the high Reynolds number flow

over the viscous surface of the rotor and the related fine computational grid requirements, it is currently difficult to model the geometry of the rotor accurately directly with an LES method. As such a different method to model the rotor flow is needed if the advantages of LES are to be gained in the global domain. Previous methods have utilized actuator disk models within LES domains, but the resultant flow fields do not capture the vortical wake structure of a rotor flow field. Instead, due to the averaging of the loading across the entire rotor plane, only a smeared out momentum deficit is captured in the wake.[57] SOWFA attempts to improve on this by using an actuator line method. Within the LES domain, the rotating blades are modeled using unsteady, rotating line elements. Along each element, the local LES flow characteristic is determined and transferred to FAST at each coupled time-step. The local blade loading is determined in FAST and the reaction forces are returned to the LES solver and perturbs the local flow field.

This of course greatly simplifies the aerodynamic modeling for the rotor itself, but the approximate rotor loading and corresponding flow field impact should be correctly captured. Furthermore, using the rotating actuator line method should also lead to an unsteady, rotating wake structure that more closely resembles an actual rotor flow. This improved wake structure, with discrete tip and root vortices, could lead to improved predictions in terms of wake coherence, wake-turbine interaction, and atmospheric boundary layer impact.

5.2.3 Aeroelastic Turbine Model

Accurately modeling a wind turbine with an aeroelastic code is far less computationally expensive than modeling a wind turbine with traditional CFD methods such as viscous 3-D RANS. In SOWFA wind turbines are modeled using a version of NREL's FAST that has been coupled with SOWFA. FAST is a medium fidelity aeroelastic wind turbine modeling tool. It is widely used in the wind industry for turbine design. FAST models turbine structural dynamics as a combination of modal dynamics and multibody dynamics. The multibody dynamics are calculated using Kanes' method.[58] FAST models aerodynamic loading on the turbine using blade element-

momentum (BEM) theory. In BEM theory turbine blades are treated as a collection of discrete 2-D airfoil segments that move in space as the turbines blades rotate and flex. The aerodynamic forces on each airfoil segment are determined from the local air flow field, as well as the motion, orientation, and aerodynamic properties of the airfoil. The aerodynamic forces on the rotor are determined by summing the forces on all of the blade segments and applying a series of correction factors to account for phenomenon such as tip and root vortices, and dynamic stall.[9]

5.2.4 LES Computational Domain

The computational domain extends $20R$ upstream, where the NREL 5-MW rotor radius R is equal to 63 m, and $40R$ downstream of the rotor. In the rotor plane the computational domain extends $20R$ from the center of the rotor in both the vertical and horizontal directions. Initially this rectangular domain is composed of 32 m \times 32 m \times 32 m cells, but portions of the domain go through a series of grid refinements to reduce the size of the cells near the NREL 5-MW rotor and wake. The refinement regions are a series of 5 concentric cylinders with centerlines laying on the rotational axis of the NREL 5-MW rotor and having the dimensions shown in Table 5.1. Table 5.1 describes the refinement regions in both meters and in terms of the NREL 5-MW rotor radius R . This grid refinement scheme is modified for some of the simulations in Sections 5.5 and 5.5. The modifications are described in those sections. SOWFA has the capability to model ground effects, as well as the effects of neutral and unstable atmospheric conditions. However, in order to match the OVERFLOW2 simulations, those capabilities are not used and this study is limited to a rotor in a uniform and unbounded freestream.

Refinement Region	Upstream (R)	Downstream (R)	Radial (R)	Max Cell Size in Region
1	$8.5R$ (536 m)	$40R$ (2320 m)	$8.9R$ (561 m)	$16\text{ m} \times 16\text{ m} \times 16\text{ m}$
2	$4.5R$ (284 m)	$40R$ (2320 m)	$4.9R$ (309 m)	$8\text{ m} \times 8\text{ m} \times 8\text{ m}$
3	$2.5R$ (158 m)	$24R$ (1512 m)	$2.9R$ (183 m)	$4\text{ m} \times 4\text{ m} \times 4\text{ m}$
4	$1.5R$ (95 m)	$12R$ (756 m)	$1.9R$ (120 m)	$2\text{ m} \times 2\text{ m} \times 2\text{ m}$
5	$1.0R$ (63 m)	$6R$ (378 m)	$1.4R$ (88 m)	$1\text{ m} \times 1\text{ m} \times 1\text{ m}$

Table 5.1. Extents and resolutions of the SOWFA grid refinement regions (distances extending from the center of the rotor).

5.3 RANS Computational Method

5.3.1 OVERFLOW2

OVERFLOW2 is a numerical simulation method that solves the compressible Reynolds-averaged Navier-Stokes equations on structured, overset grids.[59, 60] OVERFLOW2 is a robust and comprehensive code, allowing for the selection from a variety of numerical schemes, turbulence models, boundary conditions, and time advancement schemes.[61, 62] The numerical method selected for this portion of the study will be central difference Euler with a Beam-Warming pentadiagonal scheme.[63] The viscous near-body grids will utilize a spatially 2nd-order method, while the Cartesian off-body grids will use a spatially 4th-order accurate method with 6th-order matrix dissipation term. All solid boundaries are treated as viscous walls, and all calculations are performed fully turbulent, with Menter's two-equation Shear Stress Transport (SST) $k-\omega$ model [64] being used exclusively. Only the Euler terms are solved in the off-body grid system in order to reduce excess numerical dissipation. All unsteady calculations are performed second-order in time using dual-time stepping.[65] To model the rotation of the rotor, the rotational source term formulation of the Navier-Stokes equations will be used, where the Coriolis and centrifugal force terms are added to the governing equations.

5.3.2 RANS Computational Meshes

The Chimera overset structured grid scheme is employed to model the computational flow domain. This approach allows geometrically complex multi-body con-

figurations to be constructed from sets of relatively simple overlapping body-fitted grids.[66, 67] The near-body computational grid is generated with Chimera Grid Tools (CGT) 2.0.[68] The CGT package contains independent grid generation, manipulation, visualization and diagnosis tools that can be run in batch mode under the OVERGRID graphical interface.[69] Overlapping grid regions in the hub region and with the off-body grids are cut or IBLANKed using Meakin's object X-Ray method.[70] Domain connectivity is also performed automatically by OVERFLOW2 using a non-conservative interpolation between overlapping grids. All body-fitted grids are generated with a wall spacing of approximately $y^+ \cong 0.3$.[71, 72] Off-body grids were constructed using a series of cubic Cartesian layers or BRICKS [73] automatically generated by OVERFLOW2 based on user specified cell size, growth rate, and far-field distance. Layers of off-body BRICKS are built from the near-body, with each additional layer doubling in cell dimension, and having the benefit of a point-matched donor-injection pair at the overset boundaries. This allows for a very efficient and cost effective way of filling the off-body domain. In this work, the far-field distance will be defined as the distance from the rotor center to one of the five freestream imposed boundaries on the upstream and four Cartesian sides of the domain. In the downstream direction, the inflow/outflow boundary is placed twice this far-field distance away from the rotor.

5.3.3 RANS NREL 5-MW Rotor Near-body Grid

A fully viscous Overset grid system was developed for the NREL 5-MW rotor. The full 3-bladed rotor was modeled without the use of any symmetry or periodic conditions in the global domain. Each of the three $R = 63$ m blades is defined using an identical set of hub, inboard, outboard and tip near-body grids. The surface grids are hyperbolically grown out approximately 10 m ($2.13c_{max}$) away from the body with an O-type topology. The near-body grid system utilized a total of 13M grid points. The off-body grid system extends $20R$ upstream and around the rotor, and $40R$ in the downstream direction. The rotor itself is surrounded by an initial uniform Cartesian near-wake grid that extends $6R$ downstream and $0.5R$ upstream of

the rotor with a constant cell size of 1.0 m. Extensive details of the near-body and off-body grid independence studies can be found in previous studies.[66–68] Calculations were continued until both integrated power and thrust values deviated by less than 0.1% and residuals levels remained effectively constant over a period of 30-40 thousand iterations. All steady-state solutions were run between 120-160 thousand iterations. Time-accurate calculations were performed so that at least a two-order of magnitude drop in the L2-residual was achieved for all grids to ensure temporal convergence. This utilized a physical time-step of 1.4 msec (or rotor azimuthal angle increment of 0.1 degrees per step) and required 25 subiterations per global time-step for convergence. The composite overset grid topology for one of the blades from the baseline NREL 5-MW surface grid is shown in Figure 5.1 along with the thickness-to-chord ratio distribution over the span of the blade.

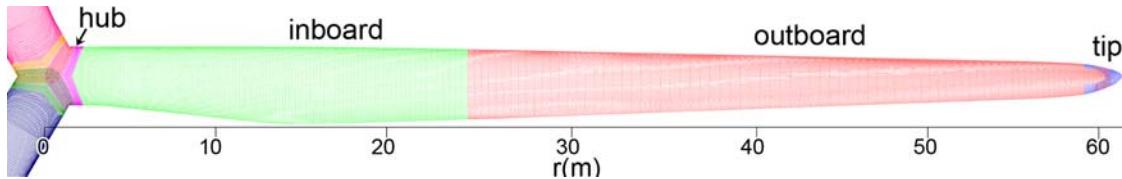


Figure 5.1. Near-body overset topology of the NREL 5-MW medium resolution grid system. (Source: R. Chow)

5.4 Comparison of SOWFA and OVERFLOW2 Results

SOWFA and OVERFLOW2 results are compared for the cases shown in Table 5.2. The incoming wind speeds (U_∞) were chosen to cover three control regions of the NREL 5-MW turbine. In 8 m/s winds the turbine operates in control region 2, variable speed control. In 15 m/s winds the turbine operates in control region 3, variable pitch control. In 11 m/s winds the turbine operates in control region 2.5, the transition between region 2 and region 3. In all cases the incoming wind field is laminar, steady and uniform. The rotor speed and pitch shown in Table 5.2 are the steady state operating conditions of the NREL 5-MW turbine at the corresponding incoming wind speed. Because there are no simulation results for the NREL 5-MW

U_∞ (m/s)	Rotor Speed (RPM)	Pitch (°)
8	9.16	0
11	11.89	0
15	12.10	10.45

Table 5.2. NREL 5-MW simulation cases

turbine a direct validation was not possible. However, the OVERFLOW2 simulation results were compared to an independent simulation of the NREL 5-MW rotor carried out by Sørensen and Johansen using EllipSys3D.[56] The thrust and power predicted with OVERFLOW2 were found to have good agreement with previous published predictions for the NREL 5-MW.[49]

The SOWFA near-wake grid extends $6R$ (378 m) downstream and $1R$ (63 m) upstream, the same dimensions used for OVERFLOW2 simulations. In the radial direction the SOWFA near-wake grid extends $1.4R$ (88 m). This makes the in-plane area of the SOWFA near-wake grid approximately 25% larger than the 140×140 m OVERFLOW2 grid and provides a constant 25 m spacing between the blade tips and the edge of the near-wake grid. In the rectangular OVERFLOW2 grid the spacing between the blade tips and the edge of the near wake grid varies from 7-36 m depending on the instantaneous orientation of the rotor. Other SOWFA parameters for these cases are chosen based on best practices.[51, 74] The Gaussian projection width was set to the mean chord length of the NREL 5-MW blade (3.5 m). Each actuator line has 123 elements, ensuring that each actuator element is less than 3/4 the near-wake grid resolution. BEM tip and root losses are disabled because the LES methodology captures the vortical wake. The simulation time step is chosen to be 0.01s so turbine blade tips can not pass through more than one cell per time step.

Table 5.3 shows the rotor power and thrust predicted by SOWFA, OVERFLOW2, and standalone FAST simulations. In standalone FAST simulations the uniform incoming wind field is specified by a simple wind input file instead of an LES field. Tip and root loss corrections are enabled for standalone FAST simulations. For the

$U_\infty = 8$ m/s and $U_\infty = 11$ m/s cases SOWFA yields the highest power predictions while OVERFLOW2 yields the lowest. For the $U_\infty = 15$ m/s case SOWFA yields the lowest power predictions while OVERFLOW2 yields the highest. In all three cases the power predicted by FAST is approximately half way between the SOWFA and OVERFLOW2 results. The power predicted by SOWFA is known to increase as the Gaussian projection width of the actuator line model is increased. It is therefore possible to tune the power prediction of SOWFA by changing the Gaussian projection width. However, since SOWFA is predicting lower power for two cases and higher power for one case, our initial choice of a 3.5 m Gaussian width appears appropriate. The thrust predicted by SOWFA is lower than the thrust predicted by OVERFLOW2 or FAST, but there is good agreement between the SOWFA and OVERFLOW2 predictions.

U_∞ (m/s)	SOWFA		OVERFLOW2		FAST	
	Power (kW)	Thrust (kN)	Power (kW)	Thrust (kN)	Power (kW)	Thrust (kN)
8	1985	382	1733	399	1875	477
11	5061	693	4650	733	4827	789
15	5093	405	5499	455	5297	520

Table 5.3. Comparison of power and thrust predicted by three turbine simulation tools.

Figure 5.2 shows iso-vorticity plots of the OVERFLOW2 and SOWFA results for all three wind conditions. The OVERFLOW2 and SOWFA results show similar wake behavior, especially for the first $2R$ downstream of the rotor. For the $U_\infty = 8$ m/s and $U_\infty = 11$ m/s cases the rotor generates strong blade tip vortices that merge approximately $1R$ downstream. Both OVERFLOW2 and SOWFA show the merged blade tip vortices breaking down into larger vortical structures and dissipating, however this process happens further upstream in the OVERFLOW2 results. OVERFLOW2 shows the merged blade tip vortices breaking apart approximately $2R$ downstream and dissipating by the time they reach $10-12R$ downstream. The SOWFA results show the merged blade tip vortices breaking apart approxi-

mately $6-8R$ downstream and dissipating more than $12R$ downstream. For the $U_\infty = 15$ m/s case both OVERFLOW2 and SOWFA show strong blade tip vortices that do not merge and persist far downstream of the rotor.

Figures 5.3 and 5.4 show instantaneous vorticity magnitude in cross sections of the wake. These figures allow more details of the interior wake structure to be seen. All three SOWFA simulations show strong distinct blade root vortices that persist far downstream of the rotor. This behavior is not shown in the OVERFLOW2 results for the $U_\infty = 8$ m/s and $U_\infty = 11$ m/s cases. In those cases the root vortices dissipate approximately 1 rotor diameter downstream, as can be seen in Figure 5.3. For the $U_\infty = 15$ m/s case OVERFLOW2 does show persistent root vortices. A significant loss of detailed wake behavior can be seen in Figure 5.4a through 5.4c when the wake passes out of the near-wake 1 m grid region into the 2 m grid region.

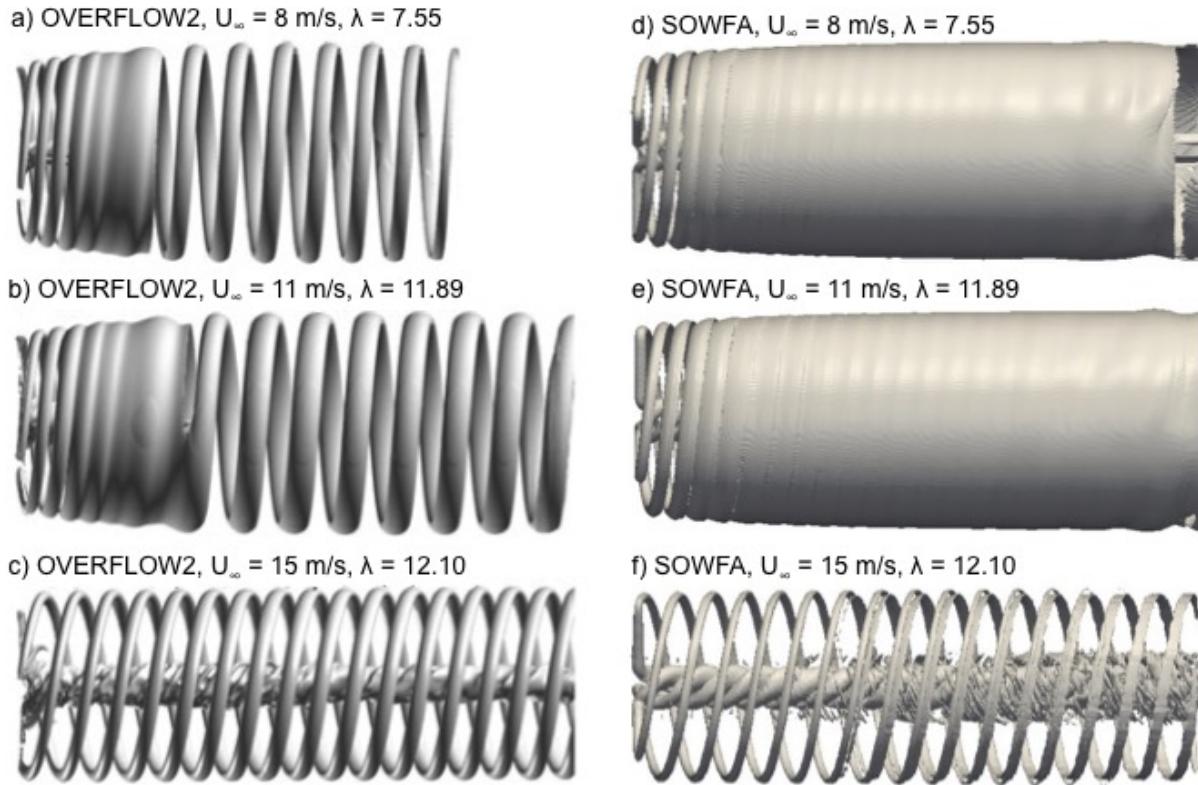


Figure 5.2. Iso-vorticity surfaces of the NREL 5-MW wake system for OVERFLOW2 and SOWFA simulations (Freestream flows from left to right. Images a,b, and c provided by R. Chow).

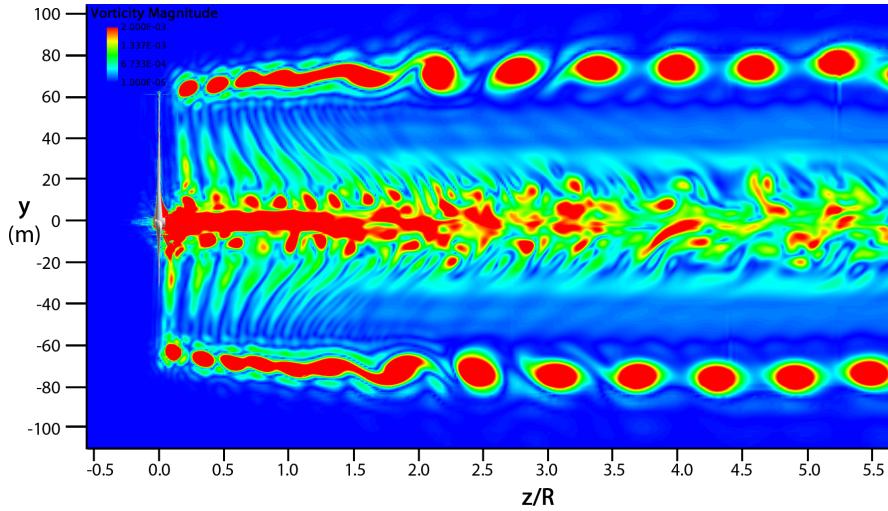


Figure 5.3. Instantaneous vorticity magnitude at a center cut plane of the NREL 5-MW rotor at $U_{\infty} = 11$ m/s using OVERFLOW2 (Freestream flows from left to right. Source: R. Chow).

Figure 5.5 shows the momentum deficit in the NREL 5-MW wake predicted by OVERFLOW2 and SOWFA for a range of stations downstream of the rotor. The downstream stations are at 10 m ($0.16R$), $0.5R$, $1R$, $2R$, $4R$, $6R$, $8R$, and $12R$. A clear difference between the SOWFA and OVERFLOW2 results is the momentum deficit at the center of the rotor. For the first few downstream stations OVERFLOW2 predicts large momentum deficits in the blade root region whereas SOWFA predicts no momentum deficit. This discrepancy may be a contributing factor to the lower thrust values predicted by SOWFA, as seen in Table 5.3, and is likely caused by the difference in how SOWFA and OVERFLOW2 treat the rotor hub. OVERFLOW2 models the entire rotor, including the blades and a simplified version of the hub. SOWFA only models the blades, in effect modeling the rotor as if it had no hub. The $U_{\infty}=8$, and 15 m/s cases also show significant discrepancies at the furthest downstream stations. At $8R$ and $12R$ downstream OVERFLOW2 shows the momentum deficit widening and decreasing in magnitude, especially near the edges of the wake. SOWFA results show the momentum deficit continuing to maintain its shape and magnitude. When the OVERFLOW2 results were originally generated the focus was on capturing power and thrust values sufficiently well.[49] Capturing

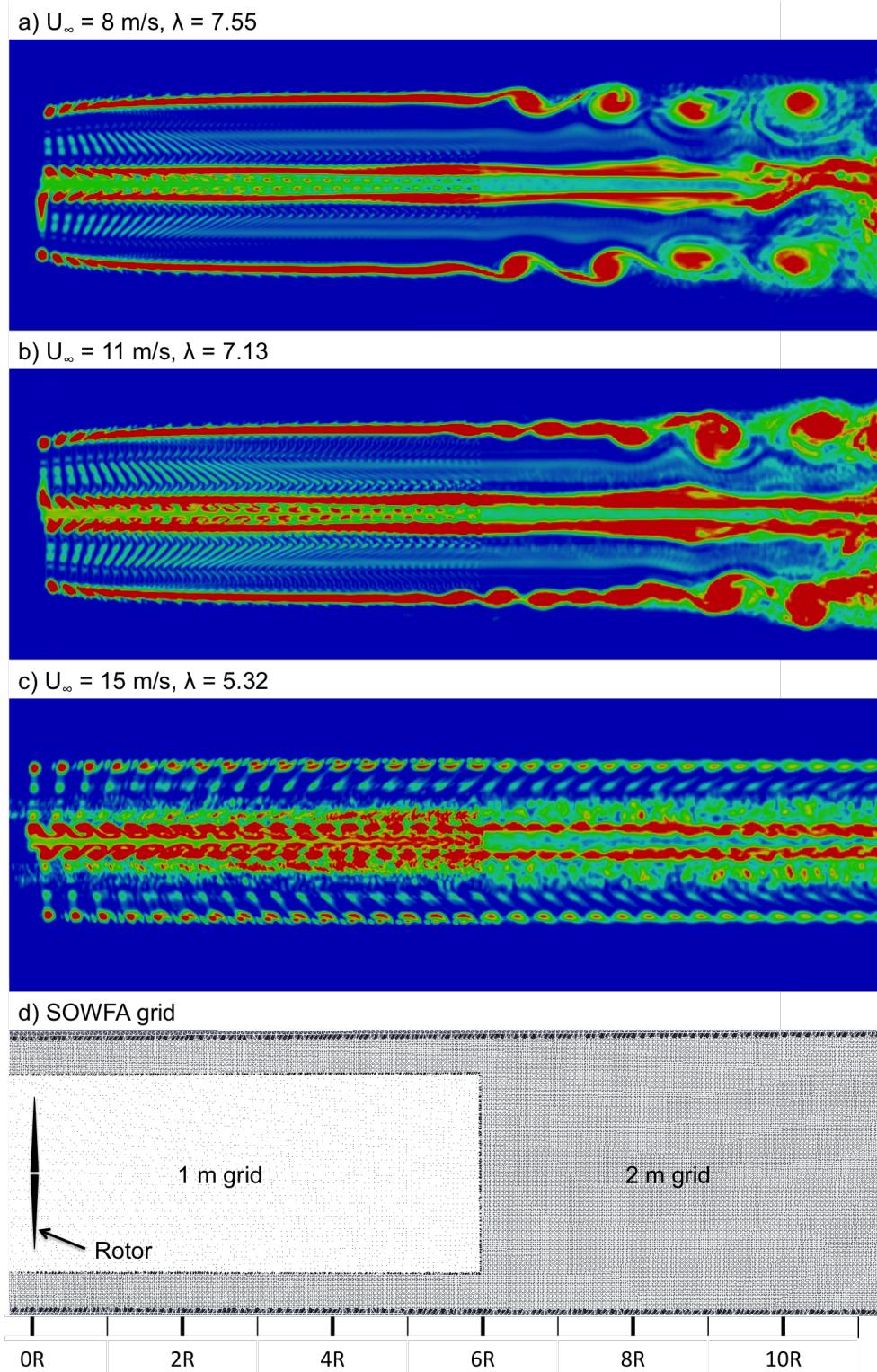


Figure 5.4. Instantaneous vorticity magnitude at a center cut plane of the NREL 5-MW rotor for SOWFA simulations at $U_\infty=8, 11$, and 15 m/s , as well as the SOWFA near-wake grid used in the simulations (Freestream flows from left to right).

the wake structure accurately was a necessary consequence and not the primary focus of the work. Because the far downstream wake behavior was of less interest in the original work and because of the high computational cost of these simulations, the far-wake grid used is coarser (8 - 16 m) than the SOWFA grid (2 m) at the $8R$ and $12R$ downstream stations. These downstream discrepancies occur when the coarser far-wake grids used in the OVEFLOW2 simulations can no longer support the fine structure details of the flow. Nevertheless there is good agreement between SOWFA and OVERFLOW2 with OVERFLOW2 generally showing slightly higher momentum deficits at the edges of the wake at most downstream stations.

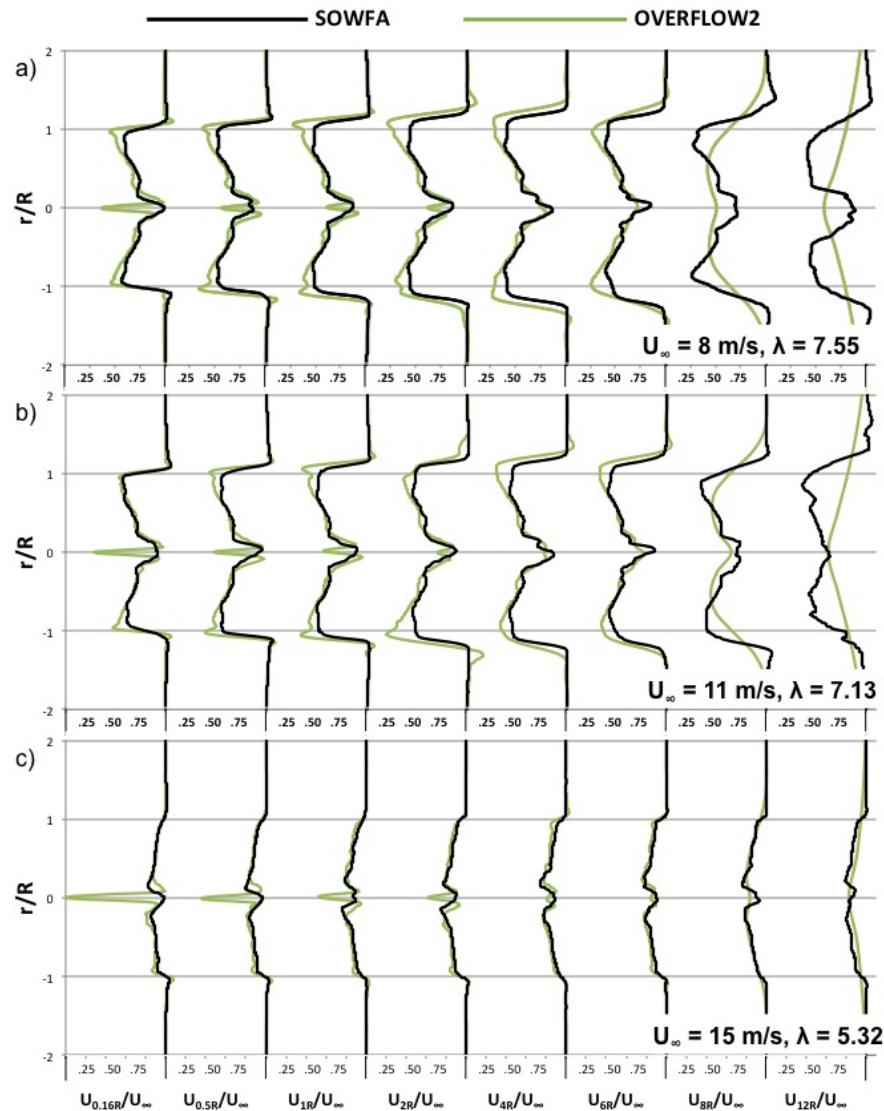


Figure 5.5. Axial momentum deficit predicted by SOWFA and OVERFLOW2 for the NREL 5-MW rotor at $U_\infty = 8, 11$, and 15 m/s .

5.5 Effects of Near-Wake Grid Length

The size of the high resolution near-wake grid has a large effect on the number of grid cells in the simulation and therefore the amount of computing effort required to run a SOWFA simulation. In this section a study is carried out to examine how the downstream length of the near-wake affects rotor wake behavior. The purpose of this study is twofold, to gain insight into the grid length necessary to accurately capture wake behavior and to confirm that the SOWFA near-wake grid used for simulations in the previous section extends sufficiently far downstream. To examine the effect of near-wake grid length on wake behavior the $U_\infty = 11$ m/s SOWFA simulation case is modified by varying the downstream length of the near-wake grid. Simulations are carried out with near-wake grids extending $1R$, $2R$, and $12R$ downstream. The results from these simulations are compared to each other as well as to results from the baseline $U_\infty = 11$ m/s simulation, which has a near-wake grid extending $6R$ downstream. All other SOWFA simulation parameters remain unchanged except for a small change to the dimensions of the 2 m grid (Refinement Region #4) for the $12R$ simulation. In the $12R$ simulation the downstream length of Refinement Region #4 is extended from $12R$ to $16R$ to allow a smooth gradual transition between the 1 m near-wake grid and the 4 m grid.

The simulations produce identical thrust predictions and nearly identical power predictions, with the $1R$ and $2R$ cases producing slightly more power (5063 kW) than the $6R$ and $12R$ cases (5061 kW). Figure 5.6 shows instantaneous vorticity magnitudes in cross sections of the wake. Though the wake behavior is similar for all four simulations, shorter near-rotor grids result in more loss of the detailed wake structure. In the $1R$ and $2R$ cases some details of the wake tip vortex merging behavior is lost. The $1R$ and $2R$ cases also show the merged vortices breaking up into large vortical structures slightly sooner than the $6R$ and $12R$ cases. The wakes in the $6R$ and $12R$ cases show almost identical behavior until the vortical structures of the wakes break up and becomes chaotic approximately $10R$ downstream.

Figure 5.7 shows the momentum deficit in the NREL 5-MW wake at a variety

of stations downstream. Note that Figure 5.7 includes two additional downstream stations, $16R$ and $20R$, that are not present in Figure 5.5. SOWFA is often used to study turbine to turbine wake interactions, where wake behavior more than $12R$ downstream may be important. Therefore, the $16R$ and $20R$ stations were added to Figure 5.7 to capture wake behavior far downstream. Up to $4R$ downstream the momentum deficits are essentially identical for all simulations. At the $6R$ and $8R$ downstream stations the momentum deficit plots for the $1R$ and $2R$ simulation cases diverge slightly from the momentum deficit plots for the $6R$ and $12R$ simulation cases. The momentum deficits of the $1R$ and $2R$ simulation cases become slightly wider and show a slightly smaller maximum magnitude as the wake begins to break down. By the time the wakes reach $12R$ downstream all four simulations are showing signs of the wake breaking down. The chaotic nature of the wakes at the $12R$, $16R$, and $20R$ downstream stations causes the instantaneous momentum deficit plots to diverge for all four simulations.

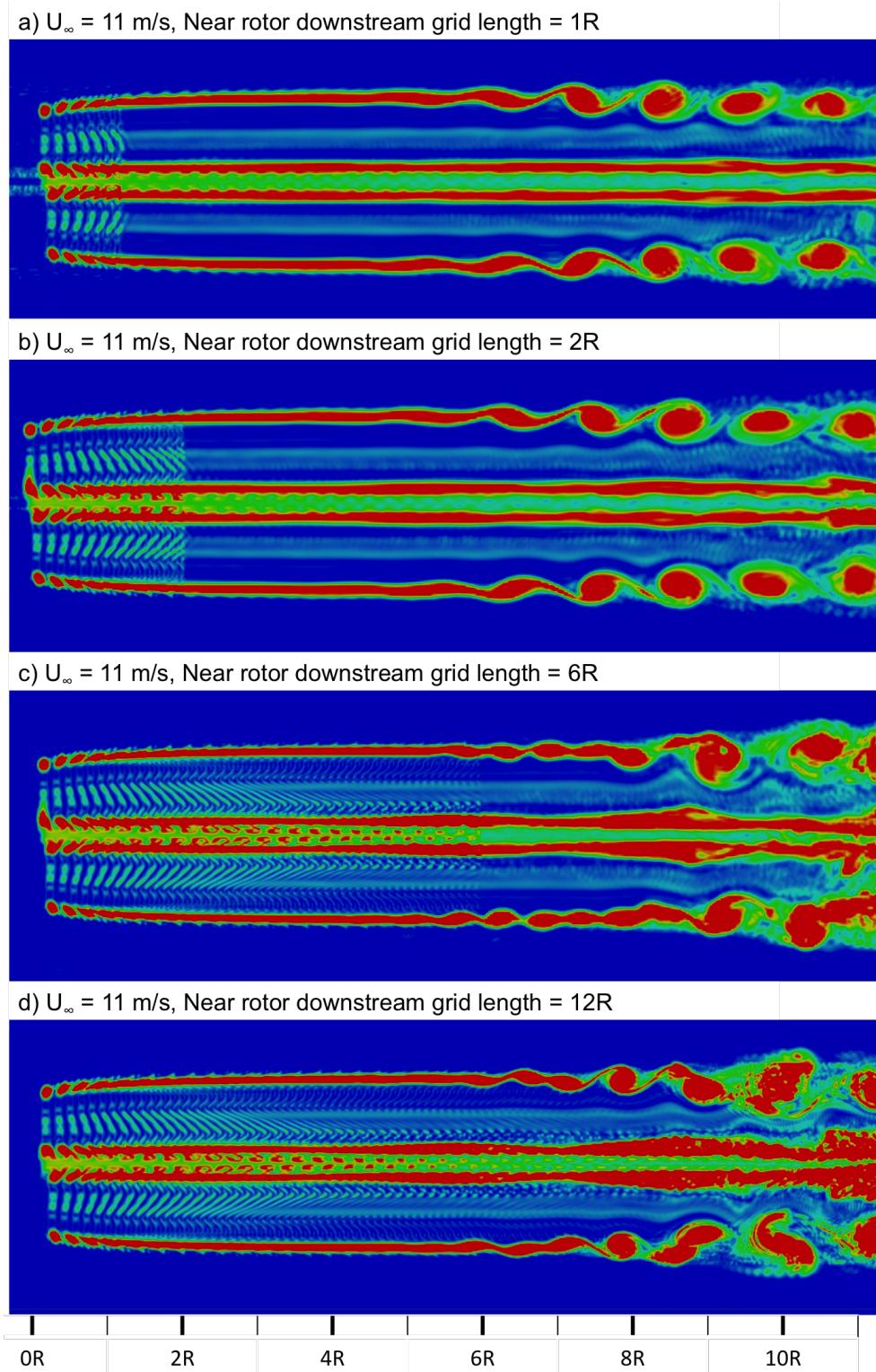


Figure 5.6. Instantaneous vorticity magnitude at a center cut plane of the NREL 5-MW rotor for SOWFA simulations at $U_\infty = 11 \text{ m/s}$ using near-wake grids of various lengths (Freestream flows from left to right).

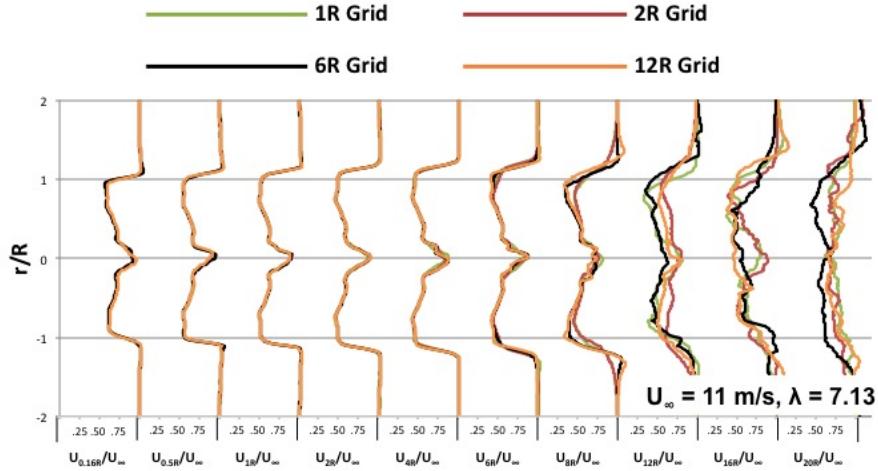


Figure 5.7. Axial momentum deficit predicted by SOWFA for NREL 5-MW rotor at $U_\infty = 11$ m/s using near-wake grids of various lengths.

5.6 Effects of Near-Wake Grid Resolution

In previous sections we use a 1 m near-wake grid for SOWFA simulations. However, in practice SOWFA is often used with a coarser near-wake grid (2.5–5 m).[52, 75–77] To examine the effect of near-wake grid resolution, SOWFA simulations of the $U_\infty = 8, 11$, and 15 m/s cases are run with a coarse near-wake grid and the results of those simulations are compared to the SOWFA simulation results in section 5.4. The coarse near-wake grid is generated by omitting two of the grid refinement steps described in section 5.2.4. By not refining the grid in Refinement Regions #4 and #5, we get a near-wake grid that has the dimensions of Refinement Region #3 ($2.5R$ upstream, $24.0R$ downstream, $2.9R$ radial) and is filled with $4\text{ m} \times 4\text{ m} \times 4\text{ m}$ cells. Previous studies have shown that the Gaussian projection width must be at least double the near-wake grid resolution for a stable solution.[51, 74] Therefore, the Gaussian projection width has been increased from the mean chord length of the NREL-5MW blade (3.5 m) to 8 m. All other SOWFA simulation parameters remain unchanged.

Table 5.4 shows the power and thrust predicted by SOWFA when a 4 m near-wake grid is utilized as well as the percent difference between these results and the SOWFA results from section 5.4. The coarse grid causes an increase in the power

U_∞ (m/s)	Power (kW)	Thrust (kN)	Δ Power (%)	Δ Thrust (%)
8	2076	390	4.6	2.1
11	5274	707	4.2	2.0
15	5208	412	2.2	1.7

Table 5.4. Power and thrust predicted by coarse grid SOWFA simulations as well as the percent difference with fine grid simulations.

and thrust predictions for all three cases. Figure 5.8 shows instantaneous vorticity magnitudes in cross sections of the wake. By comparing Figure 5.8 to Figure 5.4 we can see how the use of a coarse 4 m near-wake grid affects wake behavior. Note that the images in Figure 5.8 extend $20R$ downstream of the rotor, while the images in Figure 5.4 extend only $12R$ downstream. The wakes in Figure 5.8 show far less detail than those shown in Figure 5.4. Of particular interest, the strong tip vortices that can be seen forming and merging in Figures 5.4a and 5.4b are not present in Figures 5.8a and 5.8b. Figures 5.8a and 5.8b also show the wake extending much further downstream before breaking up, when compared to Figures 5.4a and 5.4b. The differences in wake behavior are less pronounced when we compare Figure 5.8c to Figure 5.4c. Though Figure 5.8c shows far less detail, the wake behavior does not seem to be fundamentally changed. Figure 5.8c essentially looks like a lower resolution version of Figure 5.4c.

Figure 5.9 shows the momentum deficit in the NREL 5-MW wake at a variety of stations downstream. The downstream stations shown are: 10 m ($0.16R$), $0.5R$, $1R$, $2R$, $4R$, $6R$, $8R$, $12R$, $16R$, and $20R$. At many of the downstream stations, the magnitude and shape of the momentum deficits predicted with the 4 m near-wake grid is similar to the magnitude and shape of the momentum deficit predicted with the 1 m near-wake grid. However, the momentum deficits in the coarse grid simulations show more gradual transitions near the center and edges of the wake, while the momentum deficits in the 1 m grid show sharper transitions. This makes sense based on the lower level of wake detail we see in Figure 5.8. In Figures 5.9a and 5.9b we see a larger discrepancy between the 1 m grid results and the 4 m grid

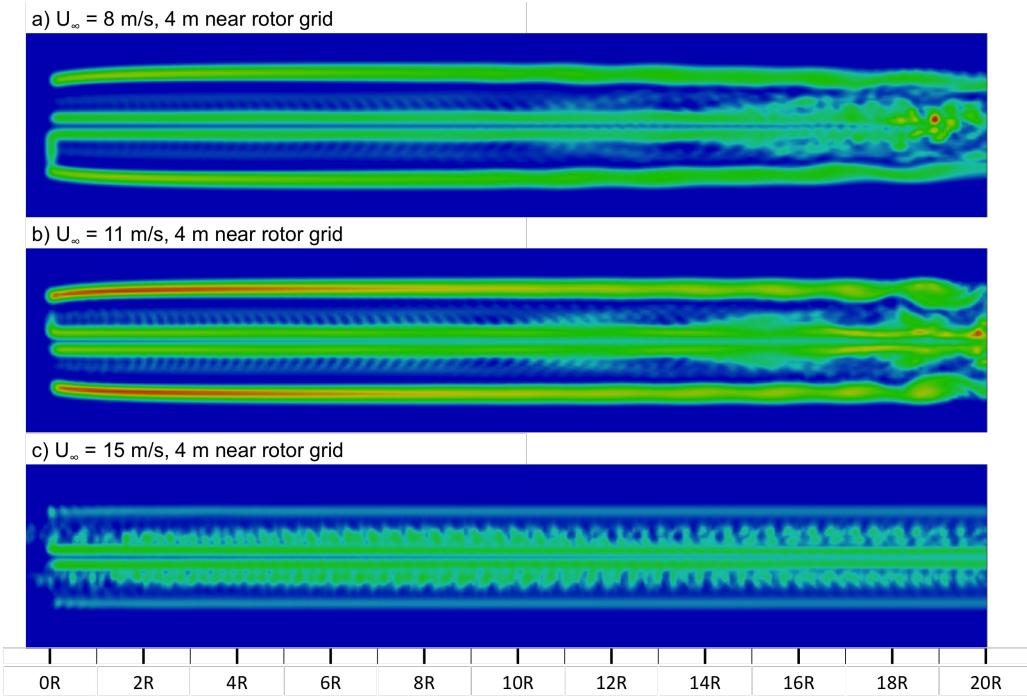


Figure 5.8. Instantaneous vorticity magnitude at a center cut plane of the NREL 5-MW rotor for SOWFA simulations at $U_{\infty} = 8, 11$, and 15 m/s , using a 4 m near-wake grid (Freestream flows from left to right).

results at the $16R$ and $20R$ downstream stations. At those stations the 1 m grid results show the shape of the instantaneous momentum deficit becoming chaotic as the wake breaks down. The 4 m grid results show the momentum deficit retaining a smooth shape, while the width of the momentum deficit increases and the magnitude of the momentum deficit decreases. For the free stream velocity $U_{\infty} = 8 \text{ m/s}$ (Figure 5.9a) the 4 m grid results also show a larger momentum deficit magnitude than the 1 m grid results at the $16R$ and $20R$ downstream stations.

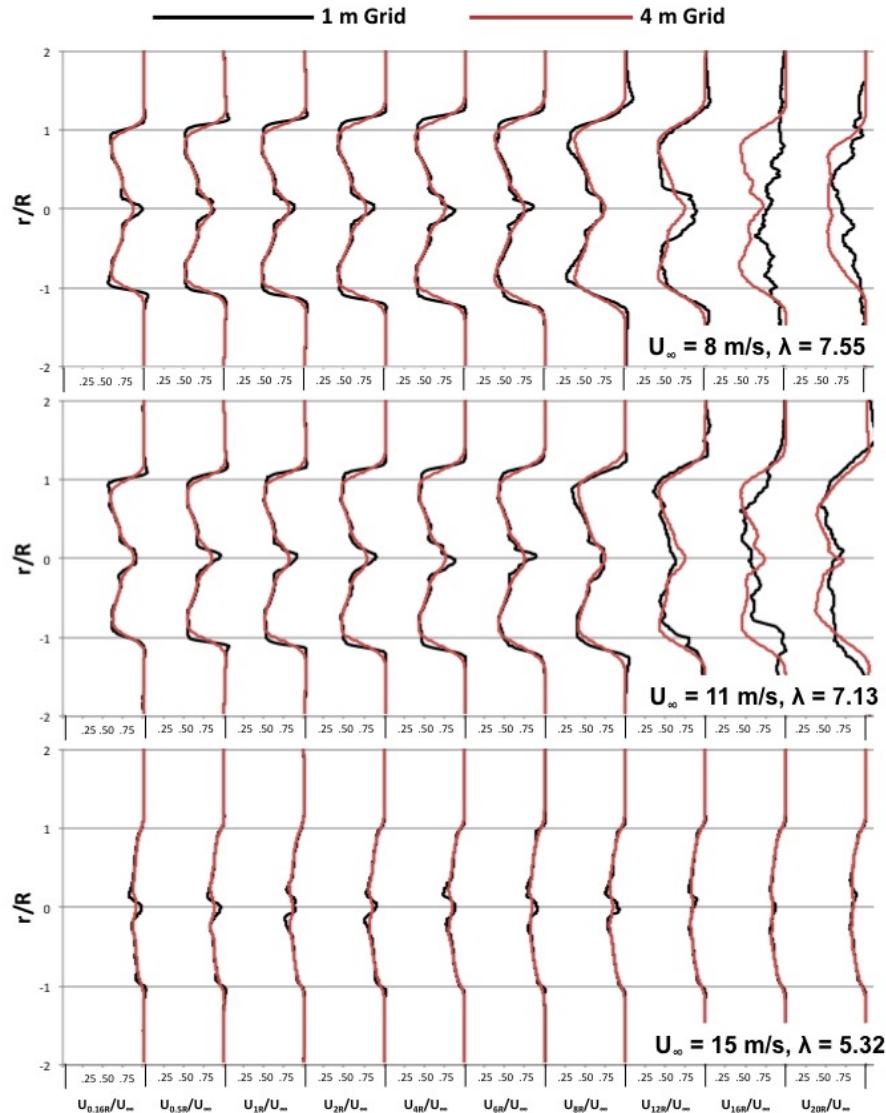


Figure 5.9. Axial momentum deficit predicted by SOWFA for NREL 5-MW rotor at $U_{\infty} = 8, 11$, and 15 m/s , using both fine (1 m) and coarse (4 m) near-wake grids.

5.7 Conclusions

The aerodynamic behavior predicted by OVERFLOW2 and SOWFA have been analyzed and compared for three laminar, steady, and uniform wind speed cases spread over the operating range of the NREL 5-MW turbine. Overall the results from OVERFLOW2 and SOWFA are found to have good agreement. Power and thrust predictions are reasonably close, with SOWFA predicting higher power in $U_\infty = 8$ m/s and $U_\infty = 11$ m/s winds while predicting lower power in $U_\infty = 15$ m/s winds. SOWFA predicts slightly lower thrust for all three cases. In all cases the vortical structure of the wakes predicted by OVERFLOW2 and SOWFA were found to be very similar for the first $2R$ downstream of the rotor. For the $U_\infty = 8$ m/s and $U_\infty = 11$ m/s cases some differences between OVERFLOW2 and SOWFA were observed further downstream. The downstream momentum deficits predicted by the two simulation tools were found to have good agreement with a few exceptions. A discrepancy in momentum deficit near the center of the rotor can be explained by the difference in how OVERFLOW2 and SOWFA model the rotor hub. A discrepancy in the far downstream momentum deficits for the $U_\infty = 8$ m/s and $U_\infty = 11$ m/s cases is consistent with the discrepancy seen in the far downstream wake behavior.

The effect of the downstream length of the near-wake grid has been examined. SOWFA simulations of an NREL 5-MW rotor in $U_\infty = 11$ m/s winds were conducted using near-wake grids of various lengths. The results show that extending the downstream length of the near-wake grid from $6R$ to $12R$ does not have an effect on wake behavior. This suggests that the $6R$ long grid used for SOWFA simulations in section 5.4 is sufficiently long. Shortening the downstream length of the near-wake grid to $1R$ or $2R$ did have an effect on the wake behavior, but the effect was slight. In future simulations the small change in wake behavior caused by using a short near-wake grid may be a reasonable price to pay in order to reduce the grid size and computational time and cost of SOWFA simulations.

The effect of near-wake grid resolution was also examined. The $U_\infty = 8$ m/s, $U_\infty = 11$ m/s, and $U_\infty = 15$ m/s cases were simulated using near-wake SOWFA grids

with 4 m resolution. The SOWFA results generated using a 4 m near-wake grid were then compared to the 1 m near-wake grid wake for SOWFA simulations in Section 5.4. Increasing the near-wake grid cell size to 4 m increased both the power and thrust predicted by SOWFA. The increased near-wake cell size also resulted in a significant loss of detailed wake behavior and some differences in the wake behavior far downstream. The differences were most prominent for the $U_{\infty} = 8$ m/s and $U_{\infty} = 11$ m/s cases, which lost the near-rotor tip vortex behavior seen in the 1 m grid results and showed the wake extending significantly farther downstream before breaking up. The momentum deficit predictions show good agreement, except for far downstream stations in the $U_{\infty} = 8$ m/s and $U_{\infty} = 11$ m/s cases.

Overall, the results documented in this chapter provide confidence in the accuracy of SOWFA simulations and provide guidance for future SOWFA simulations. The results in Section 5.4 demonstrate that SOWFA can be tuned to have good agreement with more established wind turbine simulation tools like FAST and OVERFLOW2. Sections 5.5 and 5.6 provide insight into the effect of near rotor grid length and near rotor grid resolution. These insights will be valuable when designing future SOWFA simulations. In the following chapter, SOWFA will be used to simulate and evaluate the feed forward derating control scheme developed in Chapter 4.

Chapter 6

Analysis of Feed Forward Derating Control Scheme With SOWFA

6.1 Introduction

Chapter 4 investigated the benefits and feasibility of derating a downwind turbine based on a feed forward signal from an upwind turbine. The feed forward derating control scheme developed in Chapter 4 monitors the rotor speed of the upwind turbine for large rotor overspeeds, which are indicative of large wind gusts. When those large rotor overspeeds are detected, the downwind turbine is smoothly transitioned to derated operation until the gust passes. Derating a turbine reduces power generation, but also decreases both structural loads and rotor speed, making that turbine less sensitive to the detrimental effects of a large wind gust. By derating the downwind turbine only when the upwind turbine detects a large wind gust, the downwind turbine gains the benefits of derating (reduced loads and overspeeds) when they are needed most while keeping the cost of derating (reduced energy capture) in check.

The control scheme was evaluated using a series of FAST simulations, which showed promising results (Section 4.6). The control scheme reduced peak structural loads and damage equivalent loads (DEL) while decreasing rotor overspeeds enough to avoid emergency shutdowns of the downwind turbine. The control scheme did reduce electricity capture, but the reduction was small and would likely be much

less than the energy lost in an emergency turbine shutdown. Though these results are promising, the simulation methodology used to generate them has several noteworthy limitations. First, these simulations did not model emergency turbine shutdowns due to rotor overspeeds. Second, to simulate this system in FAST we had to assume Taylor's frozen turbulence hypothesis, which provides a simplistic model of wind speed fluctuations passing through the wind farm. As a result, the simulations did not capture the evolution of the gust as it passes from the upwind turbine to the downwind turbine, it did not capture turbine wake effects, and it did not accurately capture the time it takes for the gust to travel from the upwind turbine to the downwind turbine. In this chapter we will evaluate the control scheme developed in Chapter 4 using a simulation tool that does not have these limitations.

As discussed in Chapter 5, the Simulator fOr Wind Farm Applications (SOWFA) is wind farm simulation tool. SOWFA uses FAST to model the dynamics of one or more turbines, a Large Eddy Simulation (LES) to model atmospheric airflow, and actuator line models to enable interaction of the LES and FAST models. Because SOWFA models atmospheric airflow, we can use it to design a simulation that will capture the evolution of a gust over time, wake effects, and the time it takes a gust to reach the downwind turbine. We can also add control logic that will capture the effects of emergency turbine shutdowns due to rotor overspeed. In Chapter 5 SOWFA simulations of the NREL 5-MW rotor were compared to Reynolds Averaged Navier Stokes (RANS) simulations of the same rotor and were found to yield similar results. In addition, several SOWFA simulation parameters were varied to investigate the trade offs between simulation accuracy and computational cost. Because of the work documented in Chapter 5 we have confidence in the accuracy of SOWFA simulations and a good understanding of how to achieve accurate results at an acceptable computational cost.

Sections ?? through ?? describe much of the background work that had to be done before SOWFA simulations of the feed forward selective derating scheme could be carried out. They discuss topics such as implementation of the turbine controller,

tuning and validation of the SOWFA turbine model, modeling gusts in SOWFA, as well as choosing an appropriate LES grid resolution and computational domain. Section ?? presents the first simulation case, in which the turbines are offset slightly so that the downwind turbine isn't in the wake of the upwind turbine. Section ?? presents the second simulation case, in which the downwind turbine is directly behind the upwind turbine and in its wake. Section ?? summarizes this chapter and its findings.

6.2 Controller Implementation

The simulations carried out in Chapter 4 use Simulink and Matlab to model control systems. Individual turbine control, such as determining the appropriate blade pitch and generator torque, is modeled in Simulink. The Simulink model used for Chapter 4 simulations can be found in Appendix C. Plant level control, such as monitoring the upwind turbine and determining when to derate the downwind turbine, is modeled in Matlab scripts. This method has several benefits. First, Simulink and Matlab are user friendly programming languages. They include a large number of pre-programmed functions and subsystems that make controller implementation easier. Second, these Simulink and Matlab controllers are not part of the FAST executable file. Therefore, changing the controller does not require recompiling FAST. This can save a lot of time and effort, especially when a new control system is being developed, tested, and tuned.

Unfortunately, the same controller implementation can not be used for SOWFA simulations. Though SOWFA does use FAST to model turbine dynamics, it uses a modified version of FAST that is compiled for Linux operating systems and does not have the ability to interface with Simulink. To overcome this limitation the Simulink controller developed in Chapter 4 is re-written as a set of FORTRAN subroutines and inserted into the SOWFA/FAST source code. FAST and SOWFA are then recompiled to implement the turbine control system developed in Chapter 4.

FORTRAN subroutines PitchCntrl() and UserVSCont() implement the collective

pitch and generator torque control. Subroutine updateControlParameters() implements a low pass filter on generator speed and scales control parameters to transition the turbine into and out of derated operation. Subroutine UserHSSBr() models a brake that can be used to park the rotor at the end of an emergency shutdown. All four subroutines are in file UserSubs.f90. Module EAControl(), in FAST_Mods.f90, stores variables that are accessed by multiple control subroutines. FAST_IO.f90 has been modified so information that a turbine would receive from a plant level controller, such as when to derate the turbine, can be read in as part of the input file primary.fst. The source code containing this controller implementation is presented in Appendix E.

The simulations carried out in Chapter 4 do not model emergency turbine shutdowns due to rotor overspeeds. However, these events are important. Emergency shutdowns reduce power generation and can dramatically affect a turbine's wake. To ensure those effects are captured in our SOWFA simulations, emergency shutdown functionality is added to the turbine controller. The NREL 5-MW turbine specification does not describe an emergency shutdown protocol [2], so an emergency shutdown protocol is implemented based on the "aerodynamic shutdown" process described by Pedersen and Steineche.[78] If the turbine experiences a rotor overspeed in excess of 15% an emergency shutdown is initiated. The emergency shutdown protocol overrides the pitch and generator torque controllers. Generator torque is turned off and the turbine blades are collectively pitched to 90° at a rate of 8° per second. The pitched blades induce aerodynamic braking, which rapidly slows the turbine rotor. When the rotor is almost completely stopped a high-speed-shaft brake is initiated to ensure the rotor comes to a complete stop and remains stationary.

Plant level control can not be modeled in SOWFA at this time. Though NREL did develop a version of SOWFA capable of modeling plant level control [53, 79], it was never released publicly and is currently unavailable. For the simulations carried out in this chapter plant level control are implemented outside of SOWFA.

This is accomplished by running each simulation twice. First the simulation is run without plant level control. The results of the first simulation provide a performance baseline and allow plant level control signals to be generated offline. The second simulation is run with plant level control. Plant level control signals (when to derate the downwind turbine and by how much) are fed to the downwind turbine through the input file primary.fst. Results from the second simulation can be compared to results from the first simulation to determine how the plant level controller affects system performance.

6.3 Computational Domain and Grid Resolution

The LES computational domain used in this chapter is 5040 meters \times 2520 meters \times 2520 meters, or $80R \times 40R \times 40R$ where R is the NREL 5-MW rotor radius (63 m). As in Chapter 5 this rectangular grid is initially composed of 32 m \times 32 m \times 32 m cells, but portions of the domain are refined to decrease cell size near the turbine rotors and their wakes. Two configurations will be simulated, as illustrated in Figure ???. In configuration 1 the upwind turbine will be 10 rotor diameters (1260 meters) downstream of the center of the inlet while the downwind turbine is 10 rotor diameters further downstream. In this configuration the downwind turbine will be in the wake of the upwind turbine. Configuration 2 is similar, but each turbine is offset horizontally by 0.75 rotor diameter so the downwind turbine is not in the wake of the upwind turbine.

Choosing a grid resolution is a compromise. A fine grid captures flow behavior that a coarse grid doesn't capture. However, using a fine grid has a higher computational cost than a coarse grid, requiring more cores and/or more time to run a simulation. One way to balance simulation detail and computational cost is to make the grid resolution finer in the areas of interest (such as near the turbine rotor) and coarser in areas that are of less interest (such as far from the turbine). Another way is to make the grid resolution as fine as it needs to be, but not making it any finer.

For simulations in Section 5.4 a grid resolution of 1 meter was used near the tur-

Configuration #1: Turbine in Wake Test Case



Configuration #2: Offset Turbines Test Case

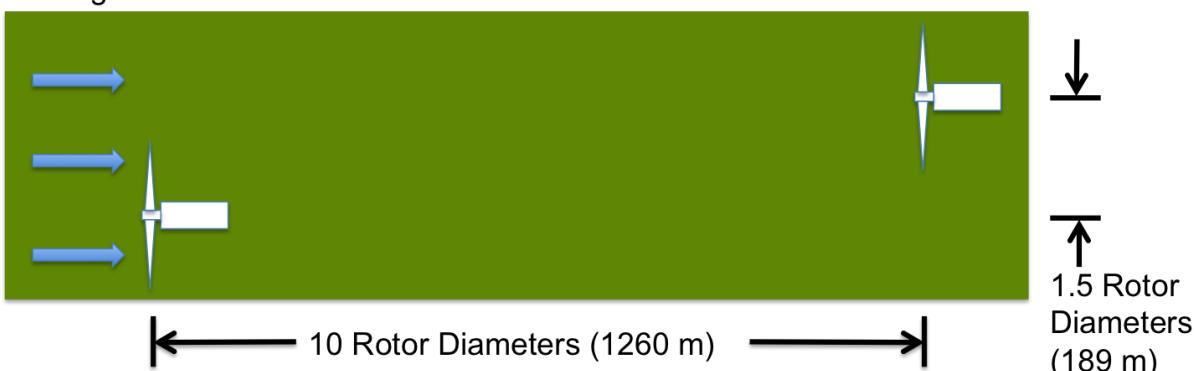


Figure 6.1. Configurations for SOWFA simulations of feed forward derating control.

bine rotor. This near-rotor grid was not very large, extending only 1 rotor diameter upstream, 6 rotor diameters downstream, and 1.4 rotor diameters radially, which is approximately 0.04% of the simulation domain volume. However, this near-rotor grid contains approximately 11 million cells, which is about 57% of the cells in the simulation domain. A $1 \text{ m} \times 1 \text{ m} \times 1 \text{ m}$ grid large enough to encompass the two turbine systems shown in Figure ?? would have a prohibitively large number of cells and have a prohibitively high computational cost.

To make the computational cost more manageable a 2 meter near-rotor grid resolution will be used for simulations in this chapter. Though simulation results will be less detailed, we still have high confidence in the accuracy of the results. As Section 5.6 shows, even a near-rotor grid resolution of 4 meters yielded good agreement for power generation, rotor thrust, wake vorticity, and momentum deficit.

6.4 Tuning and Validation of SOWFA

An actuator line model couples SOWFA's LES based atmosphere model to SOWFA's FAST based turbine dynamics model. To get accurate turbine performance from SOWFA, the actuator line model must be tuned. A series of simulations revealed that an actuator line model with 62 elements per blade and a Gaussian projection width of 7.5 meters yields good agreement between SOWFA and FAST simulations. These actuator line parameters comply with the best practices recommended by Churchfield, Lee, and Moriarty [51] as well as those recommended by Troldborg.[80] It is worth noting that the Gaussian projection width chosen here is different than the one chosen for SOWFA simulations in Chapter 5. This difference is partially due to the use of a 2 meter near-rotor grid resolution, but it is also caused by a difference in tuning requirements. The SOWFA simulations carried out in Chapter 5 did not model turbine control, so the actuator line was only tuned to produce good agreement on turbine loads and power generation. The Gaussian projection width chosen here produces good agreement on controller behavior as well as loading and power generation for wind speeds between 12 and 25 m/s (the range of wind speeds for which the turbine is operating in region 3 control).

To illustrate the close agreement between SOWFA and FAST simulations, three simple test cases from Chapter 4 are simulated in SOWFA. In these test cases the turbine is subjected to a constant, uniform wind. At 100 seconds the turbine is derated by 20%. At 200 seconds the turbine is returned to the original operation condition. Incoming wind speeds of 12 m/s, 16 m/s, and 20 m/s are simulated. FAST and SOWFA have very good agreement for rotor speed, power, and blade root bending moments. Figures ??, ??, and ?? show this agreement for the 16 m/s case. Results for the 12 m/s and 20 m/s case have been omitted to improve the clarity of these figures. However, the 12 m/s and 20 m/s cases exhibit similar behavior. There is also good agreement on blade pitch (Figure ??) and tower base fore-aft bending moment (Figure ??), though there are some differences. SOWFA predicts slightly lower blade pitch angles for the 16 m/s and 20 m/s cases while predicting slightly

higher blade pitch angles for the 12 m/s case. FAST predicts more low frequency oscillations in pitch following changes in turbine rating. This effect is most prominent in the 12 m/s case, but is present in all three cases. SOWFA predicts more high frequency oscillations in tower base bending moment than FAST. However, there is close agreement on the magnitudes of the predicted loads. The largest discrepancies in tower fore-aft bending moment are in the 12 m/s case where the discrepancy is typically less than 3%, but can be as high as 7.5% in the transient behavior following the turbine returning to full rated operation.

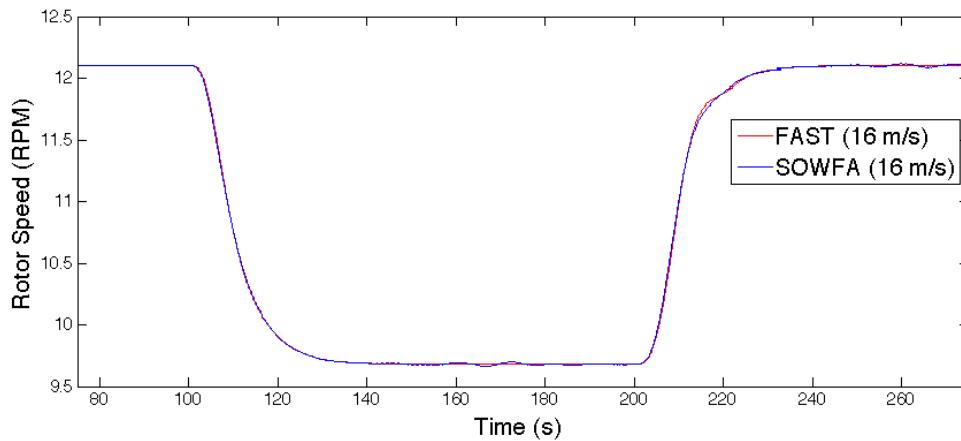


Figure 6.2. Comparison of rotor speed predicted by FAST and SOWFA for a turbine in 16 m/s winds that is derated by 20%.

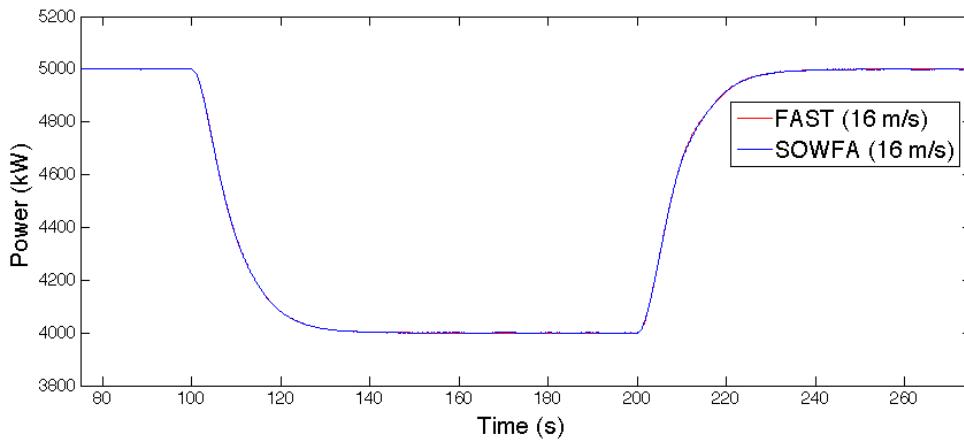


Figure 6.3. Comparison of power predicted by FAST and SOWFA for a turbine in 16 m/s winds that is derated by 20%.

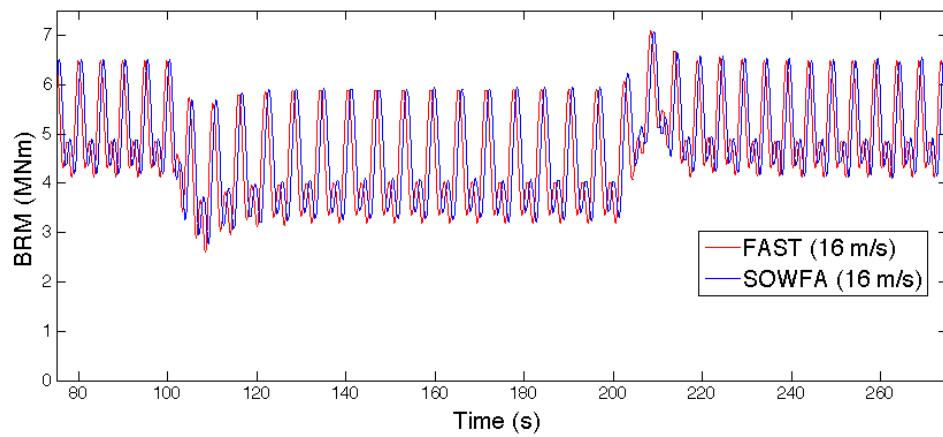


Figure 6.4. Comparison of blade blade root bending moment predicted by FAST and SOWFA for a turbine in 16 m/s winds that is derated by 20%.

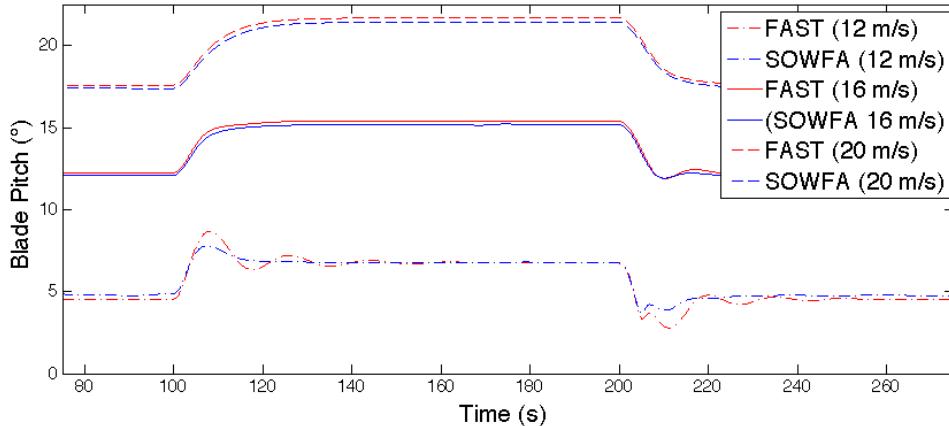


Figure 6.5. Comparison of blade pitch predicted by FAST and SOWFA for turbines in 12 m/s, 16 m/s, and 20 m/s winds that are derated by 20%.

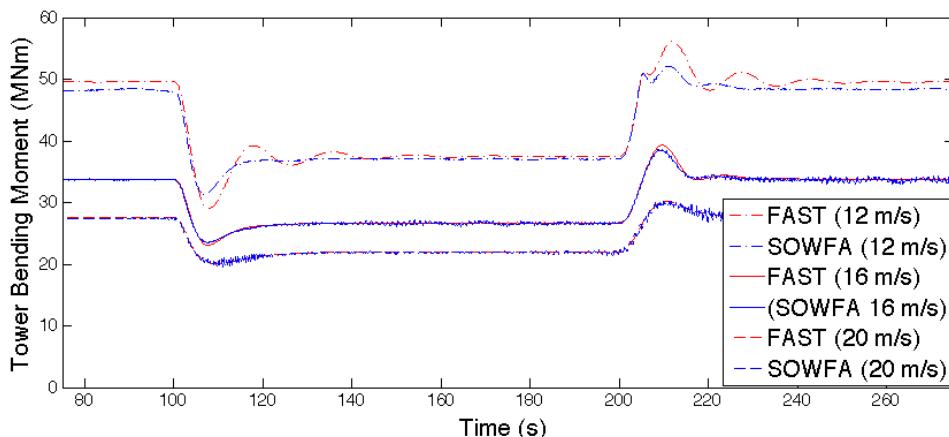


Figure 6.6. Comparison of tower base bending moment predicted by FAST and SOWFA for turbines in 12 m/s, 16 m/s, and 20 m/s winds that are derated by 20%.

6.5 Gust Modeling in SOWFA

For the FAST simulations in Chapter 4 gusts are modeled by simply increasing the incoming wind speed. However, this method does not work in SOWFA. SOWFA models airflow as an incompressible fluid in a finite computational domain. Increasing the wind speed across the inlet would cause an increase in the amount of air flowing into the computational domain. Because the fluid is incompressible and constrained, conservation of mass dictates that an increase in the amount of air flowing into the domain be immediately matched by an increase in the amount of air flowing out of the domain and an increase in the amount of air flowing from the input to the output. Therefore, increasing the wind speed across the inlet of the computational domain causes an instantaneous increase in wind speed throughout the computational domain.

To model a gust propagating through a wind farm using SOWFA we must be more subtle. This is done by increasing wind speed across part of the inlet, while decreasing wind speed elsewhere in the inlet. As long as the total amount of air flowing into the domain remains the same, the flow far downstream of the inlet will not be affected. Using this method, the gust is localized near the inlet initially then propagates through the computational domain over time.

In SOWFA, the user can specify a time varying velocity profile across the inlet by applying a `TimeVaryingMappedFixedValue` boundary condition. For this boundary condition, the user specifies a list of locations on the inlet plane, then specifies velocities at those points for several simulation times. SOWFA interpolates between the supplied data to determine the velocity profile across the inlet for all time steps in the simulation.

Several inlet velocity profiles were investigated in a series of preliminary SOWFA simulations. The inlet profile was found to have a large effect on how the gust initially appears in the simulation domain and how it behaves as it propagates through the domain. Inlet profiles that confine the gust to a small portion of the inlet were found to give the most control over gust behavior. When the gust covers a large por-

tion of the inlet profile, SOWFA smooths out the effect of the gust. This smoothing effect turns rapid velocity changes at the inlet into more gradual velocity changes within the simulation domain. In the following sections we use inlet profiles that confine the gust to a small portion of the inlet. On the remainder of the inlet, the velocity remains constant.

In Chapter 4 the turbines were subjected to a hat shaped extreme operating gust (EOG) as defined in section 6.3.2.2 of IEC61400-1.[36] Preliminary SOWFA simulations found that it is not possible to simulate a hat shaped gust that will propagate through the SOWFA simulations domain. A hat shaped fluctuation of the inlet velocity begins as a hat shaped gust near the inlet. However, the gust decreased in magnitude and changed shape as it moved through the simulation domain. Instead, simulations in the following sections will model an extreme coherent gust (ECG) similar to the one defined in section 6.3.2.5 of IEC61400-1.

Figure ?? illustrates the difference between an extreme operating gust (EOG) and an extreme coherent gust (ECG). The EOG is a brief fluctuation in wind speed, while the ECG is a sustained increase. IEC1400-1 defines the ECG as a 15 m/s increase in wind speed described by equation ??, where t_{gust} is the starting time of the gust. For simulations in this chapter, the 15 m/s ECG magnitude specified by IEC1400-1 is interpreted to be a maximum coherent gust magnitude, not a required magnitude. If a 15 m/s coherent gust is possible then smaller magnitude coherent gusts are also possible. Coherent gusts were found to propagate through the SOWFA simulation domain with only small changes to the magnitude and shape of the gust, as illustrated in Figure ??

$$U_{gust} = \begin{cases} 0 & \text{for } t < t_{gust} \\ 7.5(1 - \cos(\pi t/10))m/s & \text{for } t_{gust} \leq t < (t_{gust} + 10) \\ 15m/s & \text{for } \geq (t_{gust} + 10) \end{cases} \quad (6.1)$$

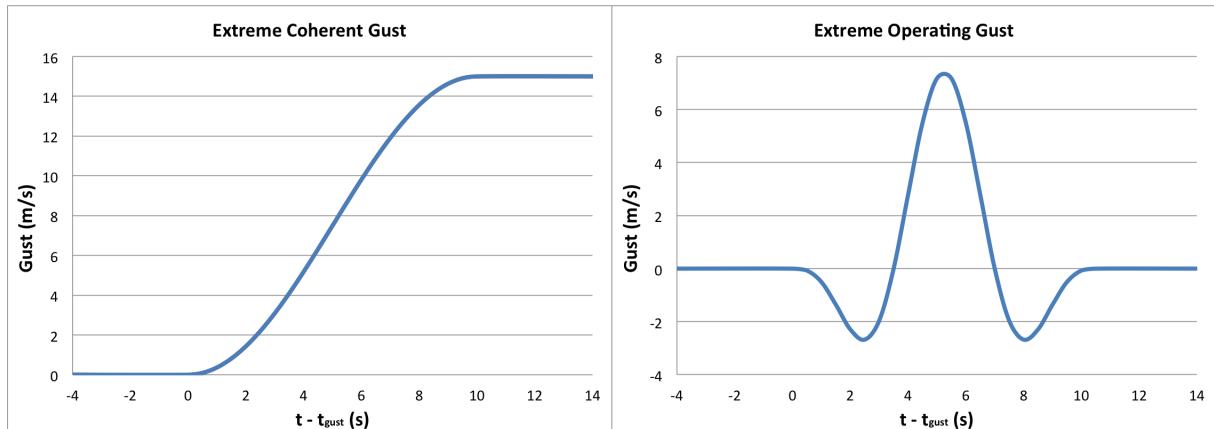


Figure 6.7. Velocity profiles for extreme operating gust (EOG) and extreme coherent gust (ECG).

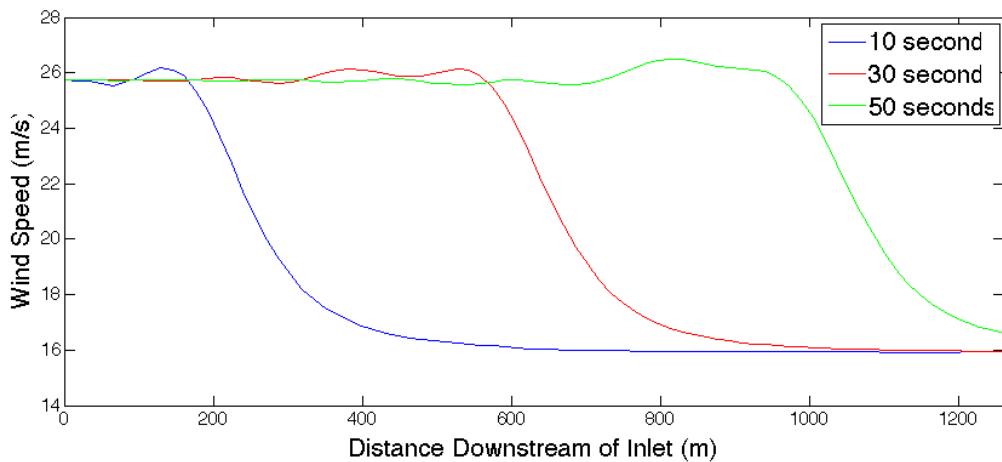


Figure 6.8. Center line velocity as ECG propagates through computational domain.

6.6 Offset Turbines Test Case

This section documents SOWFA simulations of the offset turbine test case illustrated in Figure ???. In this test case one turbine is 20R downwind of and 3R to the side of another turbine, where R is the rotor radius of 63 meters. Because of the 3R offset, the downwind turbine is not in the wake of the upwind turbine. This is the configuration that was assumed for FAST simulations of feed forward derating control carried out in Chapter 4.

6.6.1 Simulation Setup

The computational domain is $80R$ (5,040 m) long, extending $20R$ in front of the upwind turbine and $40R$ behind the downwind turbine. The domain is $40R$ tall, extending $20R$ above and below the turbine rotors. The domain is also $40R$ wide. Each rotor is $38.5R$ from one side and $41.5R$ from the other side due to the offset between the upwind and downwind turbine. Initially the domain is constructed of cube shaped cells approximately $32 \text{ m} \times 32 \text{ m} \times 32 \text{ m}$, but portions of the domain go through a series of grid refinements to reduce the size of cells to approximately $2 \text{ m} \times 2 \text{ m} \times 2 \text{ m}$ near the rotors. Table ?? describes the dimensions and resolutions of each refinement region. The first three refinement regions are rectangular cuboid shaped regions that encompass both turbines. These refinement regions are centered in the computational domain with respect to width and height. In the streamwise direction refinement regions 1, 2, and 3 begin $8.5R$, $4.5R$, and $2.5R$ (respectively) in front of the upwind turbine. Refinement regions 4 and 5 are cylindrical in shape and each encompass one of the two turbines. Each of these cylindrical refinement regions is centered on the rotational axis of a turbine rotor, extending $1.5R$ in front of and $12R$ behind the rotor.

Refinement Region	Streamwise Length(R)	Width (R)	Height (R)	Max Cell Size in Region
1	$68.5R$	$20.2R$	$17.2R$	$16 \text{ m} \times 16 \text{ m} \times 16 \text{ m}$
2	$64.5R$	$12.2R$	$8.2R$	$8 \text{ m} \times 8 \text{ m} \times 8 \text{ m}$
3	$46.5R$	$8.2R$	$5.2R$	$4 \text{ m} \times 4 \text{ m} \times 4 \text{ m}$
4 & 5		Radius = $1.6R$	$2 \text{ m} \times 2 \text{ m} \times 2 \text{ m}$	

Table 6.1. Dimensions and resolutions of the SOWFA grid refinement regions for the Offset Turbines test case. ($R = 63 \text{ m}$)

Each simulation is 700 seconds long, but the first 300 seconds are disregarded. This is to ensure that both turbines and their wakes have reached steady state before we start collecting data. Initially the inlet of the computational domain has a uniform 12 m/s wind speed. At 400 seconds a coherent gust is introduced in the

center of the computational domain inlet.

The coherent gust used in the offset turbine test case is illustrated in Figure ???. This is essentially a round gust that has been elongated by $3R$ (189 m) to ensure it encompasses both turbines. For the part of the gust between the rotor centerlines, wind speed is given by equation ??, where h_g is the distance above or below the rotor centerlines. For the part of the gust to the left or right of the rotor center lines, wind speed is given by equation ??, where r_g is the radial distance from the closest rotor centerline. As discussed in Section ??, the coherent gust contains both higher speed regions (up to 26.16 m/s) and lower speed regions (as low as 5.11 m/s) so the average wind speed across the computational domain inlet remains 12 m/s. For the regions directly upwind of each rotor, the average wind speed during the extreme operating gust is 25 m/s, which is the upper limit of the NREL 5-MW turbine operating range.

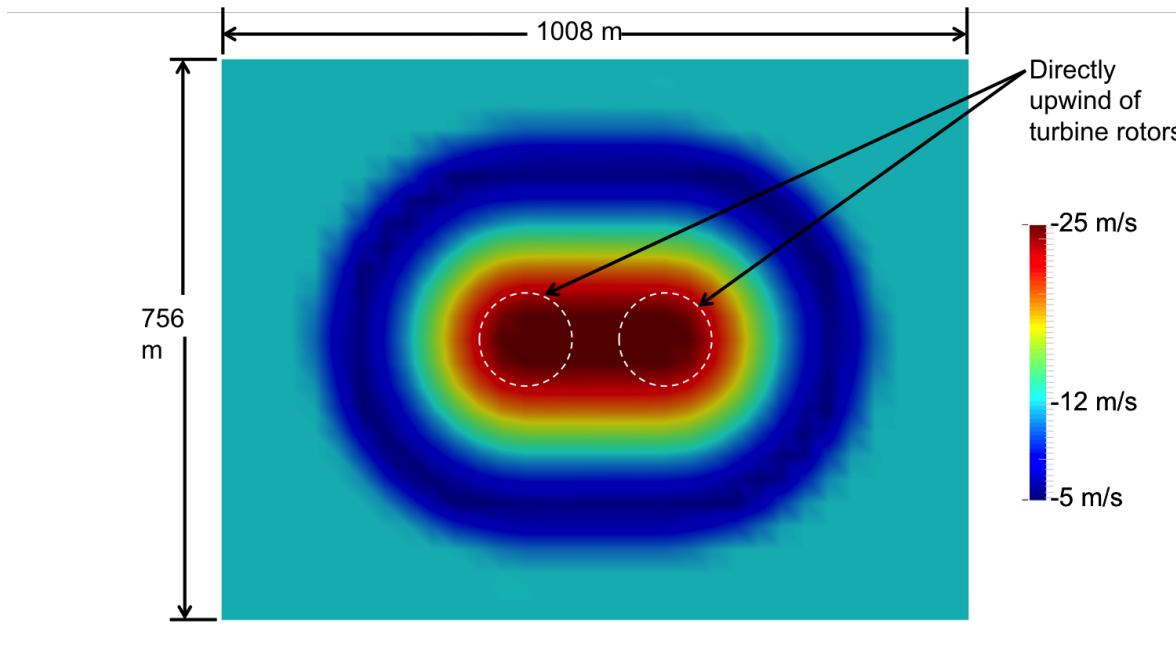


Figure 6.9. Wind speed near center of inlet during extreme coherent gust (offset turbine test case).

$$U_{gust} = \begin{cases} \left[15.6312 + 10.5248 \cos\left(\frac{h_g}{252}\pi\right) \right] m/s & \text{for } h_g < 252 \\ \left[8.5532 - 3.4468 \cos\left(\frac{h_g-252}{31.5}\pi\right) \right] m/s & \text{for } 252 \leq h_g < 283.5 \\ 12 m/s & \text{for } 283.5 \leq h_g \end{cases} \quad (6.2)$$

$$U_{gust} = \begin{cases} \left[15.6312 + 10.5248 \cos\left(\frac{r_g}{252}\pi\right) \right] m/s & \text{for } r_g < 252 \\ \left[8.5532 - 3.4468 \cos\left(\frac{r_g-252}{31.5}\pi\right) \right] m/s & \text{for } 252 \leq r_g < 283.5 \\ 12 m/s & \text{for } 283.5 \leq r_g \end{cases} \quad (6.3)$$

6.6.2 Simulation Results

This section documents the results of the Offset Turbine test case. This test case was run twice, once without feed forward derating control enabled and once with Feed forward Derating Control. The following figures and text analyze the behavior of the upwind turbine (which was identical in both runs of the Offset Turbine test case), the behavior of the downwind turbine when feed forward derating control is not enabled, and the behavior of the downwind turbine when feed forward derating control is enabled.

Figure ?? shows wind speed in the computational domain at several key moments when feed forward derating control is not enabled. Each image in Figure ?? was generated by taking a horizontal cross section through the center of the computational domain then trimming the cross section to remove regions that were far from the turbines in the downwind or horizontal directions. White lines have been superimposed on the images to show the locations and approximate sizes of the turbine rotors. At $t = 300$ s we see the turbines operating in a steady, uniform 12 m/s wind. The wakes of each turbine are clearly visible, and we see that the downwind turbine is not in the wake of the upwind turbine. At $t = 400$ s the extreme coherent gust (ECG) enters the computational domain. We see at $t = 420$ s that the gust has begun to propagate through the domain from left to right. Note that the gust has higher wind speeds at its center and lower wind speeds near its edges (as discussed in Section ??). At $t = 480$ s the gust has recently reached the upwind turbine. At t

= 540 s the gust arrives at the downwind turbine. In the $t = 540$ s image there is no wake trailing the upwind turbine. This indicates that the upwind turbine has shut down. At $t = 660$ s the gust has continued to propagate downstream and there is no wake trailing the downwind turbine, which indicates that it has also shut down. A similar set of images was generated for the Offset Turbine test case with Feed Forward Derating Control but is not shown here. Those images look very similar, except at $t = 600$ s. In the $t = 600$ s image a wake is visible trailing the downwind turbine, which indicates that feed forward derating control prevents the downwind turbine from shutting down.

It's noteworthy that the gust travels from the upwind turbine to the downwind turbine in approximately 70 seconds. This corresponds to a convection speed of approximately 18 m/s. For many of the simulations carried out in Chapters 3 and 4, Taylor's frozen turbulence hypothesis was assumed to be valid. This was done to allow the modeling of a two turbine system using FAST, but I noted at the time that the assumption was not expected to be completely valid. If Taylor's frozen turbulence hypothesis was valid for this test case we would expect the gust to propagate downstream at the average domain wind speed of 12 m/s. However, the gust propagates at neither the average wind speed (12 m/s) nor the gust wind speed (25 m/s), but somewhere in between. This reinforces the importance of using a feed forward control scheme that is tolerant of uncertainty about gust propagation speeds. As discussed in Section 4.5, the feed forward selective derating control scheme evaluated in this chapter is highly tolerant of uncertainty in gust propagation speed. The feed forward controller is not required to take any action at the exact moment the gust reaches the downwind turbine. It simply has to derate the turbine some time before the gust arrives and return the turbine to full rated operation some time after the gust has passed.

Figures ?? through ?? show the rotor speed, power generation, blade pitch, blade root bending moment, and tower base fore-aft bending moment of the turbines. In Figures ?? and ?? it can be seen that the turbines have reached steady state op-

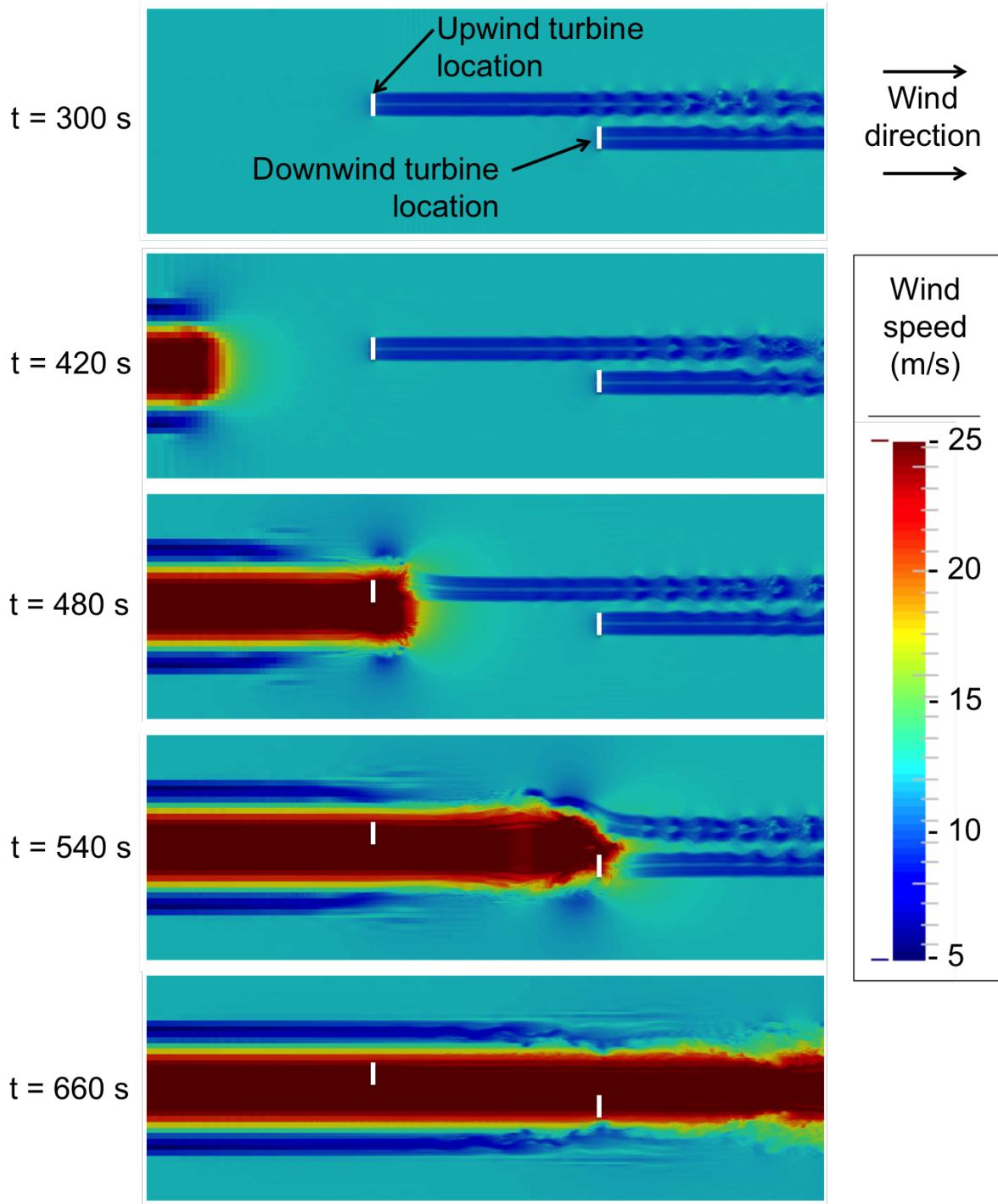


Figure 6.10. Wind speeds at several moments in the Offset Turbines test case without feed forward control.

eration by $t = 300$ s. The rotors are turning at a constant rate of 12.1 RPM and producing 5,000 kW of power. The extreme coherent gust (ECG) reaches the upwind turbine at approximately 450 s, causing both the rotor speed and power generation to increase. At $t = 469.7$ s the rotor speed exceeds the 15% overspeed limit and an emergency overspeed shutdown is initiated. If feed forward derating control is enabled, a signal is sent to the downwind turbine instructing it to derate by 10.37% beginning at $t = 481.7$ s and return to full rated operation at $t = 627.2$ s. The upwind rotor comes to a halt at $t = 480.9$ s and is stationary for the remaining 219.1 seconds of the simulation. In real world operation this turbine would remain stationary until an operator can verify that it is safe to return the turbine to operation. This could potentially be much longer than 219.1 seconds. Between $t = 481.7$ s and $t = 511.7$ s the downwind turbine with feed forward control smoothly reduces its rotor speed and power generation, while the turbine without feed forward control continues to operate at a rotor speed of 12.1 RPM and power generation of 5,000 kW. The ECG reaches the downwind turbine at approximately 520 s, causing both the rotor speed and power generation to increase. The downwind turbine without feed forward control exceeds the 15% overspeed limit at $t = 539.2$ s and an emergency overspeed shutdown is initiated. By $t = 550.5$ s the downwind turbine without feed forward control is shut down. The downwind turbine with feed forward derating control experiences a maximum overspeed of 6.9%, which does not cause an emergency overspeed shutdown. Between $t = 627.2$ s and $t = 657.2$ s the turbine transitions back to full rated operation.

Figure ?? shows the blade pitch of the turbines. Initially, when the turbines are operating in uniform 12 m/s wind, the turbines have a steady blade pitch of 4.8°. As the upwind turbine begins to experience increased wind speed its blade pitch increases. This is an effort by the turbine pitch controller to maintain a constant rotational speed of 12.1 RPM. We see in Figure ?? the upwind turbine has a blade pitch of 18.3° when the emergency shutdown procedure is initiated at $t = 469.7$ s. After the emergency shutdown is initiated blade pitch increases to 90° at a rate of 8

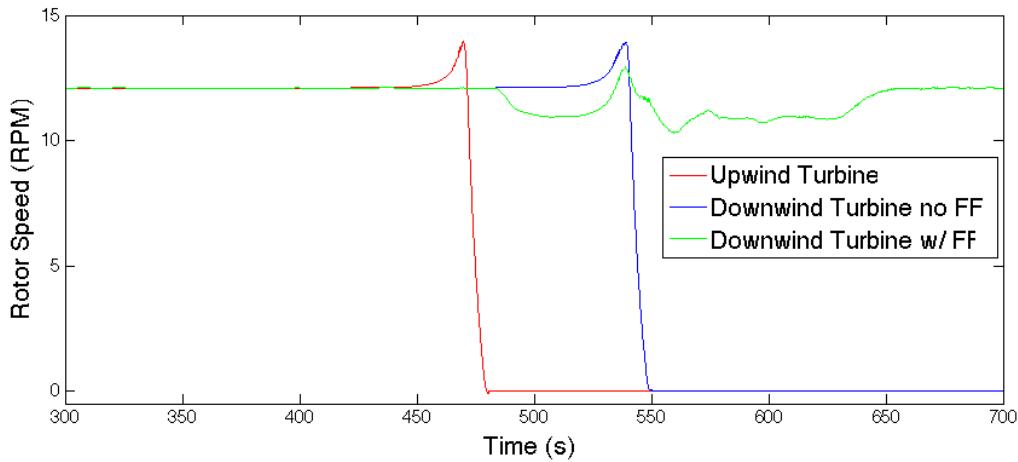


Figure 6.11. Rotor speed during the Offset Turbines test case without feed forward control.

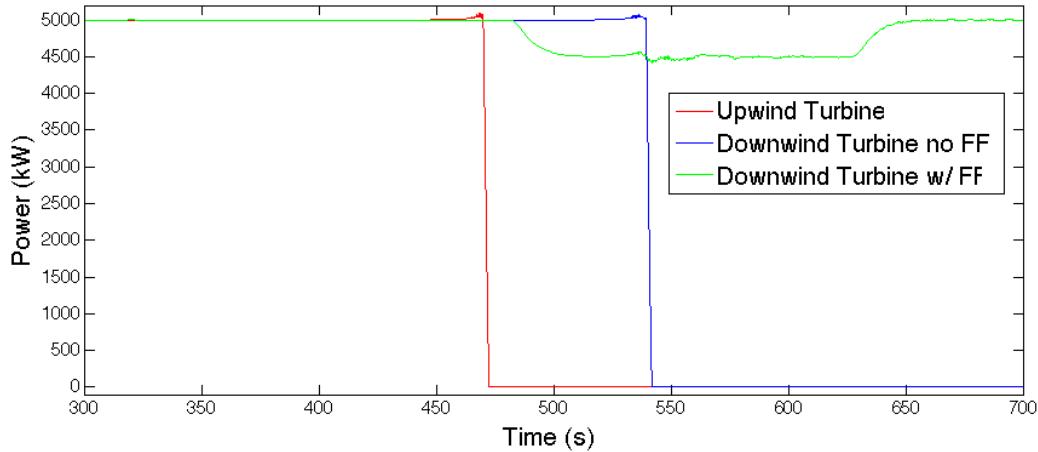


Figure 6.12. Power generation during the Offset Turbines test case without feed forward control.

deg/s. The downwind turbine without feed forward control shows similar behavior, reaching a blade pitch of 20.3° before the emergency shutdown is initiated at $t = 539.2$ s. The downwind turbine with feed forward derating control increases blade pitch from 4.8° to 6.1° when the turbine is derated (starting at $t = 481.7$ s). The gust causes an increase and some oscillation in blade pitch, with the blade pitch eventually settling to approximately 25.5° . When the turbine is returned to full rated operation, starting at $t = 627.2$ s, blade pitch decreases to 23.1° .

Figures ?? shows blade root bending moment (BRM). Initially, when the turbines

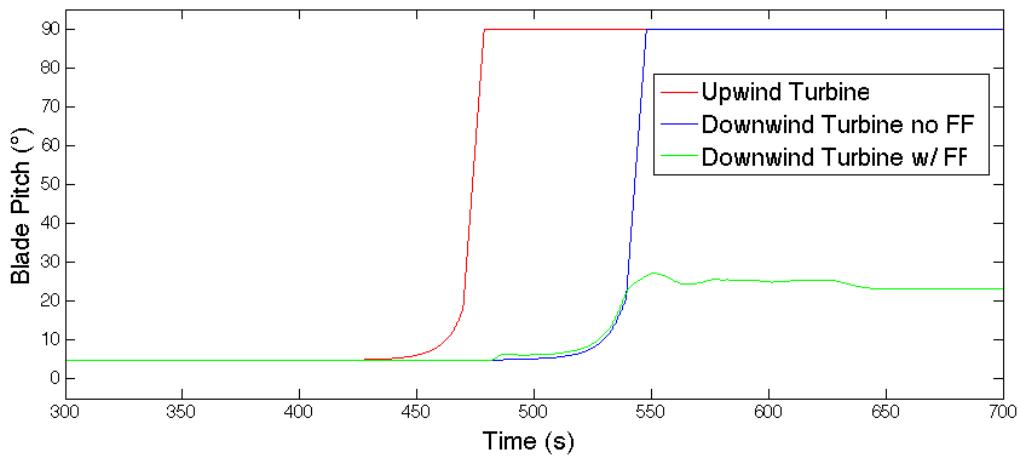


Figure 6.13. Blade pitch during the Offset Turbines test case without feed forward control.

are operating in uniform 12 m/s wind, the blade root bending moment is very regular. The upwind and downwind turbines experience nearly identical cyclical loading. When the upwind turbine begins to experience higher wind speeds and blade pitch begins to increase, the BRM loading cycle maintains its shape but begins to decrease in magnitude. This effect can be seen from $t = 430$ s to $t = 465$ s. During the emergency shutdown process the shape of the BRM loading cycle changes and larger fluctuations in BRM can be seen. When the rotor comes to a halt the BRM settles to a constant value. Note that BRM is not 0 MNm when the turbine rotor stops. This residual loading is due to gravitational loads. BRM would only be 0 MNm if this blade happened to stop pointing directly up or down. The downwind turbine without feed forward control shows similar behavior in response to the ECG and subsequent shutdown. The BRM loading cycle maintains its shape but begins to decrease in magnitude between $t = 500$ s and $t = 535$ s. The shape of the BRM loading cycle then becomes irregular and there are a few large fluctuations before BRM settles to a constant value. The downwind turbine with feed forward derating control experiences a 12% decrease in mean BRM and a 9% decrease in the magnitude of BRM fluctuations starting at $t = 481.7$ s. These changes are primarily due to the increase in blade pitch caused by derating. When the ECG arrives and blade pitch is increased further, the mean BRM decreases to approximately 3 MNm (60%

below the initial mean BRM) and the cyclical fluctuations in BRM have increased to approximately 4 MNm (140% above the initial magnitude of fluctuations).

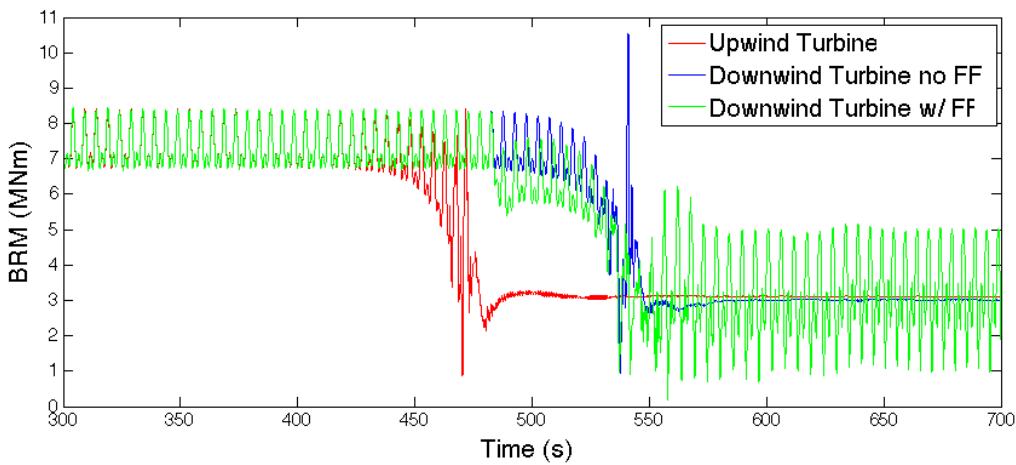


Figure 6.14. Blade root bending moment (BRM) during the Offset Turbines test case without feed forward control.

In Figure ?? we see that the upwind turbine and the downwind turbine without feed forward control exhibit similar behavior. Tower base fore-aft bending moments is initially constant at approximately 48 MNm. The tower base bending moment is highly dependent on the aerodynamic loading, which is highly dependent on blade pitch. As wind speeds begin to increase and blade pitches begin to increase the tower fore-aft bending moment smoothly decreases. The emergency shutdown process introduces a lot more oscillation in the tower base fore-aft bending moment, including one very large loading cycle where loading goes from approximately 35 MNm to approximately -40 MNm. This large spike in negative loading moment corresponds to the tower swinging forward when the aerodynamic loading on the rotor is quickly reduced to zero as the blades pitch to 90°. Eventually all of the high frequency oscillations in loading die out and the tower base fore-aft bending moment settles to 0 MNm. The downwind turbine with feed forward derating control also begins with a tower base fore-aft bending moment of 48 MNm, but experiences a reduction in loading when the turbine is derated (starting at $t = 481.7$ s). This is consistent with a reduction in aerodynamic loading on the turbine rotor due to increased blade pitch. When the ECG arrives, the downwind turbine sees a further

reduction in the average tower base fore-aft bending moment. However, we see the introduction of a cyclic loading component that persists through the end of the simulation. If we zoom in on these high frequency oscillations we see that the loading has a very similar shape to the cyclic BRM loading seen in Figure ??, but has a frequency three times higher. Since the turbine rotor has three blades, this indicates that the higher magnitude BRM oscillations seen near the end of the simulation are causing the high frequency oscillations in tower loading.

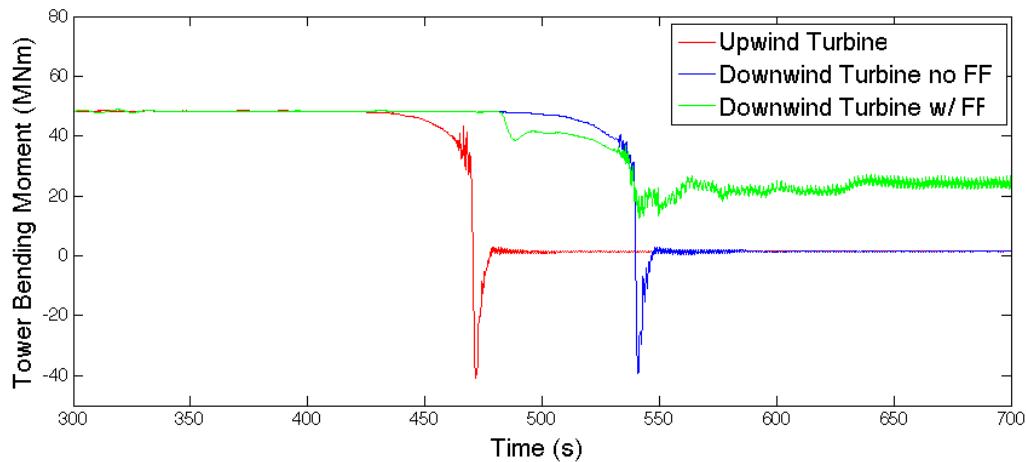


Figure 6.15. Tower base fore-aft bending moment during the Offset Turbines test case without feed forward control.

Table ?? summarizes several important performance metrics for the turbines. We see in the table that feed forward derating control improves turbine behavior in all four performance metrics. The upwind turbine and the downwind turbine without feed forward control have very similar tower base fore-aft bending moment Damage Equivalent Loads (DEL). Feed forward derating control reduces tower base fore-aft bending moment DEL by 44.2 MNm, which is a reduction of 59%. This is likely because the downwind turbine with feed forward derating control does not experience the large magnitude loading cycles observed during emergency shutdown. The downwind turbine without feed forward control experiences approximately 20% higher blade root bending moment (BRM) DEL than the upwind turbine. Part of this difference is because the downwind turbine operates for longer and experiences more BRM loading cycles before the gust arrives. A contributing factor may also be

the differences in the loading patterns observed near shutdown in Figure ???. These differences in loading patterns are caused by the interaction of the loading induced by the ECG and the cyclic gravitational loads due to the rotation of the turbine rotor. feed forward derating control reduces BRM DEL by 15% despite the turbine operating for a longer period of time and enduring a larger quantity of BRM loading cycles. The upwind turbine and the downwind turbine without feed forward control experience similar maximum overspeeds, while the downwind turbine with feed forward derating control has a significantly smaller maximum overspeed. Finally, it can be seen that the downwind turbine without feed forward control produces 96.46 KWh more energy than the upwind turbine. This is simply because the gust arrives at the downwind turbine 70 seconds after it arrives at the upwind turbine. This allows the downwind turbine without feed forward control to generate electricity for an additional 70 seconds before experiencing an emergency overspeed shutdown. feed forward derating control further increases energy capture by 201.27 KWh.

It is important to note that the difference in energy capture could be much larger in real world operation. These simulations end at $t = 700$ s, but beyond that time the downwind turbine with feed forward derating control would continue to generate 5 MW (or 1.4 kWh per second). The downwind turbine without feed forward control would not generate any power until it was returned to service. That turbine may be shut down for much longer than the 149.5 seconds captured in these simulations.

	Tower Base DEL(MNm)	Blade Root DEL(MNm)	Max Overspeed(%)	Energy Gen.(kWh)
Upwind Turbine	76.4	7.60	15.37	237.54
Downwind Turbine Without FF Control	75.2	9.15	15.04	334.00
Downwind Turbine With FF Control	31.0	7.82	6.90	535.27
Change Due to FF Control	-44.2 (-58.8%)	-1.33 (-14.5%)	-8.14	+201.27 (+60.2%)

Table 6.2. Summary of turbine performance metrics and the effects of feed forward derating control for the Offset Turbine test case.

6.7 Turbine In Wake Test Case

In previous chapters we assumed the downwind turbine would be slightly offset from the upwind turbine and would not be in that turbine's wake. This was done out of necessity because FAST is not capable of modelling turbine wake effects. However, SOWFA captures turbine wakes and can be used to study how those wakes affect downwind turbines. Mathew Churchfield and others at NREL have published several articles that use SOWFA to study such effects.[52, 75, 81, 82] Furthermore, wake effects are common in real world wind farms. This section documents SOWFA simulations of the Turbine In Wake test case illustrated in Figure ???. In this test case one turbine is 1260 m (or 20R, where R is the rotor radius) directly behind and in the wake of another turbine.

6.7.1 Simulation Setup

The simulation domain is $5040 \text{ m} \times 2520 \text{ m} \times 2520 \text{ m}$ (or $80R \times 40R \times 40R$). This domain extends 1260 m (20R) in front of the upwind rotor and 2520 m (40R) behind the downwind rotor, as well as 1260 m (20R) both horizontally and vertically from the centers of the rotors. The computational domain is initially composed of approximately $32 \text{ m} \times 32 \text{ m} \times 32 \text{ m}$ cells, but portions of the domain go through a series of grid refinements to reduce the size of the cells near the turbine rotors. The refinement regions are a series of 4 concentric cylinders with centerlines on the rotational axis of the NREL 5-MW rotors and having the dimensions shown in Table ???. In the streamwise direction, refinement regions 1, 2, 3, and 4 begin $8.5R$, $4.5R$, $2.5R$, and $1.5R$ (respectively) in front of the upwind turbine.

Refinement Region	Streamwise Length(R)	Radius (R)	Max Cell Size in Region
1	$68.5R$ (4316 m)	$8.6R$ (542 m)	$16 \text{ m} \times 16 \text{ m} \times 16 \text{ m}$
2	$64.5R$ (4064 m)	$4.6R$ (290 m)	$8 \text{ m} \times 8 \text{ m} \times 8 \text{ m}$
3	$46.5R$ (2390 m)	$2.6R$ (164 m)	$4 \text{ m} \times 4 \text{ m} \times 4 \text{ m}$
4	$33.5R$ (2111 m)	$1.6R$ (101 m)	$2 \text{ m} \times 2 \text{ m} \times 2 \text{ m}$

Table 6.3. Extents and resolutions of the SOWFA grid refinement regions for the Turbine In Wake test case.

Each simulation is 700 seconds long, but the first 300 seconds are disregarded. This is to ensure that both turbines and their wakes have reached steady state before we start collecting data. Initially the inlet of the computational domain has a uniform 12 m/s wind speed. At 400 seconds a coherent gust is introduced in the center of the computational domain inlet.

The coherent gust used for the turbine in wake test case is illustrated in Figure ?? and is described by equation ??, where r_g is the radial distance from the centerline of the turbine rotors. As discussed in Section ??, the coherent gust contains both higher speed regions (up to 26.41 m/s) and lower speed regions (as low as 7.59 m/s) so the average wind speed across the computational domain inlet remains 12 m/s. For the regions directly upwind of the turbine rotors the average wind speed during

the extreme operating gust is 25 m/s, which is the upper limit of the NREL 5-MW turbine operating range.

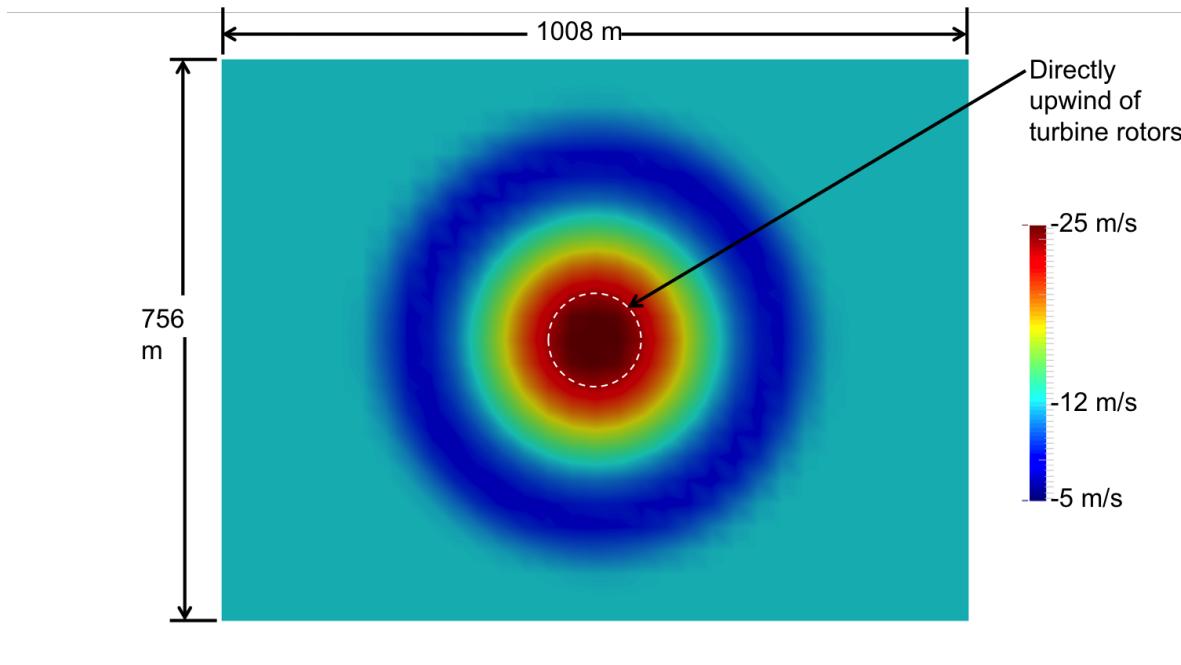


Figure 6.16. Wind speed near center of inlet during extreme coherent gust (Turbine In Wake test case).

$$U_{gust} = \begin{cases} [16.9991 + 9.4109 \cos\left(\frac{r_g}{252}\pi\right)] \text{ m/s} & \text{for } r_g < 252 \\ [9.794 - 2.2060 \cos\left(\frac{r_g - 252}{63}\pi\right)] \text{ m/s} & \text{for } 252 \leq r_g < 315 \\ 12 \text{ m/s} & \text{for } 315 \leq r_g \end{cases} \quad (6.4)$$

6.7.2 Simulation Results

This section documents the results of the Turbine In Wake test case. This test case was run twice, once without feed forward derating control enabled and once with feed forward derating control. The following figures and text analyze the behavior of the upwind turbine (which was identical in both runs of the Offset Turbine test case), the behavior of the downwind turbine when feed forward derating control is not enabled, and the behavior of the downwind turbine when feed forward derating control is enabled.

Figure ?? shows wind speed in the computational domain at several key moments when feed forward derating control is not enabled. Each image in Figure ??

was generated by taking a horizontal cross section through the center of the computational domain then trimming the cross section to remove regions that were far from the turbines in the downwind or horizontal directions. White lines have been superimposed on the images to show the locations and approximate sizes of the turbine rotors. At $t = 300$ s we see that the upwind turbine is operating in a steady, uniform 12 m/s wind. The downwind turbine is subject to a lower and less consistent wind speed because it is in the wake of the upwind turbine. At $t = 400$ s the extreme coherent gust (ECG) enters the computational domain. We see at $t = 420$ s that the gust has begun to propagate through the domain from left to right. At $t = 480$ s the gust has reached the upwind turbine. At $t = 540$ s the gust arrives at the downwind turbine. In the $t = 540$ s image there is no wake trailing the upwind turbine. This indicates that the upwind turbine has shut down. At $t = 660$ s the gust has continued to propagate downstream and there is no wake trailing the downwind turbine, which indicates that it has also shut down. A similar set of images was generated for the Turbine In Wake test case with feed forward derating control but is not shown here. Those images look very similar, except at $t = 600$ s. In the $t = 600$ s image a wake is visible trailing the downwind turbine, which indicates that feed forward derating control prevents the downwind turbine from shutting down.

Figures ?? through ?? show the rotor speed, power generation, blade pitch, blade root bending moment, and tower base fore-aft bending moment of the turbines. We see in Figures ?? and ?? that the upwind turbine is initially rotating at a constant rate of 12.1 RPM and producing 5,000 kW of power. This is expected for a turbine in 12 m/s wind because it is operating in region 3 control. The downwind turbine, which is in the wake of the upwind turbine, has a lower rotational speed and a lower power generation. The downwind turbine also has some fluctuations in rotor speed and power generation. The lower rotational speed indicates that the downwind turbine is operating in region 2 control, which does not track a constant rotational speed. Region 2 control uses generator torque to vary rotational speed in an attempt to maximize power generation. The ECG reaches the upwind tur-

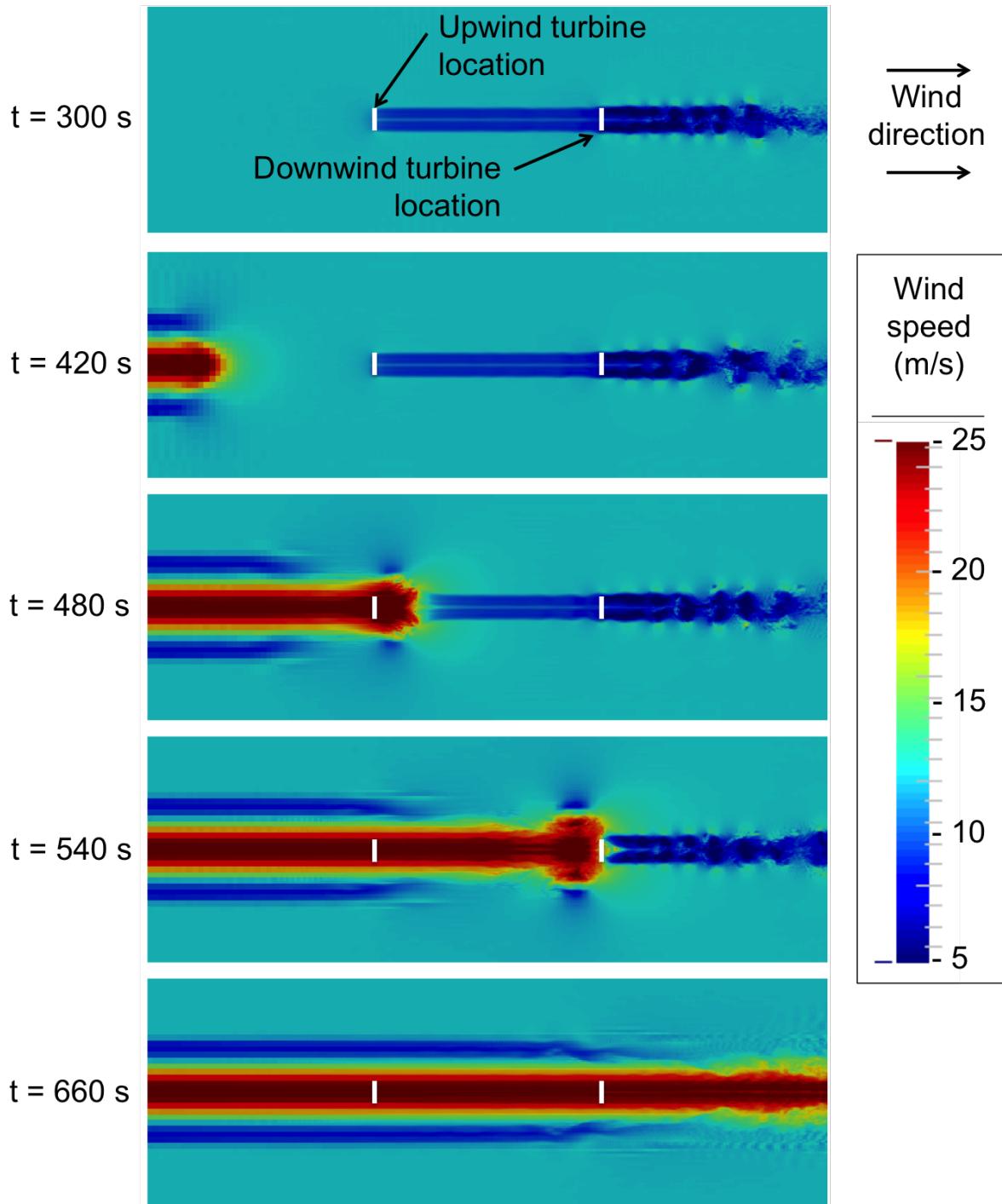


Figure 6.17. Wind speeds at several moments for the Turbine In Wake test case without feed forward control.

bine at approximately $t = 450$ s, triggering an emergency overspeed shutdown at $t = 466.7$ s. If feed forward derating control is enabled, a signal is sent to the downwind turbine instructing it to derate by 11.12% beginning at $t = 478.7$ s and return to full rated operation at $t = 624.2$ s. Between $t = 478.7$ s and $t = 508.7$ s the downwind turbine with Feed Forward Derate Control smoothly reduces its rotor speed and power generation, while the downwind turbine without feed forward control continues to operate according to normal region 2 control. The ECG reaches the downwind turbine at approximately $t = 520$ s. The downwind turbine without feed forward control experiences a large spike in rotor speed and power generation (a much larger change than what the upwind turbine experiences) as it is briefly put into region 3 operation. At $t = 538.6$ s the downwind turbine without feed forward control exceeds the 15% overspeed limit and an emergency overspeed shutdown is initiated. The downwind turbine with Feed Forward Derate Control experiences a maximum overspeed of 11.12%, which does not cause an emergency overspeed shutdown. Between $t = 624.2$ s and $t = 654.2$ s the downwind turbine transitions back to full rated operation.

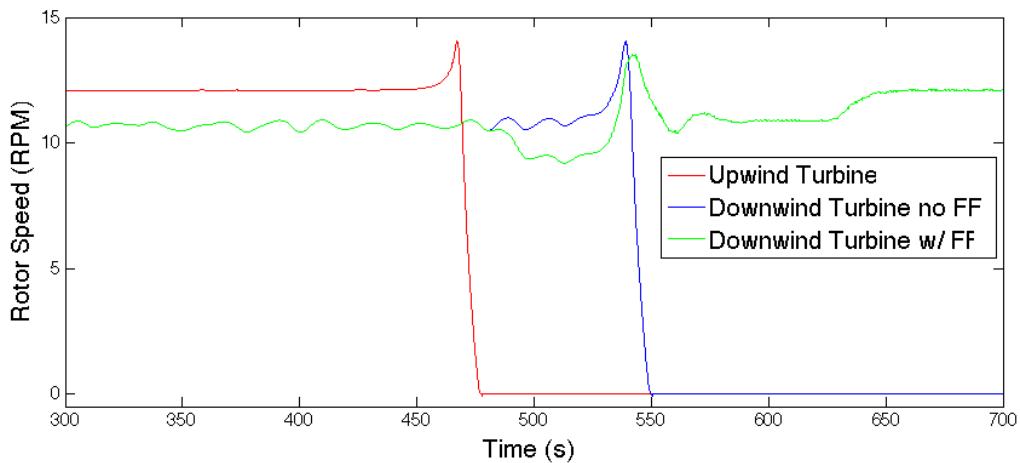


Figure 6.18. Rotor speed during the Turbine In Wake test case without feed forward control.

On average the downwind turbine is producing approximately 2800 kW of power prior to the extreme coherent gust. This is 44% less power than the upwind turbine, which illustrates that wake effects can have a very large impact on power production. The

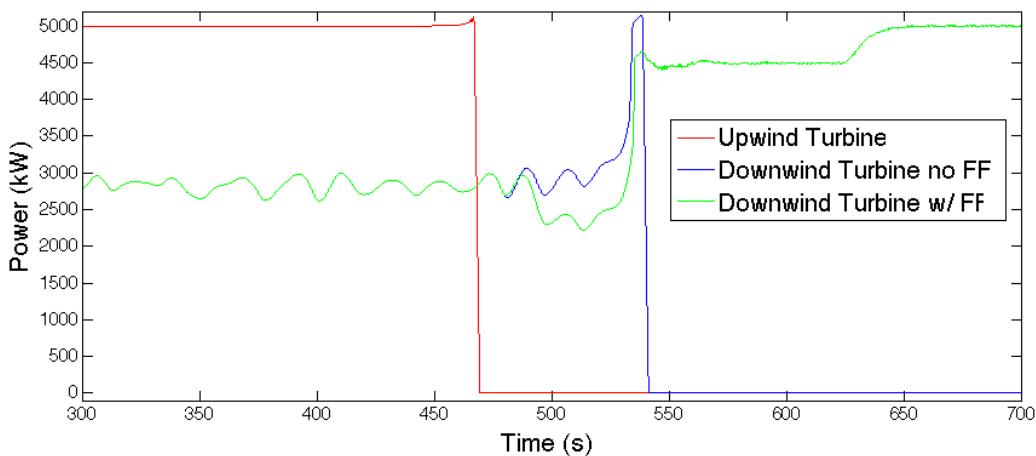


Figure 6.19. Power generation during the Turbine In Wake test case without feed forward control.

downwind power generation seen in Figure ?? is consistent with the rotor speeds seen in Figure ??, consistent with results shown in Chapter 5, and consistent with results that other researchers have published for wake effects 20R downstream of a turbine in turbulence free conditions. For an NREL 5-MW turbine, power production of 2800 kW corresponds to an effective incoming wind speed of 9.4 m/s. This indicates that the upwind turbine wake is causing an average wind velocity deficit of 22% at the downwind turbine (20R downstream). This is similar to the velocity deficit shown in Figure 5.5-b. A 22% wind velocity deficit due to the effect of an upwind turbine wake is also consistent with the findings of Troldborg et al. for RANS (Reynolds Averaged Navier Stokes) simulations of an NREL 5-MW turbine [83], and LES (Large Eddy Simulation) based analysis of a 2MW NEG Micon NM80 turbine.[84] Furthermore, a 22% wind velocity deficit is consistent with the dynamic wake meandering model described by Madsen et al.[85] However, not all turbine operating conditions will produce a 22% velocity deficit. These publications looked at a variety of turbine operating conditions and several noteworthy trends were found. Velocity deficit decreases as downstream distance increases. Velocity deficits decrease faster in turbulent conditions than in turbulence free conditions. Velocity deficits are largest when the turbine is in region 2 operation. When in region 3 operation, velocity deficit decreases as wind speed increases.

Figure ?? shows the blade pitch of the turbines. Initially, the upwind turbine has a blade pitch of 4.8° and the downwind turbine has a pitch of 0° . As the upwind turbine begins to experience increased wind speed (at $t = 450$ s) its blade pitch increases in an effort to maintain a constant rotational speed of 12.1 RPM. The upwind turbine's blade pitch is at 15.3° when the emergency shutdown procedure is initiated at $t = 466.7$ s. After the emergency shutdown is initiated blade pitch increases to 90° at a rate of 8 deg/s. The downwind turbine begins to experience increased wind speed at $t = 520$ s. The downwind turbine without feed forward control maintains a blade pitch of 0° until the rotor speed increases to 12.1 RPM. Once the rotor speed reaches 12.1 RPM the turbine transitions from region 2 control to region 3 control and increases blade pitch in an effort to maintain that rotational speed. The upwind turbine's blade pitch is at 10.4° when the emergency shutdown procedure is initiated at $t = 538.6$ s. The downwind turbine with feed forward derating control increases blade pitch from 0° to 2.6° when the turbine is derated (starting at $t = 478.7$ s). The gust causes an increase and some oscillation in blade pitch, with the blade pitch eventually settling to approximately 25.5° . When the turbine is returned to full rated operation, starting at $t = 624.2$ s, blade pitch decreases to 23.1° .

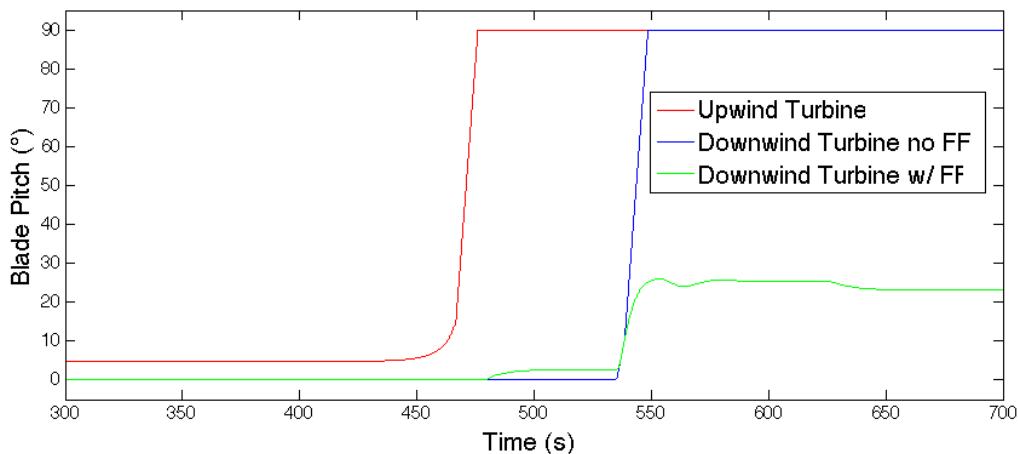


Figure 6.20. Blade pitch during the Turbine In Wake test case without feed forward control.

Figures ?? shows blade root bending moment (BRM). The BRM loading behavior

for the upwind turbine looks very similar to what we saw in the Offset Turbine test cases (Figure ??). Before the extreme coherent gust arrives, BRM loading cycles are very consistent in both shape and magnitude. When the upwind turbine begin to experience higher wind speeds and blade pitch begins to increase, the BRM loading cycle maintains it's shape but begins to decrease in magnitude. During the emergency shutdown process the shape of the BRM loading cycle changes and larger fluctuations in BRM can be seen. When the rotor comes to a halt the BRM settles to a constant value. Early in the simulation, because of wake effects, BRM loading cycles for the downwind turbine are less consistent in shape and BRM has a slightly lower magnitude on average. For the downwind turbine without feed forward control the gust causes an increase in BRM (at $t = 520$ s). This brief period of high loading is because the turbine maintains a 0° blade pitch angle until the rotor speed increases to 12.1 RPM. The combination of high wind speed and low blade pitch angle causes very high aerodynamic loading on the turbine blade. When emergency shutdown is initiated the shape of the BRM loading cycle then becomes irregular and there are a few large fluctuations before BRM settles to a constant value. Note that the downwind turbine without feed forward control experiences larger BRM loads than the upwind turbine during the emergency overspeed shutdown. The downwind turbine with feed forward derating control experiences a small decrease in BRM loads when the turbine is derated at $t = 478.7$ s. There is a brief increase in BRM loads when the ECG reaches the turbine at approximately $t = 520$ s. This period of high loading occurs while the downwind turbine is transitioning from region 2 derated control to region 3 derated control. By $t = 550$ s the mean BRM decreases to approximately 3 MNm (60% below the initial mean BRM) and the cyclical fluctuations in BRM have increased to approximately 4 MNm (140% above the initial magnitude of fluctuations).

In Figure ?? it can be seen that the tower base fore-aft bending moments for the upwind turbine is initially constant at approximately 48 MNm. As wind speeds begin to increase and blade pitch begins to increase there is a smooth reduction in

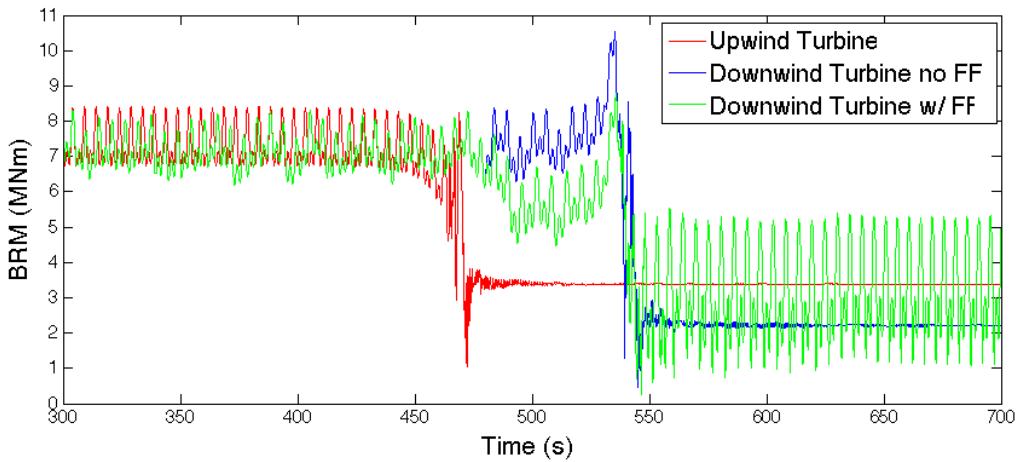


Figure 6.21. Blade root bending moment (BRM) during the Turbine In Wake test case without feed forward control.

tower fore-aft bending moment. The emergency shutdown process causes a short period of cyclical loading, including one large loading cycle where the tower base fore-aft bending moment goes from 47 MNm to -41 MNm. Early in the simulation the downwind turbine has lower tower base fore-aft bending moments, but has more fluctuations in loading. This is due to wake effects. The downwind turbine without feed forward control experiences a sharp increase in loading when the gust arrives at $t = 520$ s. As discussed in the previous paragraph, this is because the downwind turbine maintains a 0° blade pitch angle until the rotor speed increases to 12.1 RPM. The combination of high wind speed and low blade pitch angle causes very high aerodynamic loading. The emergency shutdown process causes a short period of cyclical loading, including an abrupt change from 76 MNm to -46 MNm. The downwind turbine with feed forward derating control is derated at $t = 478.7$ s. This causes a reduction in tower base fore-aft bending moment. When the gust arrives loading sharply increases to 63.4 MNm. This brief period of high loading occurs as the turbine transitions from region 2 to region 3 control. By $t = 550$ s the mean tower fore-aft bending moment has decreased to approximately 2 MNm. However, a cyclic loading component persists through the end of the simulation. If we zoom in on these high frequency oscillations, we see that the loading has a very similar shape to the cyclic BRM loading seen in Figure ??, but has a frequency three times

higher. Since the turbine rotor has three blades, this indicates that the higher magnitude BRM oscillations seen near the end of the simulation are causing the high frequency oscillations in tower loading.

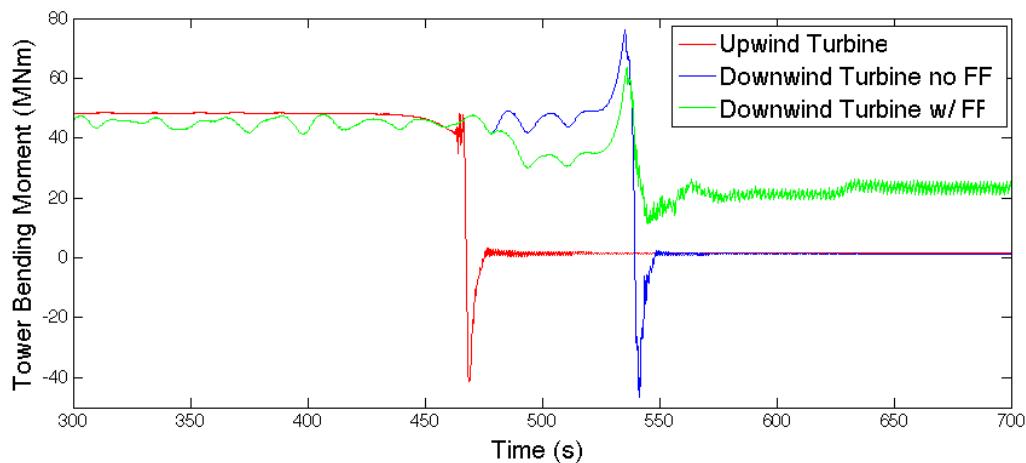


Figure 6.22. Tower base fore-aft bending moment during the Turbine In Wake test case without feed forward control.

Table ?? summarizes several important performance metrics for the turbines. The downwind turbine without feed forward control has much higher Damage Equivalent Loads (DEL) than the upwind turbine. The upwind turbine and the downwind turbine without feed forward control experience similar maximum overspeeds. The upwind turbine captures more energy than the downwind turbine without feed forward control despite operating for a shorter time. This is because the downwind turbine without feed forward control generates less power while it is operating due to wake effects (as seen in Figure ??). Feed forward derating control improves turbine behavior in all four performance metrics. Tower base fore-aft bending moment DEL has been reduced by 58.1 MNm, which is a reduction of 56%. This is likely because the downwind turbine with feed forward derating control does not experience the very large magnitude loading cycles observed immediately prior to and during emergency shutdown. Blade root bending moment DEL has been reduced by 15% despite the turbine operating for a longer period of time and enduring a larger quantity of BRM loading cycles. Maximum overspeed has been reduced to 11.7%, which will not induce an emergency shutdown. Energy generation has been

increased by 201.14 KWh.

By examining Figures ?? and ?? it is easy to see why the downwind turbine without feed forward control has higher DELs than the upwind turbine. The downwind turbine has much higher loads and more loading fluctuations due to the extreme coherent gust (ECG) and the subsequent emergency shutdown. However, I think it's useful to discuss what this means for the turbines and what the root cause of this discrepancy is. This discrepancy means that an extreme coherent gust large enough to cause emergency shutdown can be more damaging to a turbine in a wake than to a turbine that is not in a wake. The downwind turbine has to endure the change in wind speed caused by the gust in addition to a change in wind speed due to the upwind turbine shutting down. The upwind turbine is initially operating in 12 m/s wind then experiences a sudden increase of 13 m/s to a 25 m/s wind speed. The downwind turbine is initially operating in an average wind speed of 9.4 m/s then experiences a sudden increase of 15.6 m/s to a 25 m/s wind speed.

It is important to note that the difference in energy generation could be much larger in real world operation. These simulations end at $t = 700$ s, but beyond that time the downwind turbine with feed forward derating control would continue to generate 5 MW (or 1.4 KWH per second). The downwind turbine without feed forward derating control would not generate any power until it was returned to service. That turbine may be shut down for much longer than the 158.8 seconds captured in these simulations.

	Tower Base DEL(MNm)	Blade Root DEL(MNm)	Max Overspeed(%)	Energy Gen.(kWh)
Upwind Turbine	76.4	6.92	16.12	233.39
Downwind Turbine Without FF Control	104.0	9.49	16.03	194.51
Downwind Turbine With FF Control	45.9	8.03	11.70	395.65
Change Due to FF Control	-58.1 (-55.8%)	-1.46 (-15.4%)	-4.33	+201.14 (+103.4%)

Table 6.4. Summary of turbine performance metrics and the effects of feed forward derating control for the Turbine In Wake test case.

6.8 Conclusions

Chapter 4 investigated the benefits and feasibility of derating a downwind turbine based on a feed forward signal from an upwind turbine. In that chapter, a feed forward derating control scheme was developed and evaluated using a series of FAST simulations. Though the FAST based simulation results were promising, the simulation methodology used to generate them had several noteworthy limitations. The FAST based simulations did not model emergency overspeed shutdowns or turbine to turbine wake effects. Furthermore, the FAST based simulations assumed Taylor's Frozen Turbulence Hypothesis, which provides a simplistic model of wind speed fluctuations passing through the wind farm.

In this chapter, feed forward derating control was demonstrated and evaluated using a more sophisticated simulation tool that does not have the limitations of the FAST simulations carried out in Chapter 4. NREL's Simulator fOr Wind Farm Applications (SOWFA) is a wind farm simulation tool. It uses FAST to model the

dynamics of one or more turbines, a Large Eddy Simulation (LES) to model atmospheric airflow, and actuator line models to enable interaction of the LES and FAST models. Because SOWFA models atmospheric airflow, we can run simulations that capture the evolution of a gust over time, wake effects, and the time it takes a gust to reach the downwind turbine. As shown in Chapter 5, SOWFA can be tuned to provide good agreement with both Reynolds Averaged Navier Stokes (RANS) and FAST simulations of the NREL 5-MW turbine.

In this chapter, feed forward derating control was implemented as a series of FORTRAN subroutines and inserted into the SOWFA source code. Additional control logic was inserted to model emergency overspeed shutdowns (Section ??). SOWFA's actuator line model was tuned to produce good agreement with FAST. A Gaussian projection width of 7.5 produced good agreement across the range of region 3 turbine operation (Section??). The trade offs involved in choosing appropriate computational domain sizes and grid resolutions were examined. Ultimately a near rotor grid resolution of 2 m and a far field resolution of 32 m were chosen. The computational domain size was chosen to be $80R \times 40R \times 40R$. (Section ??). Section ?? discussed gust modeling in SOWFA. The gust was modeled using a mapped fixed value boundary condition on the inlet to the computational domain. Net inflow was held constant by increasing wind speed on some parts of the inlet while decreasing wind speed on other parts. After preliminary simulations, an extreme coherent gust (ECG) was chosen instead of the Extreme Operating Gust (EOG) used in Chapter 4 simulations.

Feed forward derating control was demonstrated and evaluated using two test cases. In the first test case one NREL 5-MW turbine was 20R downwind and 3R offset from another NREL 5-MW turbine. The turbines operated in a constant 12 m/s wind until an ECG increased wind speed to 25 m/s. Without feed forward derating control, the ECG caused emergency overspeed shutdowns of both turbines. Both turbines experienced similar overspeeds and Damage Equivalent Loads (DEL). In the second test case one NREL 5-MW turbine was 20R directly downwind of and

in the wake of another NREL 5-MW turbine. As in the previous test case, wind speed is 12 m/s until an extreme coherent gust (ECG) increases wind speed to 25 m/s. In this test case the upwind turbine's wake significantly reduced the power generation of the downwind turbine. Prior to the ECG's arrival the upwind turbine generated 5,000 kW while the downwind turbine produced approximately 44% less power on average. Without feed forward derating control the ECG caused emergency overspeed shutdowns of both turbines. The downwind turbine experienced much higher Damage Equivalent Loads than the upwind turbine, especially the tower fore-aft bending moment DEL.

Feed forward derating control was able to significantly improve performance in both test cases. When the test cases were run with feed forward derating control enabled, the downwind turbine was able to avoid an emergency overspeed shutdown. Rotor overspeed and DELs were significantly decreased while energy capture was dramatically increased.

These test cases illustrate that feed forward derating control can enable a turbine to avoid an emergency shutdown, reduce structural loading and increase energy capture. The usefulness of feed forward derating control will vary based on site specific conditions, such as how frequent emergency overspeed shutdowns occur. However, it would be very inexpensive to implement because it does not require any sensors that modern turbines don't already have. At a site with favorable conditions, feed forward derating control could provide a significant improvement in performance for a minimal financial investment.

Chapter 7

Conclusions

7.1 Summary and Findings

This dissertation seeks to improve the performance of a downwind turbine using information obtained from an upwind turbines. The preceding chapters investigate how an upwind turbine can be used as a sensor, then develop and test two possible feed forward control schemes. The feasibility of each control schemes is investigated using a variety of test cases and a variety of wind turbine simulation tools.

Chapter 2 investigates using a wind turbine as a sensor. It discusses the sensors that are present on a typical wind turbine and how sensor data can be used to gather useful information for feed forward control. Much of the chapter focuses on estimating wind speed. Anemometer wind speed measurements and wind speed estimates based on an inverted dynamic model of the turbine are examined. The model based estimator is found to have better performance, giving more accurate estimates of the average wind speed across the turbine rotor. The discrepancy between model based wind speed estimates and rotor average wind speed is approximately 3% over most of the NREL 5-MW turbine's operating range. The effect of turbine yaw misalignment on model based wind speed estimation is also discussed. Yaw misalignment introduces a significant amount of error in wind speed estimates. However, the error can be compensated for if the amount of misalignment is known. Convection velocity is also discussed, as well as how to estimate it from wind speed estimates

at a single location. Longer averaging times are found to produce more accurate convection velocity estimates, but there are practical limits on how long the averaging time can be. Finally, Chapter 2 discusses other turbine sensors, such as wind vanes or rotor speed sensors, and how they could be useful for feed forward control.

In Chapter 3, a feed forward optimum pitch control scheme is introduced and investigated. In this control scheme an upwind is used as a wind speed sensor. Wind speed measurements are passed from the upwind turbine to a feed forward controller on a downwind turbine. The downwind turbine uses the knowledge of incoming wind speed to proactively adjust turbine blade pitch, potentially improving energy capture and reducing damage to the turbine. The control scheme is investigated through three rounds of simulation using the NREL FAST turbine simulation tool. The initial round of simulations uses a simplified, idealistic model of the two turbine system, while the subsequent rounds systematically incorporate complicating factors that a real world system would encounter.

In the first two rounds of simulations, feed forward control significantly improves turbine performance with respect to both extreme operating gusts and turbulent wind. However, the third round of simulations shows that the feed forward optimal pitch control system is very sensitive to timing errors. If the timing of the data is off by more than a few seconds the feed forward control system can cause a turbine to perform more poorly than a turbine without feed forward control. The system's high sensitivity to timing errors, and therefore the systems high sensitivity to errors in the estimation of the convection velocity, ultimately make this control scheme impractical in real world applications.

Chapter 4 investigates the benefits and feasibility of derating a downwind turbine based on a feed forward signal from an upwind turbine. Several derating strategies from literature are investigated to determine which is most feasible for feed forward control. Derating a turbine by reducing the rated rotor speed is found to yield the largest reductions in structural loads and rotor overspeeds. The effects of derating on structural loading and rotor overspeed are examined. Reductions in ro-

tor speed, power, blade root bending moments, and tower fore-aft bending moments are found to be roughly proportional to the amount of derating. The transition into and out of derated mode is also investigated. Abruptly changing the rating of an NREL 5-MW turbine excites undesirable transient behavior. However, the undesirable behavior can be avoided by using a low pass filter.

A feed forward selective derating scheme based on these findings is developed. That feed forward control system is then evaluated using a series of FAST simulations. The FAST simulations model a two turbine system in several wind conditions, including both IEC Extreme Operating Gusts and turbulent wind with large gusts. The simulation results are promising. The downwind turbine experiences significant reductions in tower fore-aft moments in all simulation cases, as well as reductions in blade root bending moments in some cases. More importantly, the downwind turbine does not experience any rotor overspeeds large enough to trigger an emergency shut down. The downwind turbine experiences reductions in energy capture, however those reductions are smaller than the reductions that would be caused by an emergency overspeed shut down.

Chapter 5 is a verification and validation study of SOWFA (Simulator fOr Wind Farm Applications), a wind farm simulation tool. FAST must model a single turbine at time, but SOWFA can model a group of wind turbines as well as the atmospheric flow field surrounding those turbines. This gives SOWFA several advantages over FAST when modeling a two turbine system with feed forward control. However, SOWFA is much more computationally expensive than FAST and is less established in the wind turbine community. In Chapter 5 the NREL 5-MW rotor is simulated using SOWFA, OVERFLOW2 and FAST. The results from these fundamentally different simulation tools are compared. Overall the results from OVERFLOW2 and SOWFA are found to have good agreement. Power and thrust predictions are reasonably close for all test cases, as are the downstream momentum deficit predictions. The vortical structure of the wakes predicted by OVERFLOW2 and SOWFA are found to be very similar for the first $2R$ downstream of the rotor though some

differences are seen further downstream.

In addition, Chapter 5 evaluates how near-rotor grid resolution and near-rotor grid length affect SOWFA simulation results. Lengthening the near rotor grid has no effect on wake behavior, though shortening the near rotor grid has a small effect. Increasing the near rotor grid resolution to 4 m slightly increases the power and thrust predicted by SOWFA. It also results in the loss of some detailed wake behavior, such as rotor tip vortices in the 8 m/s and 11 m/s test cases. However, the downstream momentum deficit predictions still have good agreement. Overall, the results documented in Chapter 5 increase confidence in SOWFA and provide guidance for future SOWFA simulations.

In Chapter ??, Feed Forward Derating Control is simulated and evaluated using SOWFA. The first portion of the chapter documents background work required to simulate the control system in SOWFA. Feed forward derating control is implemented as a series of FORTRAN subroutines and inserted into the SOWFA source code. SOWFA's actuator line model is tuned to produce good agreement with FAST. The trade offs involved in choosing appropriate computational domain sizes and grid resolutions are examined. In addition, a method for modeling large gusts in SOWFA is developed.

The second portion of the chapter documents SOWFA simulation results for two test cases. In the first test case one NREL 5-MW turbine is 20R downwind and 3R offset from another NREL 5-MW turbine. In the second test case one NREL 5-MW turbine is 20R directly downwind of and in the wake of another NREL 5-MW turbine. The turbines operate in a constant 12 m/s wind until an extreme coherent gust (ECG) increased wind speed to 25 m/s. Feed forward derating control significantly improves performance in both test cases. Rotor overspeeds and DELs were significantly decreased while energy capture is dramatically increased by avoiding emergency overspeed shutdowns. At a site with favorable conditions, feed forward derating control could provide a significant improvement in performance for a minimal financial investment.

7.2 Future Work

The research presented in this dissertation demonstrates that feed forward derating control can provide significant performance improvements in the right conditions. However, there are several lines of future research that should be pursued.

The first line of research is to continue modeling feed forward derating control in more sophisticated and more realistic scenarios. The simulations carried out in Chapter ?? are more sophisticated and closer to real world operation than the simulations carried out in Chapter 4. However, there are still several real world factors that were not captured in these simulations. It would be beneficial to study the complications that would be caused by larger numbers of turbines, changes in wind direction, complex terrain, and atmospheric turbulence.

In this dissertation feed forward derating control was modeled using a two turbine system where the wind direction was known. This made the turbine to turbine communication simple. It was clear which turbine was operating as a sensor and which turbine was using that sensor data for feed forward control. Turbine to turbine communication becomes more complicated when there are more than two turbines. In a wind plant each turbine would have many other turbines it could share information with. Determining which turbines should provide feed forward data to which other turbines is not a trivial task. Wind plants often have complex layouts and at many sites wind direction changes seasonally or over the course of each day. A plant level controller capable of directing turbine-to-turbine communication and adapting to changes in wind direction must be developed.

The simulations in this dissertation did not examine the effects of complex terrain. Many wind plants, such as offshore wind plants or wind plants in the plains of the central united states, have simple flat terrain. Other wind plants, including many of the most productive wind plants in California, have complex hilly terrain. These hills can significantly affect both gusts of wind and turbine wakes. More research is needed to determine how complex terrain affects feed forward derating control.

Some of the simulations in Chapter 4 model atmospheric turbulence, but they do not model turbine wake effects or changes in wind speed fluctuations as they travel downstream. The simulations in Chapter ?? modeled turbine wake effects and changes in wind speed fluctuations as they travel downstream, but did not model atmospheric turbulence, boundary layer shape, or boundary layer stability effects. Research has shown that turbulence has a significant effect on turbine wakes.[83–85] It may also have an effect on the performance of feed forward derating control. More research is needed to determine how feed forward derating control would be affected by a wider range of atmospheric conditions.

The second line of research is to better quantify the benefits that can be achieved with feed forward derating control. One aspect of this research would be to work with a wind farm that has emergency overspeed shutdowns to determine how much energy is lost per year and how much feed forward derating control could increase energy capture. Another aspect of this research would be to better quantify structural damage caused by large gusts and how much repair costs could be reduced by feed forward derating control. This would require calculating the fatigue damage ($D_{fatigue}$) caused by large gusts. As discussed in Appendix D, fatigue damage is an estimate of how much a component has been damaged as a fraction of its life ($D_{fatigue} = 0$ indicates a new part while $D_{fatigue} = 1$ indicates component failure). Fatigue damage calculations could potentially be used to assign a monetary value to gust induced loads. Fatigue damage calculations are similar to Damage Equivalent Load calculations but can only be carried out if the ultimate load of the component is known. The NREL 5MW turbine is a fictional turbine and ultimate component loads do not exist, so they would need to be estimated.

Appendix A

Sample FAST Simulation Input Files

The following pages contain samples of the *primary.fst* and *Aerodyn.ipt* FAST input files. Many FAST simulations were run in this dissertation and some settings varied from one simulation to another, but the parameters shown in these input files illustrate typical settings used in this dissertation. FAST also uses input files to specify the wind flow field, the structural properties of the turbine tower, and the structural and aerodynamic properties of the turbine blades. Those other input files have not been included here because they are either dictated by the properties of the NREL 5-MW turbine, or they are generated by another simulation tool.

A.1 primary.fst

```
----- FAST INPUT FILE -----
NREL 5.0 MW Baseline Wind Turbine for Use in Offshore Analysis.
Properties from Dutch Offshore Wind Energy Converter (DOWEC) 6MW Pre-Design
(10046_009.pdf) and REpower 5M 5MW (5m_uk.pdf); Compatible with FAST v6.0.

----- SIMULATION CONTROL -----
False      Echo          - Echo input data to "echo.out" (flag)
1          ADAMSPrep    - ADAMS preprocessor mode {1: Run FAST, 2: use FAST as a
preprocessor to create an ADAMS model, 3: do both} (switch)
1          AnalMode     - Analysis mode {1: Run a time-marching simulation, 2: create
a periodic linearized model} (switch)
3          NumBl        - Number of blades (-)
600.0      TMax         - Total run time (s)
0.0125    DT            - Integration time step (s)

----- TURBINE CONTROL -----
0          YCMode       - Yaw control mode {0: none, 1: user-defined from routine
UserYawCont, 2: user-defined from Simulink} (switch)
9999.9    TYCon        - Time to enable active yaw control (s) [unused when YCMode=0]
2          PCMode       - Pitch control mode {0: none, 1: user-defined from routine
PitchCntrl, 2: user-defined from Simulink} (switch)
0.0        TPCOn        - Time to enable active pitch control (s) [unused when PCMode=0]
3          VSContrl     - Variable-speed control mode {0: none, 1: simple VS, 2:
user-defined from routine UserVSCont, 3: user-defined from Simulink} (switch)
9999.9    VS_RtGnSp   - Rated generator speed for simple variable-speed generator
control (HSS side) (rpm) [used only when VSContrl=1]
9999.9    VS_RtTq     - Rated generator torque/constant generator torque in Region 3
for simple variable-speed generator control (HSS side) (N-m) [used only when
VSContrl=1]
9999.9    VS_Rgn2K    - Generator torque constant in Region 2 for simple
variable-speed generator control (HSS side) (N-m/rpm^2) [used only when VSContrl=1]
9999.9    VS_SlPc     - Rated generator slip percentage in Region 2 1/2 for simple
variable-speed generator control (%) [used only when VSContrl=1]
2          GenModel     - Generator model {1: simple, 2: Thevenin, 3: user-defined
from routine UserGen} (switch) [used only when VSContrl=0]
True       GenTiStr    - Method to start the generator {T: timed using TimGenOn, F:
generator speed using SpdGenOn} (flag)
True       GenTiStp   - Method to stop the generator {T: timed using TimGenOf, F:
when generator power = 0} (flag)
9999.9    SpdGenOn   - Generator speed to turn on the generator for a startup (HSS
speed) (rpm) [used only when GenTiStr=False]
0.0        TimGenOn   - Time to turn on the generator for a startup (s) [used only
when GenTiStr=True]
9999.9    TimGenOf   - Time to turn off the generator (s) [used only when
GenTiStp=True]
1          HSSBrMode   - HSS brake model {1: simple, 2: user-defined from routine
UserHSSBr} (switch)
9999.9    THSSBrDp   - Time to initiate deployment of the HSS brake (s)
9999.9    TiDynBrk   - Time to initiate deployment of the dynamic generator brake
[CURRENTLY IGNORED] (s)
9999.9    TTpBrDp(1) - Time to initiate deployment of tip brake 1 (s)
9999.9    TTpBrDp(2) - Time to initiate deployment of tip brake 2 (s)
9999.9    TTpBrDp(3) - Time to initiate deployment of tip brake 3 (s) [unused for 2
blades]
9999.9    TBDepISp(1) - Deployment-initiation speed for the tip brake on blade 1
(rpm)
9999.9    TBDepISp(2) - Deployment-initiation speed for the tip brake on blade 2
(rpm)
9999.9    TBDepISp(3) - Deployment-initiation speed for the tip brake on blade 3
(rpm) [unused for 2 blades]
9999.9    TYawMansS  - Time to start override yaw maneuver and end standard yaw
control (s)
```

9999.9 TYawManE - Time at which override yaw maneuver reaches final yaw angle (s)
 0.0 NacYawF - Final yaw angle for yaw maneuvers (degrees)
 9999.9 TPitManS(1) - Time to start override pitch maneuver for blade 1 and end standard pitch control (s)
 9999.9 TPitManS(2) - Time to start override pitch maneuver for blade 2 and end standard pitch control (s)
 9999.9 TPitManS(3) - Time to start override pitch maneuver for blade 3 and end standard pitch control (s) [unused for 2 blades]
 9999.9 TPitManE(1) - Time at which override pitch maneuver for blade 1 reaches final pitch (s)
 9999.9 TPitManE(2) - Time at which override pitch maneuver for blade 2 reaches final pitch (s)
 9999.9 TPitManE(3) - Time at which override pitch maneuver for blade 3 reaches final pitch (s) [unused for 2 blades]
 0.0 BlPitch(1) - Blade 1 initial pitch (degrees)
 0.0 BlPitch(2) - Blade 2 initial pitch (degrees)
 0.0 BlPitch(3) - Blade 3 initial pitch (degrees) [unused for 2 blades]
 0.0 BlPitchF(1) - Blade 1 final pitch for pitch maneuvers (degrees)
 0.0 BlPitchF(2) - Blade 2 final pitch for pitch maneuvers (degrees)
 0.0 BlPitchF(3) - Blade 3 final pitch for pitch maneuvers (degrees) [unused for 2 blades]

----- ENVIRONMENTAL CONDITIONS -----

9.80665 Gravity - Gravitational acceleration (m/s^2)

----- FEATURE FLAGS -----

True	FlapDOF1	- First flapwise blade mode DOF (flag)
True	FlapDOF2	- Second flapwise blade mode DOF (flag)
True	EdgeDOF	- First edgewise blade mode DOF (flag)
False	TeetDOF	- Rotor-teeter DOF (flag) [unused for 3 blades]
True	DrTrDOF	- Drivetrain rotational-flexibility DOF (flag)
True	GenDOF	- Generator DOF (flag)
True	YawDOF	- Yaw DOF (flag)
True	TwFADOF1	- First fore-aft tower bending-mode DOF (flag)
True	TwFADOF2	- Second fore-aft tower bending-mode DOF (flag)
True	TwSSDOF1	- First side-to-side tower bending-mode DOF (flag)
True	TwSSDOF2	- Second side-to-side tower bending-mode DOF (flag)
True	CompAero	- Compute aerodynamic forces (flag)
False	CompNoise	- Compute aerodynamic noise (flag)

----- INITIAL CONDITIONS -----

0.0	OoPDefl	- Initial out-of-plane blade-tip displacement (meters)
0.0	IPDefl	- Initial in-plane blade-tip deflection (meters)
0.0	TeetDefl	- Initial or fixed teeter angle (degrees) [unused for 3 blades]
0.0	Azimuth	- Initial azimuth angle for blade 1 (degrees)
12.1	RotSpeed	- Initial or fixed rotor speed (rpm)
0.0	NacYaw	- Initial or fixed nacelle-yaw angle (degrees)
0.0	TTDspFA	- Initial fore-aft tower-top displacement (meters)
0.0	TTDspSS	- Initial side-to-side tower-top displacement (meters)

----- TURBINE CONFIGURATION -----

63.0	TipRad	- The distance from the rotor apex to the blade tip (meters)
1.5	HubRad	- The distance from the rotor apex to the blade root (meters)
1	PSpnElN	- Number of the innermost blade element which is still part of the pitchable portion of the blade for partial-span pitch control [1 to BldNodes] [CURRENTLY IGNORED] (-)
0.0	UndSling	- Undersling length [distance from teeter pin to the rotor apex] (meters) [unused for 3 blades]
0.0	HubCM	- Distance from rotor apex to hub mass [positive downwind] (meters)
-5.01910	OverHang	- Distance from yaw axis to rotor apex [3 blades] or teeter pin [2 blades] (meters)
1.9	NacCMxn	- Downwind distance from the tower-top to the nacelle CM (meters)
0.0	NacCMyn	- Lateral distance from the tower-top to the nacelle CM (meters)
1.75	NacCMzn	- Vertical distance from the tower-top to the nacelle CM (meters)

```

87.6      TowerHt      - Height of tower above ground level [onshore] or MSL
[offshore] (meters)
1.96256   Twr2Shft    - Vertical distance from the tower-top to the rotor shaft (meters)
0.0        TwrRBHt     - Tower rigid base height (meters)
-5.0       ShftTilt     - Rotor shaft tilt angle (degrees)
0.0        Delta3       - Delta-3 angle for teetering rotors (degrees) [unused for 3 blades]
-2.5       PreCone(1)   - Blade 1 cone angle (degrees)
-2.5       PreCone(2)   - Blade 2 cone angle (degrees)
-2.5       PreCone(3)   - Blade 3 cone angle (degrees) [unused for 2 blades]
0.0        AzimBlUp    - Azimuth value to use for I/O when blade 1 points up (degrees)
----- MASS AND INERTIA -----
0.0        YawBrMass   - Yaw bearing mass (kg)
240.00E3   NacMass     - Nacelle mass (kg)
56.78E3    HubMass     - Hub mass (kg)
0.0        TipMass(1)  - Tip-brake mass, blade 1 (kg)
0.0        TipMass(2)  - Tip-brake mass, blade 2 (kg)
0.0        TipMass(3)  - Tip-brake mass, blade 3 (kg) [unused for 2 blades]
2607.89E3  NacYIner   - Nacelle inertia about yaw axis (kg m^2)
534.116    GenIner     - Generator inertia about HSS (kg m^2)
115.926E3  HubIner    - Hub inertia about rotor axis [3 blades] or teeter axis [2
blades] (kg m^2)
----- DRIVETRAIN -----
100.0      GBoxEff    - Gearbox efficiency (%)
94.4       GenEff      - Generator efficiency [ignored by the Thevenin and
user-defined generator models] (%)
97.0       GBRatio    - Gearbox ratio (-)
False      GBRevers   - Gearbox reversal {T: if rotor and generator rotate in
opposite directions} (flag)
28.1162E3  HSSBrTqF   - Fully deployed HSS-brake torque (N-m)
0.6        HSSBrDT    - Time for HSS-brake to reach full deployment once initiated
(sec) [used only when HSSBrMode=1]
"Dummy"    DynBrkFi   - File containing a mech-gen-torque vs HSS-speed curve for a
dynamic brake [CURRENTLY IGNORED] (quoted string)
867.637E6  DTTorSpr   - Drivetrain torsional spring (N-m/rad)
6.215E6    DTTorDmp   - Drivetrain torsional damper (N-m/(rad/s))
----- SIMPLE INDUCTION GENERATOR -----
9999.9     SIG_SlPc    - Rated generator slip percentage (%) [used only when
VSContrl=0 and GenModel=1]
9999.9     SIG_SySp    - Synchronous (zero-torque) generator speed (rpm) [used only
when VSContrl=0 and GenModel=1]
9999.9     SIG_RtTq    - Rated torque (N-m) [used only when VSContrl=0 and
GenModel=1]
9999.9     SIG_PORT    - Pull-out ratio (Tpullout/Trated) (-) [used only when
VSContrl=0 and GenModel=1]
----- THEVENIN-EQUIVALENT INDUCTION GENERATOR -----
9999.9     TEC_Freq    - Line frequency [50 or 60] (Hz) [used only when VSContrl=0
and GenModel=2]
9998      TEC_NPol    - Number of poles [even integer > 0] (-) [used only when
VSContrl=0 and GenModel=2]
9999.9     TEC_SRes    - Stator resistance (ohms) [used only when VSContrl=0 and
GenModel=2]
9999.9     TEC_RRes    - Rotor resistance (ohms) [used only when VSContrl=0 and
GenModel=2]
9999.9     TEC_VLL    - Line-to-line RMS voltage (volts) [used only when VSContrl=0
and GenModel=2]
9999.9     TEC_SLR    - Stator leakage reactance (ohms) [used only when VSContrl=0
and GenModel=2]
9999.9     TEC_RLR    - Rotor leakage reactance (ohms) [used only when VSContrl=0
and GenModel=2]
9999.9     TEC_MR     - Magnetizing reactance (ohms) [used only when VSContrl=0 and
GenModel=2]

```

```

----- PLATFROM -----
      0      PtfmModel - Platform model {0: none, 1: onshore, 2: fixed bottom
offshore, 3: floating offshore} (switch)
"Dummy"      PtfmFile - Name of file containing platform properties (quoted string)
[unused when PtfmModel=0]
----- TOWER -----
      20      TwrNodes - Number of tower nodes used for analysis (-)
"..\..\..\..\NREL5MW_Properties\NRELOffshrBsline5MW_Tower_Onshore.dat"
TwrFile - Name of file containing tower properties (quoted string)
----- NACELLE-YAW -----
9028.32E6    YawSpr - Nacelle-yaw spring constant (N-m/rad)
  19.16E6    YawDamp - Nacelle-yaw damping constant (N-m/(rad/s))
  0.0   YawNeut - Neutral yaw position--yaw spring force is zero at this yaw (degrees)
----- FURLING -----
False      Furling - Read in additional model properties for furling turbine (flag)
"Dummy"      FurlFile - Name of file containing furling properties (quoted string)
[unused when Furling=False]
----- ROTOR-TEETER -----
      0      TeetMod - Rotor-teeter spring/damper model {0: none, 1: standard, 2:
user-defined from routine UserTeet} (switch) [unused for 3 blades]
      0.0     TeetDmpP - Rotor-teeter damper position (degrees) [used only for 2
blades and when TeetMod=1]
      0.0     TeetDmp - Rotor-teeter damping constant (N-m/(rad/s)) [used only for 2
blades and when TeetMod=1]
      0.0     TeetCDmp - Rotor-teeter rate-independent Coulomb-damping moment (N-m)
[used only for 2 blades and when TeetMod=1]
      0.0     TeetSStP - Rotor-teeter soft-stop position (degrees) [used only for 2
blades and when TeetMod=1]
      0.0     TeetHStP - Rotor-teeter hard-stop position (degrees) [used only for 2
blades and when TeetMod=1]
      0.0     TeetSSSp - Rotor-teeter soft-stop linear-spring constant (N-m/rad)
[used only for 2 blades and when TeetMod=1]
      0.0     TeetHSSp - Rotor-teeter hard-stop linear-spring constant (N-m/rad)
[used only for 2 blades and when TeetMod=1]
----- TIP-BRAKE -----
      0.0     TBDrConN - Tip-brake drag constant during normal operation, Cd*Area (m^2)
      0.0     TBDrConD - Tip-brake drag constant during fully-deployed operation,
Cd*Area (m^2)
      0.0     TpBrDT - Time for tip-brake to reach full deployment once released (sec)
----- BLADE -----
"..\..\..\..\NREL5MW_Properties\NRELOffshrBsline5MW_Blade.dat"
BldFile(1) - Name of file containing properties for blade 1 (quoted string)
"..\..\..\..\NREL5MW_Properties\NRELOffshrBsline5MW_Blade.dat"
BldFile(2) - Name of file containing properties for blade 2 (quoted string)
"..\..\..\..\NREL5MW_Properties\NRELOffshrBsline5MW_Blade.dat"
BldFile(3) - Name of file containing properties for blade 3 (quoted string) [unused
for 2 blades]
----- AERODYN -----
"AeroDyn.ipt"          ADFile - Name of file containing AeroDyn input
parameters (quoted string)
----- NOISE -----
"Dummy"      NoiseFile - Name of file containing aerodynamic noise input parameters
(quoted string) [used only when CompNoise=True]
----- ADAMS -----
"..\..\..\..\NREL5MW Properties\NRELOffshrBsline5MW_ADAMSSpecific.dat"
ADAMSSfile - Name of file containing ADAMS-specific input parameters (quoted string)
[unused when ADAMSPrep=1]
----- LINEARIZATION CONTROL -----
"..\..\..\..\NREL5MW Properties\NRELOffshrBsline5MW_Linear.dat"
LinFile - Name of file containing FAST linearization parameters (quoted string)
[unused when AnalMode=1]

```

```
----- OUTPUT -----
False      SumPrint   - Print summary data to "<RootName>.fsm" (flag)
True       TabDelim   - Generate a tab-delimited tabular output file. (flag)
"ES10.3E2" OutFmt    - Format used for tabular output except time. Resulting field
should be 10 characters. (quoted string) [not checked for validity!]
30.0      TStart     - Time to begin tabular output (s)
1         DecFact    - Decimation factor for tabular output {1: output every time
step} (-)
1.0       SttsTime   - Amount of time between screen status messages (sec)
-3.09528  NcIMUxn   - Downwind distance from the tower-top to the nacelle IMU
(meters)
0.0       NcIMUyn   - Lateral distance from the tower-top to the nacelle IMU
(meters)
2.23336  NcIMUzn   - Vertical distance from the tower-top to the nacelle IMU
(meters)
1.912    ShftGagL   - Distance from rotor apex [3 blades] or teeter pin [2 blades]
to shaft strain gages [positive for upwind rotors] (meters)
0         NTwGages   - Number of tower nodes that have strain gages for output [0
to 9] (-)
TwrGagNd  - List of tower nodes that have strain gages [1 to TwrNodes]
(-) [unused if NTwGages=0]
3         NB1Gages   - Number of blade nodes that have strain gages for output [0
to 9] (-)
5,9,13   BldGagNd   - List of blade nodes that have strain gages [1 to BldNodes]
(-) [unused if NB1Gages=0]
OutList   - The next line(s) contains a list of output parameters. See
OutList.txt for a listing of available output channels, (-)
"WindVxi,WindVyi,WindVzi"   - Wind-speed components (m/s)
"Azimuth, RotSpeed"        - Rotor azimuth angle (deg) and angular speed (rpm)
"TSR"                      - Tip speed ratio (-)
"GenSpeed, GenAccel"       - Generator speed (rpm), and generator acceleration(deg/s^2)
"LSShftTq"                 - LSS torque (kNm)
"GenTq, GenPwr"            - Generator torque (kNm) and power (kW)
"TipDxb1, TipDyb1"         - Blade 1 flapwise and edgewise tip deflection (m)
"TipDxb2, TipDyb2"         - Blade 2 flapwise and edgewise tip deflection (m)
"TipDxb3, TipDyb3"         - Blade 3 flapwise and edgewise tip deflection (m)
"TipDxc1, TipDyc1"         - Blade 1 tip out-plane and in-plane defl (m)
"TipDxc2, TipDyc2"         - Blade 2 tip out-plane and in-plane defl (m)
"TipDxc3, TipDyc3"         - Blade 3 tip out-plane and in-plane defl (m)
"BldPitch1"                - Blade 1 pitch angle (deg)
"BldPitch2"                - Blade 2 pitch angle (deg)
"BldPitch3"                - Blade 3 pitch angle (deg)
"YawBrTDxp"                - Tower-top/yaw bearing fore-aft (translational) deflection (m)
"YawBrTDyp"                - Tower-top/yaw bearing side-to-side (translational) deflection (m)
"YawBrTDzp"                - Tower-top/yaw bearing axial (translational) deflection (m)
"RootMxb1, RootMyb1, RootMzb1" - Blade 1 root moments (kNm)
"RootMxb2, RootMyb2, RootMzb2" - Blade 2 root moments (kNm)
"RootMxb3, RootMyb3, RootMzb3" - Blade 3 root moments (kNm)
"TwrbSMyt"                  - Tower fore-aft moment.
END of FAST input file
-----
```

A.2 Aerodyn.ipt

```

NREL 5.0 MW offshore baseline aerodynamic input properties; Compatible with AeroDyn
v12.58.

SI           SysUnits      - System of units for used for input and output [must be SI
for FAST] (unquoted string)
BEDDOES     StallMod      - Dynamic stall included [BEDDOES or STEADY] (unquoted string)
USE_CM       UseCm        - Use aerodynamic pitching moment model? [USE_CM or NO_CM]
(unquoted string)
EQUIL        InfModel     - Inflow model [DYNIN or EQUIL] (unquoted string)
SWIRL        IndModel     - Induction-factor model [NONE or WAKE or SWIRL] (unquoted string)
0.005        AToler       - Induction-factor tolerance (convergence criteria) (-)
PRANDtl     TLModel      - Tip-loss model (EQUIL only) [PRANDtl, GTECH, or NONE]
(unquoted string)
PRANDtl     HLModel      - Hub-loss model (EQUIL only) [PRANDtl or NONE] (unquoted string)
"16ms_wGust.hh" "TurbSim.bts"      WindFile - Name of file containing wind data
(quoted string)
90.0         HH           - Wind reference (hub) height
[TowerHt+Twr2Shft+OverHang*SIN(ShftTilt)] (m)
0.0          TwrShad      - Tower-shadow velocity deficit (-)
9999.9       ShadHWid     - Tower-shadow half width (m)
9999.9       T_Shad_Refpt - Tower-shadow reference point (m)
1.225        AirDens      - Air density (kg/m^3)
1.464E-5    KinVisc      - Kinematic air viscosity [CURRENTLY IGNORED] (m^2/sec)
0.02479     DTAero       - Time interval for aerodynamic calculations (sec)
8            NumFoil      - Number of airfoil files (-)
"..\..\..\..\NREL5MW_Properties\AeroData\Cylinder1.dat"                   FoilNm
- Names of the airfoil files [NumFoil lines] (quoted strings)
"..\..\..\..\NREL5MW_Properties\AeroData\Cylinder2.dat"
"..\..\..\..\NREL5MW_Properties\AeroData\DU40_A17.dat"
"..\..\..\..\NREL5MW_Properties\AeroData\DU35_A17.dat"
"..\..\..\..\NREL5MW_Properties\AeroData\DU30_A17.dat"
"..\..\..\..\NREL5MW_Properties\AeroData\DU25_A17.dat"
"..\..\..\..\NREL5MW_Properties\AeroData\DU21_A17.dat"
"..\..\..\..\NREL5MW_Properties\AeroData\NACA64_A17.dat"
17           BldNodes     - Number of blade nodes used for analysis (-)
RNodes      AeroTwst     DRNodes   Chord   NFoil   PrnElm
2.8667     13.308      2.7333   3.542   1       NOPRINT
5.6000     13.308      2.7333   3.854   1       NOPRINT
8.3333     13.308      2.7333   4.167   2       NOPRINT
11.7500    13.308      4.1000   4.557   3       NOPRINT
15.8500    11.480      4.1000   4.652   4       NOPRINT
19.9500    10.162      4.1000   4.458   4       NOPRINT
24.0500    9.011       4.1000   4.249   5       NOPRINT
28.1500    7.795       4.1000   4.007   6       NOPRINT
32.2500    6.544       4.1000   3.748   6       NOPRINT
36.3500    5.361       4.1000   3.502   7       NOPRINT
40.4500    4.188       4.1000   3.256   7       NOPRINT
44.5500    3.125       4.1000   3.010   8       NOPRINT
48.6500    2.319       4.1000   2.764   8       NOPRINT
52.7500    1.526       4.1000   2.518   8       NOPRINT
56.1667    0.863       2.7333   2.313   8       NOPRINT
58.9000    0.370       2.7333   2.086   8       NOPRINT
61.6333    0.106       2.7333   1.419   8       NOPRINT

```

Appendix B

WT_Perf

WT_Perf uses blade-element momentum (BEM) theory to predict the aerodynamic performance of a turbine rotor for a given set of operating conditions. WT_Perf was written by National Wind Technology Center (NWTC) staff at the National Renewable Energy Laboratory (NREL) and is descended from the PROP simulation tool developed at Oregon State University. WT_Perf is documented in [?] and [?]. However, NREL has stopped supporting WT_Perf since the work in Chapter 2 was performed and [?] is no longer readily available.

The aerodynamics of a wind turbine rotor are complex and three dimensional. For a typical wind turbine blade the chord length, twist angle, and airfoil shape change along the length of the blade. In addition, the blade follows a circular trajectory so the end of the blade does not see the same apparent wind as the base of the blade. Simulating the entire rotor would be difficult and computationally expensive. However, a more computationally efficient estimate of rotor performance can be obtained by breaking this large complex 3-D problem into many smaller 2-D simulations. In essence, blade-element momentum theory (BEM) models each wind turbine blade as a collection of many blade elements of width δr subject to 2-D airflow (Figure B.1). If δr is small enough the chord length, twist angle, airfoil shape, and apparent wind can be considered constant along the width of each blade element. The aerodynamic forces on the rotor can then be determined by summing the forces on all of the blade elements and applying a series of correction factors to

account for 3-D flow effects such as tip vortices, root vortices, and dynamic stall.

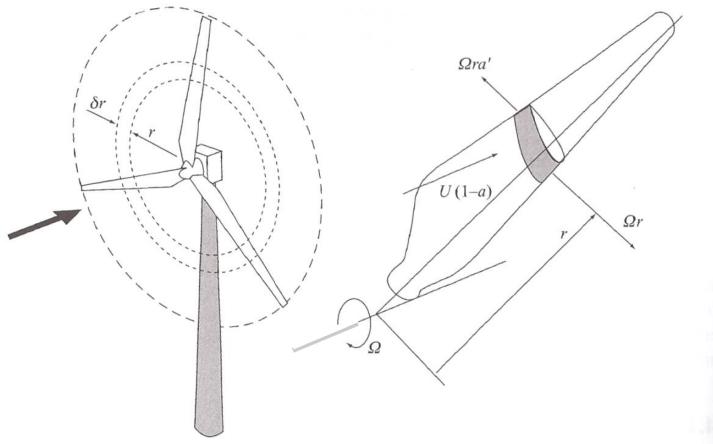


Figure B.1. Blade element of width $\hat{l}tr$ at radius r .[9]

Section ?? contains a typical WT_Perf input file. The *Model Configuration* and *Algorithm Configuration* sections define the BEM methods to be used. *Model Configuration* specifies how many elements each blade will be divided into (Numsect) and specifies the tolerances used to determine adequate convergence of the iterative BEM calculations. *Algorithm Configuration* controls a set of optional correction factors that can be applied to BEM calculations. Some of these correction factors (such as TipLoss, HubLoss, and Swirl) approximate 3-D flow effects that would not otherwise be captured by BEM. The *Turbine Data* and *Aerodynamic Data* sections define the turbine rotor being modeled. *Turbine Data* quantifies many physical properties of the rotor, including the number of blades (NumBlade) as well as the physical dimensions, location, and orientation of the rotor (RotorRad, HubRad, PreCone, Tilt, Yaw, HubHt). The remainder of *Turbine Data* defines each turbine blade by describing a series of blade segments. Each segment has a defined radial length (RElm), axial twist (Twist), chord length (Chord), and airfoil shape (AFfile). *Aerodynamic Data* contains information required for aerodynamic calculations. The first three lines (Rho, KinVisc, and Shear) contain physical properties of the incoming wind. The remainder of *Aerodynamic Data* defines a set of .dat files that contain detailed descriptions of airfoil shapes that make up the turbine blade.

Unlike FAST, which is discussed in Appendix A, WT_Perf cannot model turbine controls and cannot model dynamic turbine behavior, such as the effect of changing wind speed, changing rotor speed, or changing blade pitch. However, for a given set of operating conditions (blade pitch, rotor speed, and wind speed) WT_Perf can calculate the rotor power, power coefficient (C_p), shaft torque, flap bending moment, and/or rotor thrust. Calculating rotor performance based on specified operating conditions instead of dynamic turbine behavior makes WT_Perf very computationally inexpensive. WT_Perf can calculate rotor performance metrics for thousands of operating conditions in a few minutes.

The *Combined-Case Analysis* and *Parametric Analysis* sections of the WT_Perf input file tell WT_Perf which rotor performance metrics to output and which operating conditions to perform BEM calculations for. The input file shown below instructs WT_Perf to calculate the power coefficient (C_p) for all integer blade pitches between -5° and 5° all integer rotor speeds between 1 RPM and 14 RPM, and all integer wind speeds between 3 m/s and 16 m/s. This causes WT_Perf to calculate C_p at 2,156 sets of operating conditions (11 blade pitches \times 14 rotor speeds \times 14 wind speeds).

B.1 Example WT_Perf input file.

Appendix C

Simulink Controllers

This appendix documents the simulink controllers developed in chapters 3 and 4. The subsections below include images of the controllers as well as brief descriptions of the Simulink blocks and their functions.

C.1 Baseline NREL 5-MW Feedback Control.

The figures in this section illustrate the Simulink model used to simulate the Upwind Turbine in chapters 3 and 4. The Simulink systems shown below model an NREL 5-MW turbine with the control system described in the NREL 5-MW specification[2]. Figure C.1 shows the top level Simulink system. The green box contains a version of FAST that has been specially compiled to interact with Simulink control systems. At each time step this version of FAST takes generator torque, generator power, yaw position, and blade pitch as inputs. The FAST outputs are defined by the input file *primary.fst*. A typical set of outputs can be seen in Appendix A. All of the yellow boxes shown in the figures below record simulation data and write it to output files at each time step. At each time step, generator speed is extracted from the FAST output and passed through a low pass filter (the small blue box in Figure C.1) to remove high frequency oscillations. This filtered generator speed is then fed to the pitch and torque controllers (the two largest blue boxes), which determine the blade pitch, generator torque, and generator power inputs to FAST. The Yaw position input is specified by the Yaw controller (the grey box), but

in this case the yaw controller simply specifies a constant value of 0°.

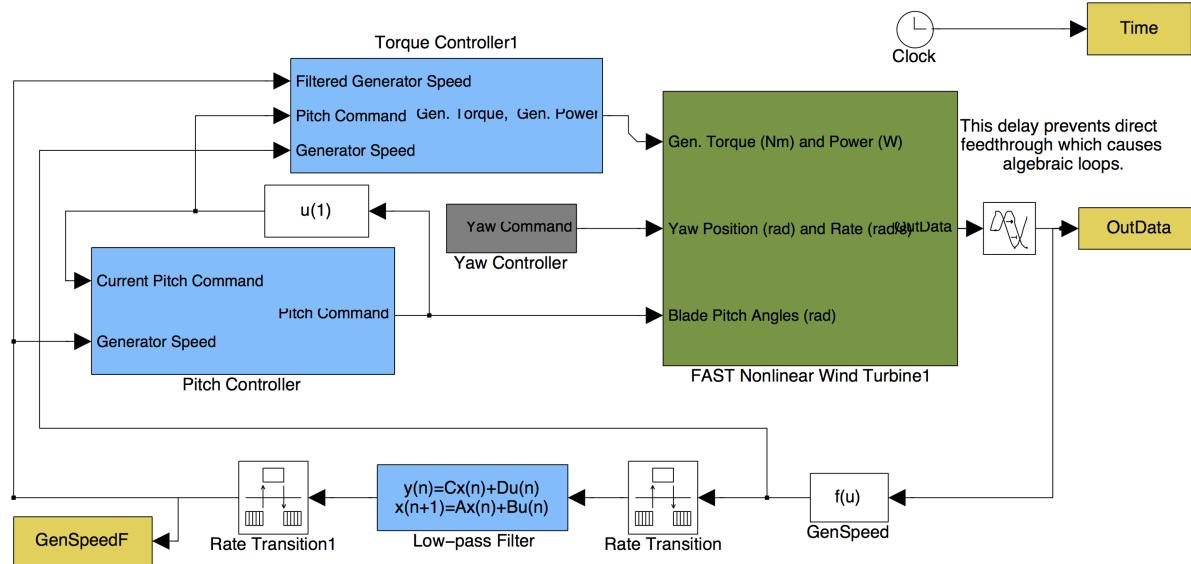


Figure C.1. Top level Simulink system for model of NREL 5-MW with baseline (no feed forward) control.

Figure C.2 shows the Pitch Controller subsystem. It takes filtered generator speed and current blade pitch angle as inputs. The NREL 5-MW turbine uses a nonlinear Proportional and Integral (PI) controller for pitch control. In this control scheme the control gains are a function of blade pitch. The Gain Schedule subsystem uses the current blade pitch to adjust the proportional (K_p) and Integral (K_i) control gains using the logic shown in Figure C.3. The filtered generator speed is compared to the rated generator speed to determine the generator speed error. The Total Pitch Command subsystem takes the generator speed error and adjusted control gains as inputs then applies PI control using the logic shown in Figure C.4 to produce a blade pitch command. That blade pitch command is fed through the Blade Pitch Actuator block, which is a 2nd order model of the blade pitch actuator dynamics with a natural frequency of 1 Hz and a damping ratio of 0.7, to produce the pitch command that will be input to FAST. The Blade Pitch Actuator block is included here because FAST does not model blade pitch actuator dynamics.

Figure C.5 shows the Torque Controller subsystem. It takes filtered generator speed, current blade pitch angle, and unfiltered generator speed as inputs. The

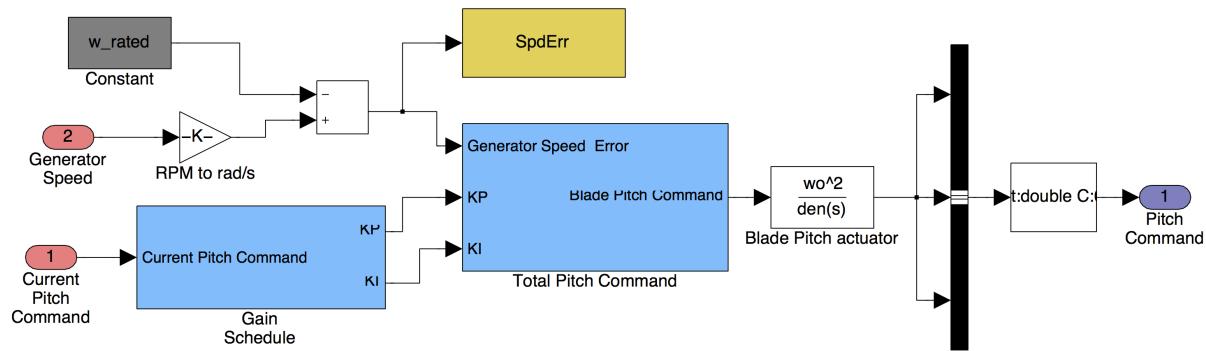


Figure C.2. Pitch Controller subsystem for Simulink model of NREL 5-MW with baseline control.

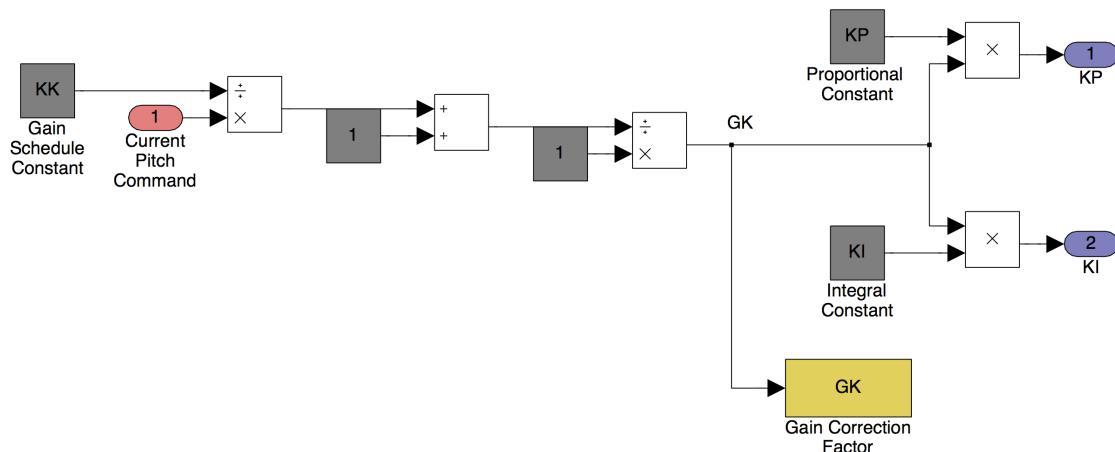


Figure C.3. Gain Schedule subsystem for Simulink model of NREL 5-MW with baseline control.

embedded Matlab function GenTorque uses blade pitch angle to determine if the turbine is in region 3 control. It then calculates the appropriate generator torque for the current filtered generator speed and control region. The full text of GenTorque is shown in Figure C.6. The generator torque command is passed through value and rate limiters to produce the generator torque command that will be input to FAST. To calculate generator power, the unfiltered generator speed is multiplied by the generator torque command then adjusted to account for the efficiency of the generator.

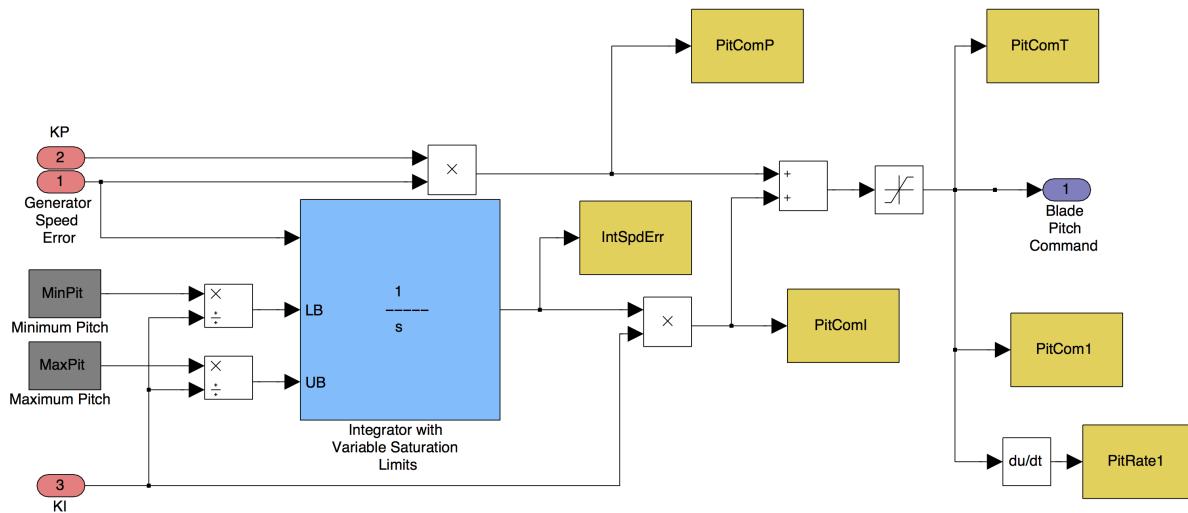


Figure C.4. Total Pitch Command subsystem for Simulink model of NREL 5-MW with baseline control.

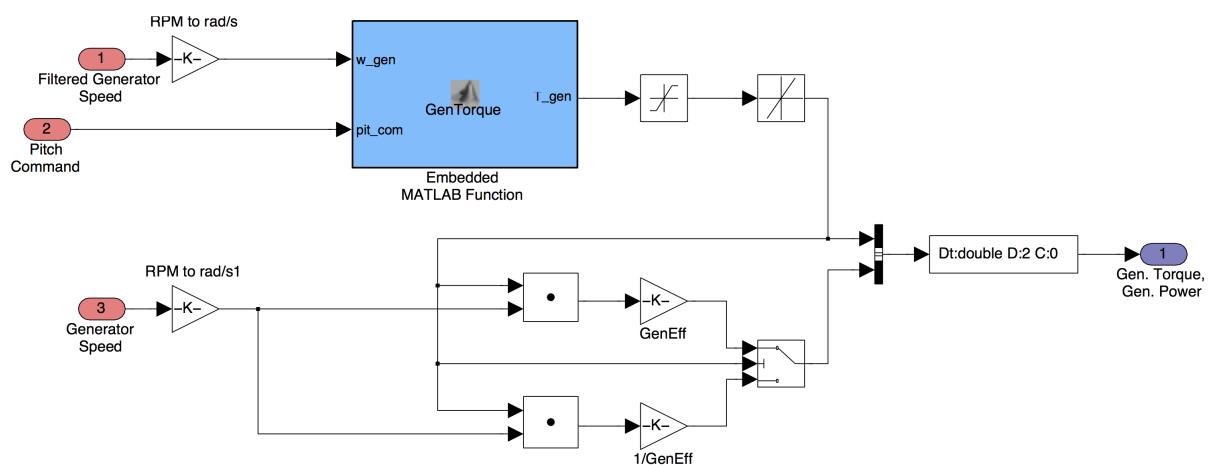


Figure C.5. Torque Controller subsystem for Simulink model of NREL 5-MW with baseline control.

```

function T_gen = GenTorque(w_gen,pit_com)

%Set up all of the torque controller constants
w_cutin = 70.16224; %rad/s
w_rated = 121.6805; %rad/s
k_gen = 2.332287; %(N-m)/(rad/s)^2
P_max = 5296610.0; %Watts

m15 = (1.3*w_cutin)^2*k_gen/(1.3*w_cutin-w_cutin); %Region 1.5 Slope
m25 = (P_max/w_rated)/(w_rated-w_rated/1.1); %Region 2.5 slope
w_syst = w_rated/1.1; %speed where the turbine operates as a synchronous induction
generator
w_TrGen = (m25-sqrt(m25^2-4*k_gen*w_syst*m25))/(2*k_gen); %Transition speed from region
2 to region 2.5
minPit3 = 1*pi/180; %Minimum pitch angle command that puts the turbine into region 3

if w_gen < w_cutin
    T_gen = 0;
elseif w_gen < 1.3*w_cutin
    T_gen = m15*(w_gen-w_cutin);
elseif w_gen < w_TrGen
    T_gen = k_gen*w_gen^2;
elseif w_gen < w_rated
    T_gen = m25*(w_gen-w_syst);
else
    T_gen = P_max/w_gen;
end

if pit_com >= minPit3
    T_gen = P_max/w_gen;
end

```

Figure C.6. Embedded Matlab function GenTorque for baseline control.

C.2 NREL 5-MW Turbine With Feed Forward Optimal Pitch Control.

The figures in this section illustrate the Simulink model used to simulate A Downwind turbine with feed forward optimum pitch control. This is the feed forward controller analyzed in Chapter 3. Figure C.7 shows the top level Simulink system. Much of the figure looks similar to Figure C.1, which illustrates the baseline (or upwind) turbine. In fact, the FAST block and all of the components of the baseline closed loop controller are present in Figure C.7. However, there is a new component. In the lower left corner there is a Feed Forward Controller subsystem that sends a FF Pitch Command signal to the Pitch Controller subsystem.

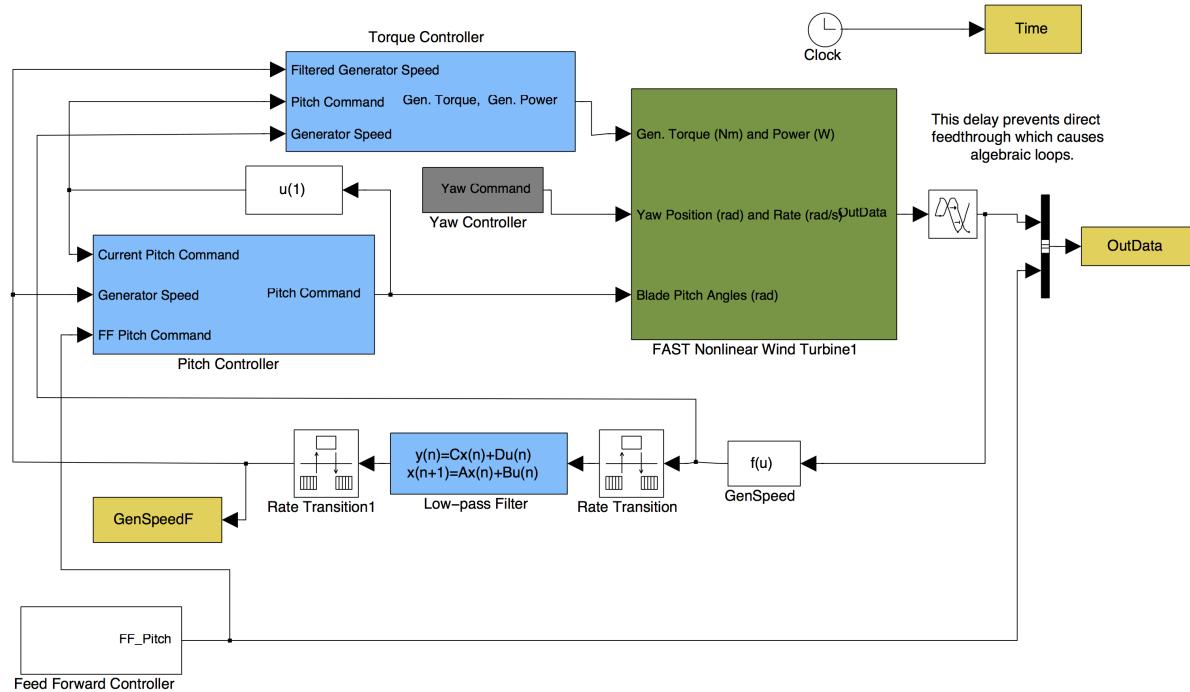


Figure C.7. Top level Simulink system for model of NREL 5-MW with feed forward optimal pitch control.

Figure C.8 shows the Feed Forward Controller subsystem. This subsystem is simple because most of the calculations are done prior to the simulation by a set of Matlab files. The FF Dynamic Wind Speed Lookup block contains a time history of the dynamic feed forward velocity (U_{ff}), which has been generated based on the

output data from the Upwind Turbine simulation. Dynamic feed forward velocity is described in Section 3.3.3 and Equation 3.3. To save space, the Matlab files that are used to process Upwind Turbine output data and create the FF Dynamic Wind Speed Lookup table have not been included in this appendix. Those Matlab files are fairly long and several sets of files were required for the different sets of assumptions and operating conditions used in Chapter 3. The FF Pitch Lookup Table contains a table relating dynamic feed forward velocity to the corresponding optimum pitch angle that will keep the turbine at a 12.1 RPM rotor speed. The relationship between dynamic feed forward velocity and optimum pitch angle is shown in Figure 3.11.

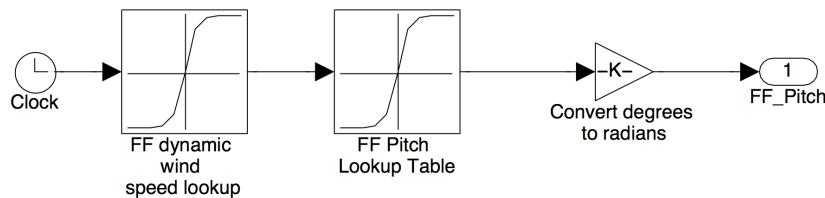


Figure C.8. Feed Forward Controller subsystem for Simulink model of NREL 5-MW with feed forward optimal pitch control.

The Pitch Controller subsystem is shown in Figure C.9. It has been modified to accept the FF Pitch Command. The entire closed loop pitch control system is still present in this subsystem. The feed forward pitch is added to the Blade Pitch Command coming from the Total Pitch Command subsystem. This combined pitch command, from the closed loop controller and the feed forward controller, is then fed through the blade pitch actuator model before being passed to FAST.

The Gain Schedule, Total Pitch Command, Torque Controller, and GenTorque subsystems are identical to the subsystems in the baseline NREL 5-MW controller, so they are not shown here. The form and function of those subsystems are described in Section C.1.

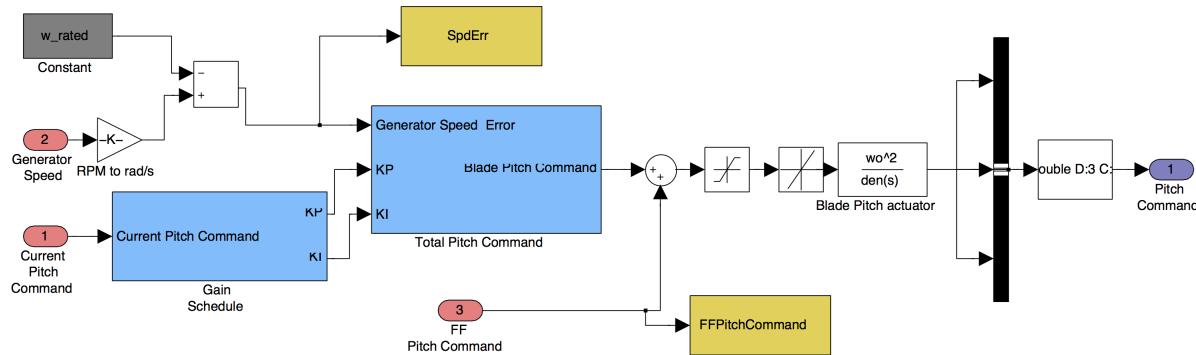


Figure C.9. Pitch Controller subsystem for Simulink model of NREL 5-MW with feed forward optimal pitch control.

C.3 NREL 5-MW Turbine With Feed Forward Derating Control.

The figures in this section illustrate the Simulink model used to simulate a Downwind Turbine with feed forward optimum pitch control. This is the feed forward controller analyzed in Chapter 4. Figure C.10 shows the top level Simulink system. Much of the figure looks similar to Figure C.1, which illustrates the baseline (or upwind) turbine. The FAST block and all of the components of the baseline closed loop controller are present in Figure C.10. However, in the lower left corner there is a Feed Forward Controller subsystem that sends FF_pwrFactor and FF_minPitch commands to the Pitch Controller and Torque Controller subsystems. FF_pwrFactor stands for Feed Forward Power Factor which is a measure of the turbine's current rated capacity as a fraction of full rated capacity. For example, a turbine in normal operation would have a power factor of 1. A turbine that has been derated by 20% would have a power factor of 0.8. FF_minPitch stands for Feed Forward Minimum Pitch. The minimum pitch is a function of the power factor, but it was easiest to implement it as a separate signal here. The minimum pitch is used to derate a turbine operating in region 2 control.

Figure C.11 shows the Feed Forward Controller subsystem. The FF Derate Factor Lookup Table block contains a time history of the derate factor, which has been

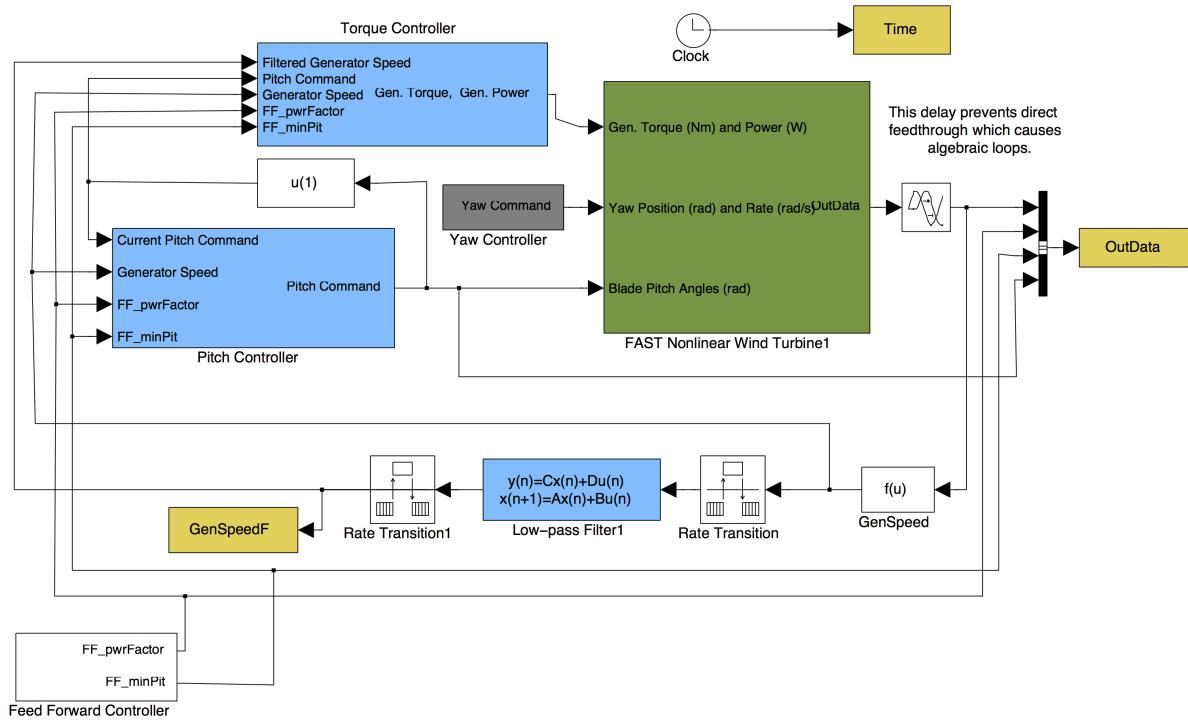


Figure C.10. Top level Simulink system for model of NREL 5-MW with feed forward derating control.

generated based on the output data from the Upwind Turbine simulation. The de-rate factor is the amount of derating requested by the feed forward controller (as a fraction of full turbine capacity). Most of the time the derate factor is 0, but it is increased when the turbine is derated. The derate factor signal is fed through the Low Pass Filter block to prevent abrupt changes in turbine rating (as discussed in Section 4.4). Subtracting the derate factor from 1 turns it into a power factor signal. The Min Pitch Lookup Table block determines the minimum pitch corresponding to the current power factor. Finally, the FF_pwrFactor and FF_minPitch signals are then passed out of the controller.

The Pitch Controller subsystem is shown in Figure C.12. It has been modified to accept the FF_pwrFactor and FF_minPitch commands. The rated generator speed (GenSpRef) is multiplied by the power factor (FF_pwrFactor). When the turbine is derated, and the power factor is less than 1, this reduces the generator speed that the turbine pitch controller is attempting to track. The minimum pitch (FF_minPitch)

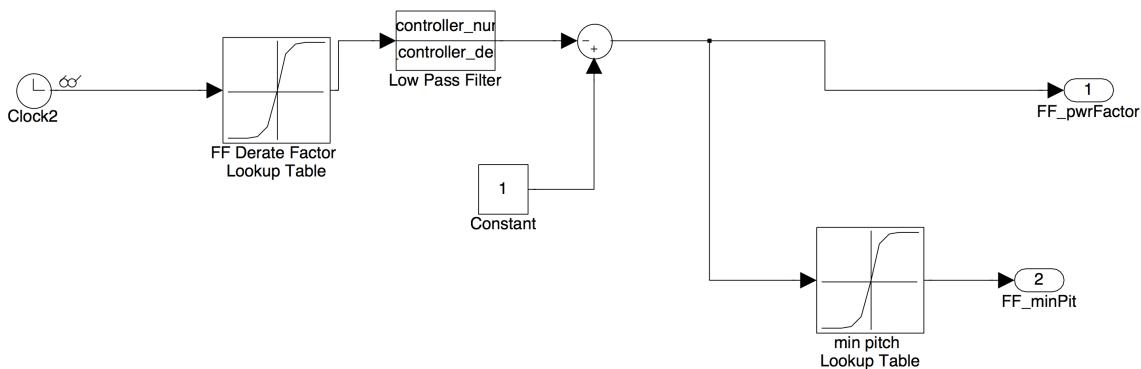


Figure C.11. Feed Forward Controller subsystem for Simulink model of NREL 5-MW with feed forward derating control.

is passed to the Total Pitch Command block. The Gain Schedule block is identical to the PI Control Gain Adjustment block described by Section C.1 and Figure C.3. It will not be described here.

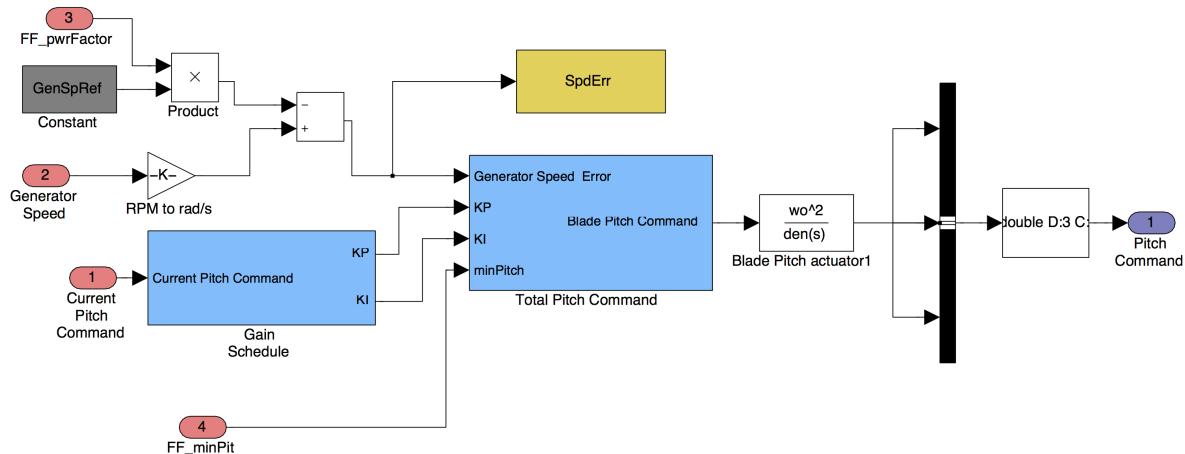


Figure C.12. Pitch Controller subsystem for Simulink model of NREL 5-MW with feed forward derating control.

The Total Pitch Command subsystem is shown in Figure C.13. This subsystem applies nonlinear PI control, much like it does in the baseline controller (Figure C.4). However, this subsystem uses the minimum pitch signal (minPitch) as the lower bounds for blade pitch control. The baseline controller uses 0° as the lower bounds.

Figure C.14 shows the Torque Controller subsystem. Much of this subsystem

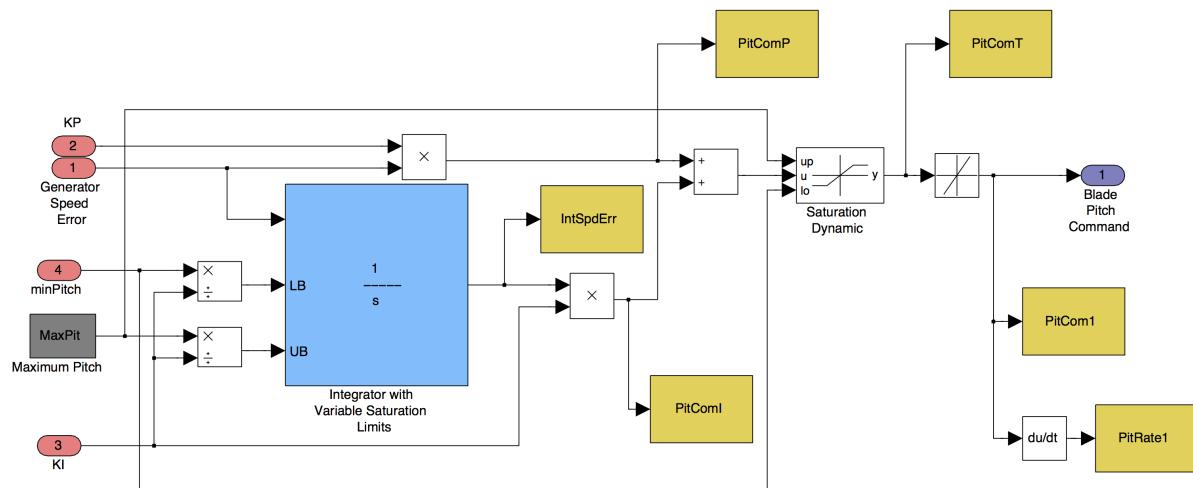


Figure C.13. Total Pitch Command subsystem for Simulink model of NREL 5-MW with feed forward derating control.

remains unchanged from the baseline controller. The power factor (FF_pwrFactor) and minimum blade pitch (FF_minPitch) commands are passed to the embedded Matlab function GenTorque along with the filtered generator speed and current pitch command. The full text of GenTorque is shown in Figure C.15. GenTorque uses the derate factor to adjust the shapes of the NREL 5-MW torque-speed curves when the turbine is derated. It then compares the current pitch command to the minimum pitch to determine if the turbine is in region 3 control. Finally, it calculates the appropriate generator torque for the current filtered generator speed, control region, and derate factor.

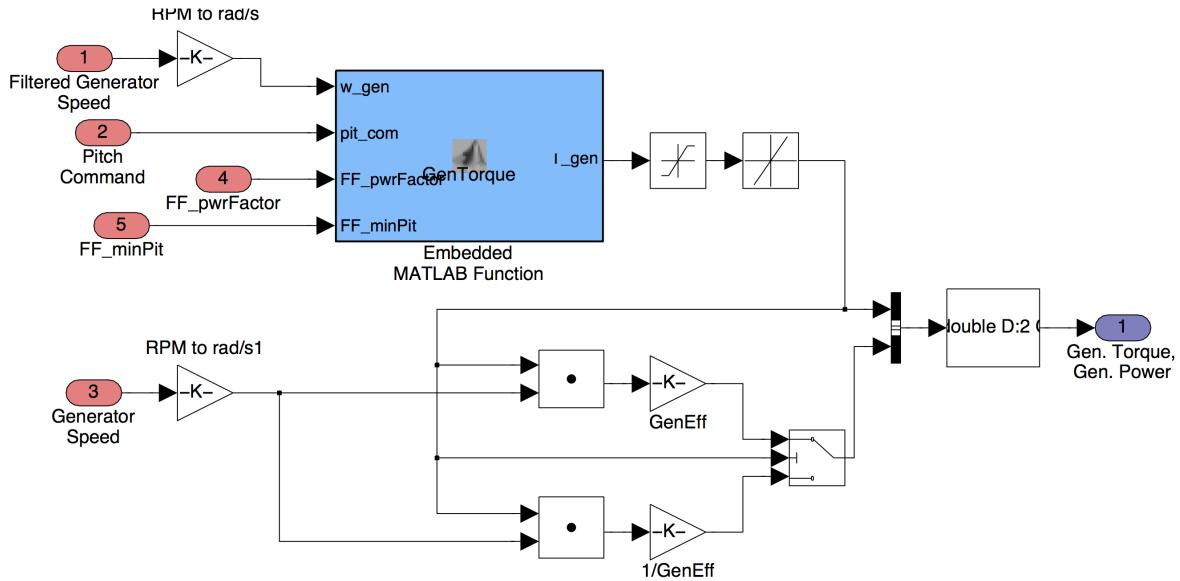


Figure C.14. Torque Controller subsystem for Simulink model of NREL 5-MW with feed forward derating control.

```

function T_gen = GenTorque(w_gen,pit_com,FF_pwrFactor,FF_minPitch)

%Set up all of the torque controller constants
w_cutin = 70.16224; %rad/s
w_rated = 121.6805; %rad/s
k_gen = 2.332287; %(N-m)/(rad/s)^2
P_max = 5296610.0; %Watts

%% Adjust parameters based on FF_pwrFactor
w_rated = 121.6805*FF_pwrFactor; %rad/s
k_gen = 2.332287/(FF_pwrFactor^2); %(N-m)/(rad/s)^2
P_max = 5296610.0*FF_pwrFactor; %Watts

m15 = (1.3*w_cutin)^2*k_gen/(1.3*w_cutin-w_cutin); %Region 1.5 Slope
m25 = (P_max/w_rated)/(w_rated-w_rated/1.1); %Region 2.5 slope
w_systp = w_rated/1.1; %speed where the turbine operates as a synchronous induction
generator
w_TrGen = (m25-sqrt(m25^2-4*k_gen*w_systp*m25))/(2*k_gen); %Transition speed from region
2 to region 2.5
minPitch3 = 1*pi/180; %Minimum pitch angle command that puts the turbine into region 3
minPitch3 = FF_minPitch + 1*pi/180;

if w_gen < w_cutin
    T_gen = 0;
elseif w_gen < 1.3*w_cutin
    T_gen = m15*(w_gen-w_cutin);
elseif w_gen < w_TrGen
    T_gen = k_gen*w_gen^2;
elseif w_gen < w_rated
    T_gen = m25*(w_gen-w_systp);
else
    T_gen = P_max/w_gen;
end

if pit_com >= minPitch3
    T_gen = P_max/w_gen;
end

```

Figure C.15. Embedded Matlab function GenTorque for feed forward derating control.

Appendix D

Damage Equivalent Load (DEL) Calculation

Damage equivalent loads are calculated by MLife, A fatigue analysis tool developed by NREL [86, 87]. MLife automates the damage analysis techniques outlined in Annex G of IEC 61400-1 [36]. The load history of a component, as shown in Figure D.1 , can be thought of as a series of load cycles. Each of those load cycles has some mean load and cyclic loading range. If damage accumulates linearly and independently for each cycle, as proposed by Palmgren [88] and Miner[89], then the total damage from all cycles will be given by:

$$D_{fatigue} = \sum_i \frac{1}{N_{fatigue,i}} \quad (D.1)$$

where $N_{fatigue,i}$ denotes the number of cycles to failure for the i^{th} loading cycle. For example, if the i^{th} loading cycle would have to be repeated 10,000 times to drive a new component ($D_{fatigue} = 0$) to failure ($D_{fatigue} = 1$) then $N_{fatigue,i} = 10,000$ and the damage accumulated due to the i^{th} loading cycle would be 0.0001. For a given loading cycle $N_{fatigue,i}$ is estimated by:

$$N_{fatigue,i} = \left(\frac{L_{ultimate} - L_{mean}}{\frac{1}{2}L_{range,i}} \right)^m \quad (D.2)$$

where $L_{ultimate}$ is the ultimate design load of the component, L_{mean} is the mean

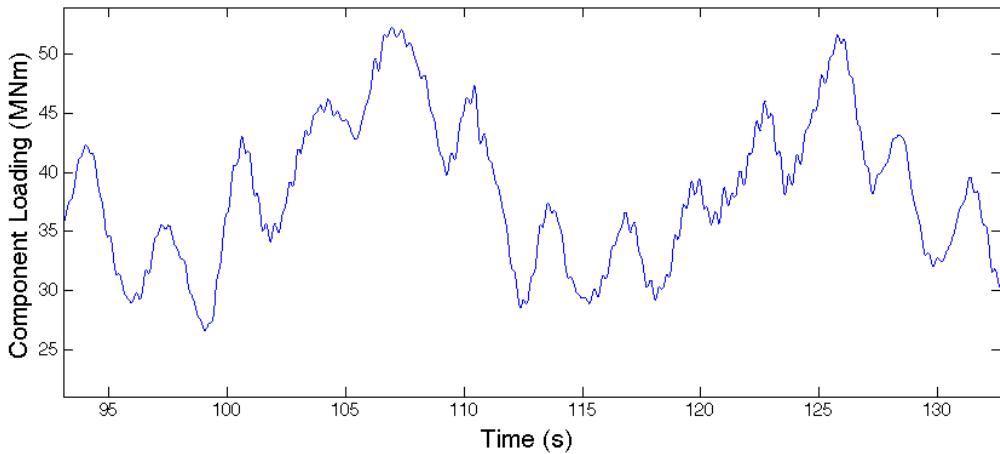


Figure D.1. Example of component loading.

load over the component's load history, $L_{range,i}$ is the i^{th} cycles range, and m is the Wöhler exponent. It can be seen in equation D.2 that larger loads cause more damage. Increasing the load cycle range or the mean load lead to smaller values of $N_{fatigue,i}$, which means more damage per cycle.

The Wöhler exponent is a property of the component under consideration, but it is highly dependent on the material the component is made from. For damage equivalent load calculations in this dissertation m was assumed to be 10 for steel components (tower) and 4 for fiberglass components (blades). These are the values of m that Stewart, Lackner, Haid, Matha, Jonkman, and Robertson used to perform fatigue analysis of the NREL 5 MW turbine in [90].

Because the NREL 5MW turbine is a fictional turbine ultimate loads ($L_{ultimate}$) are not available for its components. This presents a problem when attempting to calculate accurate fatigue damage ($D_{fatigue}$) estimates. However, this problem can be circumvented if we use the component load history to calculate a damage equivalent load instead of using it to calculate fatigue damage.

The damage equivalent load (DEL) is a constant amplitude fatigue load that occurs at a fixed frequency and produces the same damage as the actual load history of the component. The damage caused by the DEL is:

$$D_{fatigue} = \frac{n_{equivalent}}{N_{equivalent}} \quad (\text{D.3})$$

$$N_{equivalent} = \left(\frac{L_{ultimate} - L_{mean}}{\frac{1}{2}DEL} \right) \quad (\text{D.4})$$

where $N_{equivalent}$ is the number of cycles to failure for the damage equivalent load, and $n_{equivalent}$ is the number of damage equivalent loading cycles the component is subjected to. $n_{equivalent}$ is chosen by the analyst. All DEL calculations in this dissertation have used $n_{equivalent} = 1$. In other words, the complex load history of the components under consideration have been converted to single loading cycles that cause equivalent damage. Setting $n_{equivalent} = 1$ and combining equations D.1 and D.3 yields:

$$\sum \frac{1}{N_{fatigue,i}} = \frac{1}{N_{equivalent}} \quad (\text{D.5})$$

Substituting in equations D.2 and D.4 then solving for DEL gives:

$$DEL = \left(\sum_i (L_{range,i})^m \right)^{\frac{1}{m}} \quad (\text{D.6})$$

We see from equation D.6 that the ultimate load of the component is not required to calculate the damage equivalent load. the DEL does quantify the amount of damage done to a component, but it is a useful tool for comparing the relative damage done by two time histories. A higher DEL indicates more damage. In this dissertation DEL is used to compare various control strategies to see if they reduce or increase the damage experienced by the turbine.

A few notes on the Goodman correction: The DEL calculations carried out in this dissertation, as outlined above, do not include the Goodman correction. When the Goodman correction is used, equation D.2 becomes:

$$N_{fatigue,i} = \left(\frac{L_{ultimate} - L_{mean,i}}{\frac{1}{2}L_{range,i}} \right)^m \quad (\text{D.7})$$

where $L_{mean,i}$ is the mean load for the i^{th} loading cycle, not the mean load of the load history. Using the Goodman correction leads to more accurate estimates of fatigue damage. However, when the goodman correction is used, the equation for DEL is not independent of the ultimate load:

$$DEL = \left(\sum_i \left(L_{range,i} \left(\frac{L_{ultimate} - L_{mean}}{L_{ultimate} - L_{mean,i}} \right) \right)^m \right)^{\frac{1}{m}} \quad (\text{D.8})$$

Since accurate estimates of $L_{ultimate}$ were not available for NREL 5 MW components, the Goodman correction was not used for DEL calculations in this dissertation.

Appendix E

SOWFA Implementation of Feed Forward Derating Control

This appendix documents the controller implementation discussed in Section ???. The code shown below is a FORTRAN implementation of the Simulink based controller documented in Chapter 4 and Appendix C.3 with a few additions. These subroutines, modules, and code changes are added to the SOWFA/FAST source code, which enables SOWFA to model feed forward derating control. This controller is used for simulations documented in Chapter ??.

E.1 Subroutine PitchCtrl()

The following subroutine is added to UserSubs.f90:

```

SUBROUTINE PitchCntrl ( BlPitch, ElecPwr, HSS_Spd, GBRatio, TwrAccel, NumBl, &
ZTime, DT, DirRoot, BlPitchCom_out )

! This pitch control routine was written by Eric Anderson of UC Davis. It is based
! on the baseline NREL 5MW pitch control scheme outlined in the NREL 5MW
! specifications with the addition the derating scheme outlined in Eric's
! dissertation. Derating is achieved by using subroutine ControlParameters() to scale
! relevant control parameters.

USE precision
USE EAControl ! contains variables: TimeDRStart,
TimeDREnd, DerateFactor, TEmShutdown, maxOverspeed, EmergencyShutdown, GenSpeedF,
PC_RefSpd, PC_MinPit, VS_Rgn2_K, and VS_RtPwr. See EAControl module in FAST_Mods.f90
for variable descriptions.

IMPLICIT NONE

! Passed variables:

INTEGER(4), INTENT(IN ) :: NumBl           ! Number of blades, (-).
REAL(ReKi), INTENT(IN ) :: BlPitch   (NumBl)    ! Current values of the blade
pitch angles, rad.

```

```

REAL(ReKi), INTENT(IN )      :: DT          ! Integration time step, sec.
REAL(ReKi), INTENT(IN )      :: ElecPwr    ! Electrical power, watts.
REAL(ReKi), INTENT(IN )      :: GBRatio    ! Gearbox ratio, (-).
REAL(ReKi), INTENT(IN )      :: HSS_Spd    ! HSS speed, rad/s.
REAL(ReKi), INTENT(OUT)      :: BlPitchCom_out(NumBl) ! Commanded blade pitch angles
(demand pitch angles), rad.
REAL(ReKi), INTENT(IN )      :: TwrAccel   ! Tower Acceleration, m/s^2.
REAL(ReKi), INTENT(IN )      :: ZTime       ! Current simulation time,
sec.

CHARACTER(1024), INTENT(IN ) :: DirRoot      ! The name of the root file
including the full path to the current working directory. This may be useful if you
want this routine to write a permanent record of what it does to be stored with the
simulation results: the results should be stored in a file whose name (including path)
is generated by appending any suitable extension to DirRoot.

! Local Variables:

REAL(ReKi)                  :: ElapTime     ! Elapsed time since the last call to
the controller, sec.
REAL(ReKi)                  :: GK           ! Current value of the gain
correction factor, used in the gain scheduling law of the pitch controller, (-).
REAL(ReKi), SAVE             :: IntSpdErr   ! Current integral of speed error
w.r.t. time, rad.
REAL(ReKi), SAVE             :: LastTimePC  ! Last time the pitch controller was
called, sec.
REAL(ReKi), PARAMETER        :: OnePlusEps = 1.0 + EPSILON(OnePlusEps) ! The number
slightly greater than unity in single precision.
REAL(ReKi), PARAMETER        :: PC_DT = 0.00125      ! Communication interval for
pitch controller, sec.
REAL(ReKi), PARAMETER        :: PC_KI = 0.008068634 ! Integral gain for
pitch controller at rated pitch (zero), (-).
REAL(ReKi), PARAMETER        :: PC_KK = 0.1099965    ! Pitch angle were the
the derivative of the aerodynamic power w.r.t. pitch has increased by a factor of two
relative to the derivative at rated pitch (zero), rad.
REAL(ReKi), PARAMETER        :: PC_KP = 0.01882681   ! Proportional gain for
pitch controller at rated pitch (zero), sec.
REAL(ReKi), PARAMETER        :: PC_MaxPit = 1.570796   ! Maximum pitch
setting in pitch controller, rad.
REAL(ReKi), PARAMETER        :: PC_MaxRat = 0.1396263  ! Maximum pitch
rate (in absolute value) in pitch controller, rad/s.
REAL(ReKi), SAVE             :: PitCom (3)    ! Commanded pitch of each blade the
last time the controller was called, rad.
REAL(ReKi)                  :: PitComI      ! Integral term of command pitch,
rad.
REAL(ReKi)                  :: PitComP      ! Proportional term of command
pitch, rad.
REAL(ReKi)                  :: PitComT      ! Total command pitch based on the
sum of the proportional and integral terms, rad.
REAL(ReKi)                  :: PitRate(3)    ! Pitch rates of each blade based on
the current pitch angles and current pitch command, rad/s.
REAL(ReKi)                  :: SpdErr        ! Current speed error, rad/s.
LOGICAL                      :: Initialize = .TRUE. ! A status flag set by the
simulation as follows: 0 if this is the first call, 1 for all subsequent time steps,
-1 if this is the final call at the end of the simulation.
INTEGER(ReKi)                :: K            ! Loops through blades.
INTEGER(ReKi)                :: pitCoun = 1  !Used for debug output

=====
!Initialize variables:

```

```

IF ( Initialize ) THEN ! .TRUE. if we're on the first call to the subroutine
  WRITE(*,*) 'Running with pitch control programmed by Eric Anderson' // &
             'in subroutine PitchCntrl(), which can be found in UserSubs.f90 '
  Initialize = .FALSE.
! Initialize the SAVED variables:
  PitCom      = BlPitch                                ! This will ensure that the variable
speed controller picks the correct control region and the pitch controller picks the
correct gain on the first call. If pitchCtrl() is called before UsrVSCtrl()
initializing it here will work, if not I need to do something else.
  GK          = 1.0/( 1.0 + PitCom(1)/PC_KK ) ! This will ensure that the pitch
angle is unchanged if the initial SpdErr is zero
  IntSpdErr  = PitCom(1)/( GK*PC_KI )                  ! This will ensure that the pitch
angle is unchanged if the initial SpdErr is zero
  LastTimePC = ZTime - PC_DT                          ! This will ensure that the pitch
controller is called on the first pass
ENDIF

=====
! Pitch control:

! Compute the elapsed time since the last call to the controller:
ElapTime = ZTime - LastTimePC

! Only perform the control calculations if the elapsed time is greater than
! or equal to the communication interval of the pitch controller:
! NOTE: Time is scaled by OnePlusEps to ensure that the controller is called
! at every time step when PC_DT = DT, even in the presence of
! numerical precision errors.

IF ( ( ZTime*OnePlusEps - LastTimePC ) >= PC_DT ) THEN

  CALL updateControlParameters( HSS_Spd, ZTime )

  IF ( EmergencyShutdown ) THEN
    PitComT = 3.1415926535/2
  ELSE
    ! Compute the gain scheduling correction factor based on the previously
    ! commanded pitch angle for blade 1:
    GK = 1.0/( 1.0 + PitCom(1)/PC_KK )

    ! Compute the current speed error and its integral w.r.t. time;
    SpdErr     = GenSpeedF - PC_RefSpd                ! Current
speed error
    IntSpdErr = IntSpdErr + SpdErr*ElapTime           ! Current
integral of speed error w.r.t. time

    ! saturate the integral term using the pitch angle limits:
    IntSpdErr = MIN( MAX(IntSpdErr,PC_MinPit/(GK*PC_KI)), &
                     PC_MaxPit/( GK*PC_KI) ) ! Saturate the integral term using
the pitch angle limits, converted to integral speed error limits

    ! Compute the pitch commands associated with the proportional and integral
gains:
    PitComP   = GK*PC_KP*   SpdErr                   !
Proportional term
    PitComI   = GK*PC_KI*IntSpdErr                  ! Integral
term (saturated)

    ! Superimpose the individual commands to get the total pitch command;
    PitComT   = PitComP + PitComI                    ! Overall

```

```

command (unsaturated)

    !  saturate the overall command using the pitch angle limits:
    PitComT = MIN( MAX( PitComT, PC_MinPit ), PC_MaxPit )           ! Saturate
the overall command using the pitch angle limits
ENDIF

DO K = 1,NumBl ! Loop through all blades
    PitRate(K) = ( PitComT - BlPitch(K) )/ElapTime                  ! Pitch rate
of blade K (unsaturated)
    PitRate(K) = MIN( MAX( PitRate(K), -PC_MaxRat ), PC_MaxRat )   ! Saturate the
pitch rate of blade K using its maximum absolute value
    PitCom (K) = BlPitch(K) + PitRate(K)*ElapTime                   ! Saturate the
overall command of blade K using the pitch rate limit
    ENDDO               ! K - all blades

! Reset the value of LastTimePC to the current value:
LastTimePC = ZTime

IF ( controlDebug ) THEN
    WRITE(*,*) 'Time=',ZTime,'pitCount=',pitCount,'HSS_Spd=',HSS_Spd,&
                'GenSpeedF=',GenSpeedF,'PitCom=',PitCom(1),'PitComP= ',&
                PitComP,'PitComI= ',PitComI,'PitComT= ',PitComT
    pitCount=pitCount+1
ENDIF

ENDIF

BlPitchCom_out = PitCom                                ! Pass the most recent blade pitch command
out of the subroutine

RETURN
END SUBROUTINE PitchCntrl

```

E.2 Subroutine UserVSCntrl()

The following subroutine is added to UserSubs.f90:

```

SUBROUTINE UserVSCntrl ( HSS_Spd, GBRatio, NumBl, ZTime, DT, GenEff, DelGenTrq,
DirRoot, GenTrq, ElecPwr )

    ! This torque control routine was written by Eric Anderson of UC Davis. The torque
control scheme
    ! is based on the controller described in the NREL 5MW specifications, with the
addition of a
    ! derating scheme described in Eric's dissertation. Derating is achieved by using
subroutine
    ! ControlParameters() to scale relevant control parameters. WARNING: This routine
does not include the
    ! logic required to run linearizations.

USE                                     precision
USE                                     Output
USE                                     EAControl ! contains variables: TimeDRStart,
TimeDREnd, DerateFactor, TEmShutdown, maxOverspeed, EmergencyShutdown, GenSpeedF,
PC_RefSpd, PC_MinPit, VS_Rgn2_K, and VS_RtPwr. See EAControl module in FAST_Mods.f90
for variable descriptions.

```

IMPLICIT	NONE
----------	------

```

! Passed Variables:

INTEGER(4), INTENT(IN )      :: NumBl      ! Number of blades, (-).
REAL(ReKi), INTENT(IN )      :: DelGenTrq ! Perturbation in generator torque used
during FAST linearization (zero otherwise), N-m.
REAL(ReKi), INTENT(IN )      :: DT          ! Integration time step, sec.
REAL(ReKi), INTENT(OUT)      :: ElecPwr    ! Electrical power (account for losses),
watts.
REAL(ReKi), INTENT(IN )      :: GBRatio    ! Gearbox ratio, (-).
REAL(ReKi), INTENT(IN )      :: GenEff     ! Generator efficiency, (-).
REAL(ReKi), INTENT(OUT)      :: GenTrq     ! Electrical generator torque, N-m.
REAL(ReKi), INTENT(IN )      :: HSS_Spd    ! HSS speed, rad/s.
REAL(ReKi), INTENT(IN )      :: ZTime       ! Current simulation time, sec.
CHARACTER(1024), INTENT(IN ) :: DirRoot    ! The name of the root file including the
full path to the current working directory. This may be useful if you want this
routine to write a permanent record of what it does to be stored with the simulation
results: the results should be stored in a file whose name (including path) is
generated by appending any suitable extension to DirRoot.

! Local Variables:
REAL(ReKi)                  :: ElapTime    ! Elapsed time since the last call to
the controller, sec.
REAL(ReKi), SAVE             :: LastGenTrq ! Commanded electrical generator
torque the last time the controller was called, N-m.
REAL(ReKi), SAVE             :: LastTimeVS  ! Last time the torque controller was
called, sec.
REAL(ReKi), PARAMETER        :: OnePlusEps = 1.0 + EPSILON(OnePlusEps) ! The
number slightly greater than unity in single precision.
REAL(ReKi)                  :: TrqRate     ! Torque rate based on the current
and last torque commands, N-m/s.
REAL(ReKi), PARAMETER        :: VS_Rgn3MP = 0.01745329 ! Minimum pitch angle at
which the torque is computed as if we are in region 3 regardless of the generator
speed, rad. -- chosen to be 1.0 degree above PC_MinPit
REAL(ReKi), PARAMETER        :: VS_CtInSp = 70.16224   ! Transitional generator
speed (HSS side) between regions 1 and 1 1/2, rad/s.
REAL(ReKi), PARAMETER        :: VS_DT      = 0.00125   ! Communication interval for
torque controller, sec.
REAL(ReKi), PARAMETER        :: VS_MaxRat = 15000.0   ! Maximum torque rate
(in absolute value) in torque controller, N-m/s.
REAL(ReKi), PARAMETER        :: VS_MaxTq  = 47402.91   ! Maximum generator
torque in Region 3 (HSS side), N-m. -- chosen to be 10% above VS_RtTq = 43.09355kNm
REAL(ReKi), PARAMETER        :: VS_Rgn2Sp = 91.21091   ! Transitional
generator speed (HSS side) between regions 1 1/2 and 2, rad/s.

REAL(ReKi)                  :: VS_RtGnSp   ! Rated generator speed (HSS side),
rad/s. -- cequal to 99% of PC_RefSpd
REAL(ReKi)                  :: VS_Slope15  ! Torque/speed slope of region 1 1/2
cut-in torque ramp , N-m/(rad/s).
REAL(ReKi)                  :: VS_Slope25  ! Torque/speed slope of region 2 1/2
induction generator, N-m/(rad/s).
REAL(ReKi), PARAMETER        :: VS_SlPc = 10.0   ! Rated generator slip percentage
in Region 2 1/2, %.
REAL(ReKi)                  :: VS_SySp     ! Synchronous speed of region 2 1/2
induction generator, rad/s.
REAL(ReKi)                  :: VS_TrGnSp   ! Transitional generator speed (HSS
side) between regions 2 and 2 1/2, rad/s.
REAL(ReKi)                  :: BlPitchCom ! Commanded blade pitch angle for

```

```

blade 1 (demand pitch angle), rad.
REAL(ReKi), PARAMETER :: R2D = 57.295780 ! Factor to convert radians to
degrees.

LOGICAL, SAVE :: Initialize1 = .TRUE. !Flag used to initialize some
saved variables on the first call to this subroutine
LOGICAL, SAVE :: Initialize2 = .TRUE. !Flag used to initialize some
saved variables on the first call to this subroutine

INTEGER(4) :: TqCount = 1 ! Counter to see how many time subroutine is called

!=====
! Initialize saved variables on first call to subroutine
IF ( Initialize1 ) THEN
  WRITE(*,*) 'Running with torque control programmed by Eric Anderson // &
               'in subroutine UserVSCont(), which can be found in UserSubs.f90 '
  Initialize1 = .FALSE.
  ! NOTE: LastGenTrq, though SAVED, is initialized below for simplicity, not
here.
  LastTimeVS = ZTime - VS_DT ! This will ensure that the torque controller is
called on the first pass
ENDIF
!=====
! Variable-speed torque control:

! Compute the elapsed time since the last call to the controller:
ElapTime = ZTime - LastTimeVS

! Only perform the control calculations if the elapsed time is greater than
! or equal to the communication interval of the torque controller:
! NOTE: Time is scaled by OnePlusEps to ensure that the controller is called
! at every time step when VS_DT = DT, even in the presence of
! numerical precision errors.

IF ( ( ZTime*OnePlusEps - LastTimeVS ) >= VS_DT ) THEN

  ! Get up to date control parameters
  CALL updateControlParameters( HSS_Spd, ZTime )

  IF ( EmergencyShutdown ) THEN
    GenTrq = 0
  ELSE

    ! Determine some torque control parameters not specified directly:
    VS_RtGnSp = 0.99*PC_RefSpd
    VS_SySp = VS_RtGnSp/( 1.0 + 0.01*VS_SlPc )
    VS_Slope15 = ( VS_Rgn2_K*VS_Rgn2Sp*VS_Rgn2Sp )/( VS_Rgn2Sp - VS_CtInSp )
    VS_Slope25 = ( VS_RtPwr/VS_RtGnSp )/( VS_RtGnSp - VS_SySp )
    VS_TrGnSp = ( VS_Slope25 - SQRT( VS_Slope25*( VS_Slope25 - &
      4.0*VS_Rgn2_K*VS_SySp ) ) )/( 2.0*VS_Rgn2_K ) !Transition speed
from region 2 to region 2.5

    BlPitchCom = AllOuts(PtchPMzc1)/R2D
    ! Compute the generator torque, which depends on which region we are in:

    IF ( ( GenSpeedF >= VS_RtGnSp ) .OR. ( BlPitchCom >= (VS_Rgn3MP + &
      PC_MinPit ) ) ) THEN ! We are in region 3 - power is
constant
      GenTrq = VS_RtPwr/GenSpeedF
    ELSEIF ( GenSpeedF <= VS_CtInSp ) THEN ! We are in region 1 - torque is

```

```

zero
    GenTrq = 0.0
    ELSEIF ( GenSpeedF < VS_Rgn2Sp ) THEN ! We are in region 1 1/2 - linear
ramp in torque from zero to optimal
        GenTrq = VS_Slope15*( GenSpeedF - VS_CtInSp )
    ELSEIF ( GenSpeedF < VS_TrGnSp ) THEN ! We are in region 2 -
optimal torque is proportional to the square of the generator speed
        GenTrq = VS_Rgn2_K*GenSpeedF*GenSpeedF
    ELSE
        GenTrq = VS_Slope25*( GenSpeedF - VS_SySp ) ! We are in region 2 1/2
    - simple induction generator transition region
        GenTrq = VS_Slope25*( GenSpeedF - VS_SySp )
    ENDIF

    ! Saturate the commanded torque using the maximum torque limit:
    GenTrq = MIN( GenTrq , VS_MaxTq ) ! Saturate the
command using the maximum torque limit

    !Initialize saved variables on first call to subroutine
    IF ( Initialize2 ) THEN
        Initialize2 = .FALSE.
        LastGenTrq = GenTrq ! Initialize the value of LastGenTrq
on the first pass only
    ENDIF
    ENDIF

    ! Saturate the commanded torque using the torque rate limit:
    TrqRate = ( GenTrq - LastGenTrq )/ElapTime ! Torque rate
(unsaturated)
    TrqRate = MIN( MAX( TrqRate, -VS_MaxRat ), VS_MaxRat ) ! Saturate the torque
rate using its maximum absolute value
    GenTrq = LastGenTrq + TrqRate*ElapTime ! Saturate the command
using the torque rate limit

    ! Reset the values of LastTimeVS and LastGenTrq to the current values:
    LastTimeVS = ZTime
    LastGenTrq = GenTrq

    IF ( controlDebug ) THEN
        WRITE(*,*,'Time=',ZTime,'TqCount=',TqCount,&
        'GenTrq=',GenTrq,'HSS_Spd=',HSS_Spd,'GenSpeedF=',GenSpeedF
        TqCount = TqCount+1
    ENDIF

    ELSE
        GenTrq = LastGenTrq
    ENDIF

    IF ( GenTrq > 0.0 ) THEN
        ElecPwr = GenTrq*HSS_Spd*GenEff
    ELSE
        ElecPwr = GenTrq*HSS_Spd/GenEff
    ENDIF

    RETURN
END SUBROUTINE UserVSCont

```

E.3 Subroutine updateControlParameters()

The following subroutine is added to UserSubs.f90.

```
SUBROUTINE updateControlParameters( HSS_Spd, ZTime )

USE precision
USE EAControl ! contains variables: TimeDRStart,
TimeDREnd, DerateFactor, TEmShutdown, maxOverspeed, EmergencyShutdown, GenSpeedF,
PC_RefSpd, PC_MinPit, VS_Rgn2_K, and VS_RtPwr. See EAControl module in FAST_Mods.f90
for variable descriptions.

IMPLICIT NONE

! Passed Variables:
REAL(ReKi), INTENT(IN)      :: HSS_Spd ! Current HSS (generator) speed, rad/s.
REAL(ReKi), INTENT(IN )     :: ZTime    ! Current simulation time, sec.

! Local variables storing baseline control parameters (FROM the controller
described in the NREL 5MW specifications)
REAL(ReKi), PARAMETER       :: PC_MinPit_baseline      = 0.0          ! Minimum
pitch setting in NREL 5MW baseline pitch controller, rad.
REAL(ReKi), PARAMETER       :: PC_RefSpd_baseline      = 122.9096 ! Desired
(reference) HSS speed for NREL 5MW baseline pitch controller, rad/s.
REAL(ReKi), PARAMETER       :: VS_Rgn2K_baseline      = 2.332287 ! Generator
torque constant in Region 2 (HSS side) for NREL 5MW baseline controller,
N-m/(rad/s)^2.
REAL(ReKi), PARAMETER       :: VS_RtGnSp_baseline      = 121.6805 ! Rated
generator speed (HSS side) for NREL 5MW baseline controller, rad/s. -- chosen to be
99% of PC_RefSpd
REAL(ReKi), PARAMETER       :: VS_RtPwr_baseline      = 5296610.0 ! Rated
generator generator power in Region 3 for NREL 5MW baseline controller, Watts. --
chosen to be 5MW divided by the electrical generator efficiency of 94.ReKi%
REAL(ReKi), PARAMETER       :: VS_Rgn2Sp_baseline      = 91.21091   !
Transitional generator speed (HSS side) between regions 1 1/2 and 2 for NREL 5MW
baseline controller, rad/s.
REAL(ReKi), PARAMETER       :: VS_CtInSp_baseline      = 70.16224   !
Transitional generator speed (HSS side) between regions 1 and 1 1/2 for NREL 5MW
baseline controller, rad/s.

!Local variables used to filter generator speed
REAL(ReKi)                  :: Alpha                   ! Current coefficient in the
recursive, single-pole, low-pass filter, (-).
REAL(ReKi), SAVE             :: LastTime               ! Last time this subroutine
was called, sec.
REAL(ReKi), PARAMETER        :: CornerFreq      = 1.570796 ! Corner frequency
(-3dB point) in the recursive, single-pole, low-pass filter, rad/s. -- chosen to be
1/4 the blade edgewise natural frequency ( 1/4 of approx. 1Hz = 0.25Hz =
1.570796rad/s)

!Local variables used for derate calculations
REAL(ReKi), PARAMETER        :: pDR = 0.2      !- poles of the second order
derate input filter.
REAL(ReKi), SAVE              :: FF_pwrFactor = 1.0 ! The derate factor. A
fraction of 1, where 1 is not derated.
REAL(ReKi), PARAMETER, DIMENSION (17) :: DRPitchArray = (/ 0.1178, 0.1091, &
0.1004, 0.0916, 0.0829, 0.0742, 0.0654, 0.0611, 0.0524, 0.0436, 0.0393, 0.0349, &
0.0305, 0.0262, 0.0218, 0.0131, 0.0 /) !Array of minimum pitch values, (radians)
REAL(ReKi), PARAMETER, DIMENSION (17) :: DRArray      = (/ 0.5789, 0.6184, 0.6579, &
0.6974, 0.7368, 0.7763, 0.8158, 0.8289, 0.8684, 0.9079, 0.9211, 0.9342, 0.9474, &
0.9605, 0.9737, 0.9868, 1.0000 /) !Array of derate values corresponding to the minimum
pitch array
INTEGER(4)                   :: interpCounter !This is an index used by the
minimum pitch interpolation DO loop.
```

```

LOGICAL, SAVE :: Initialize = .TRUE. !Flag used to
initialize some saved variables on the first call to this subroutine

!=====
! Initialize saved variables on first call to subroutine
IF ( Initialize ) THEN
  WRITE(*,*) 'First call to subroutine updateControlParameters(), '// &
'programmed by Eric Anderson. Subroutine can be found in UserSubs.f90 '
  Initialize = .FALSE.
  GenSpeedF = HSS_Spd ! This will ensure that generator speed filter
will use the initial value of the generator speed on the first pass
  LastTime = ZTime ! This will ensure that generator speed filter
will use the initial value of the generator speed on the first pass
ENDIF
!=====
! Filter the HSS (generator) speed measurement:
! NOTE: This is a very simple recursive, single-pole, low-pass filter with
!       exponential smoothing.
! Update the coefficient in the recursive formula based on the elapsed time
! since the last call to the controller:
Alpha = EXP( ( LastTime - ZTime )*CornerFreq )
! Apply the filter:
GenSpeedF = ( 1.0 - Alpha )*HSS_Spd + Alpha*GenSpeedF
!=====

!Derate Calculations
IF ( ZTime >= TimeDREnd) THEN
  !return turbine to normal operation
  FF_pwrFactor = 1.0 - DerateFactor + DerateFactor*(1.0 - pDR*(ZTime - &
  TimeDREnd)*EXP(-pDR*(ZTime - TimeDREnd)) - EXP(-pDR*(ZTime - TimeDREnd)))
ELSEIF ( ZTime >= TimeDRStart) THEN
  !Derate turbine
  FF_pwrFactor = 1.0 - DerateFactor*(1.0 - pDR*(ZTime - TimeDRStart)*&
  EXP(-pDR*(ZTime - TimeDRStart)) - EXP(-pDR*(ZTime - TimeDRStart)))
ENDIF
!=====

! Set pitch control parameters
PC_RefSpd = PC_RefSpd_baseline*FF_pwrFactor
DO interpCounter = 2, size(DRPitchArray)
  IF ( (FF_pwrFactor .GT. DRArray(interpCounter-1) ) .AND. &
  (FF_pwrFactor .LT. DRArray(interpCounter) ) ) THEN
    PC_MinPit = DRPitchArray(interpCounter-1) + ( DRPitchArray&
    (interpCounter) - DRPitchArray(interpCounter-1) )*( FF_pwrFactor - &
  DRArray(interpCounter-1) )/( DRArray(interpCounter) - DRArray(interpCounter-1) )
    WRITE(*,*) 'PowerFactor = ',FF_pwrFactor, ' PC_MinPit = ',PC_MinPit
  ENDIF
ENDDO
!=====

! Set torque control parameters
VS_Rgn2_K = VS_Rgn2K_baseline/(FF_pwrFactor**2) ! Region 2 torque constant
VS_RtPwr = VS_RtPwr_baseline*FF_pwrFactor ! Rated power
!=====

! Check to see if emergency shutdown should be initiated
IF ((ZTime > 30.0) .AND. (EmergencyShutdown .EQV. .FALSE.)) THEN ! If simulation
has run long enough to pass the initial transient behavior and an emergency shutdown
hasn't been requested yet.
  IF ( ( ZTime > TEMShutdown ) .OR. ( (100*(HSS_Spd-PC_RefSpd_baseline)&
  /PC_RefSpd_baseline) .GE. maxOverspeed) ) THEN ! Should an emergency
shutdown be requested now?
    EmergencyShutdown = .TRUE.
    WRITE(*,*) 'Emergency shutdown requested at T = ',ZTime, &
    ' Overspeed = ',(100*(HSS_Spd-PC_RefSpd_baseline)/PC_RefSpd_baseline)
  ENDIF
ENDIF

```

```

        ENDIF
    ENDIF
!=====
! Reset the value of LastTime to the current value:
    LastTime = ZTime

    RETURN
END SUBROUTINE updateControlParameters

```

E.4 Subroutine UserHSSBr()

The following subroutine is added to UserSubs.f90.

```

SUBROUTINE UserHSSBr ( GenTrq, ElecPwr, HSS_Spd, GBRatio, NumBl, ZTime, DT, DirRoot, &
                      HSSBrFrac )

    ! wrote a very simple High speed brake torque control routine. If the high speed
    ! shaft comes to a complete stop the high speed shaft brake will be engaged at
    ! full torque. Otherwise the brake will not be used. The intention is to keep the
    ! rotor stationary after an emergency shutdown, but not use the brake durring the
    ! emergency shutdown or in normal operation. (Eric Anderson)

USE                                     Precision
USE                                     EAControl      ! contains LOGICAL variable
EmergencyShutdown.

IMPLICIT                               NONE

    ! Passed Variables:

INTEGER(4), INTENT(IN)      :: NumBl      ! Number of blades, (-).
REAL(ReKi), INTENT(IN)      :: DT          ! Integration time step, sec.
REAL(ReKi), INTENT(IN)      :: ElecPwr    ! Electrical power (account for losses), watts.
REAL(ReKi), INTENT(IN)      :: GBRatio    ! Gearbox ratio, (-).
REAL(ReKi), INTENT(IN)      :: GenTrq     ! Electrical generator torque, N-m.
REAL(ReKi), INTENT(IN)      :: HSS_Spd    ! HSS speed, rad/s.
REAL(ReKi), INTENT(OUT)     :: HSSBrFrac ! Fraction of full braking torque: 0 (off)
<= HSSBrFrac <= 1 (full), (-).
REAL(ReKi), INTENT(IN)      :: ZTime       ! Current simulation time, sec.
CHARACTER(1024), INTENT(IN) :: DirRoot    ! The name of the root file including the
full path to the current working directory.
REAL(ReKi), SAVE             :: brakeStartTime ! Time when HHS Brake is initiated.
LOGICAL, SAVE                :: brakeOff = .TRUE.
REAL(ReKi), PARAMETER       :: HSSBrDT = 0.6 ! Time it takes for HSS brake to reach
full deployment once deployed.

IF ( ( EmergencyShutdown ) .AND. ( GenSpeedF < 1 ) ) THEN ! If emergency shutdown has
been initiated and generator speed is almost zero.

    IF ( brakeOff ) THEN
        brakeStartTime = ZTime
        brakeOff = .FALSE.
        WRITE(*,*) 'HSS Brake initiated at T = ', ZTime, ' GenSpeedF = ', GenSpeedF
        HSSBrFrac = 0.0
    ELSEIF ( (ZTime-brakeStartTime) < HSSBrDT ) THEN
        HSSBrFrac = (ZTime-brakeStartTime)/HSSBrDT
    ELSE
        HSSBrFrac = 1.0 !Engage brake
    ENDIF
ELSE
    HSSBrFrac = 0.0
ENDIF

RETURN
END SUBROUTINE UserHSSBr

```

E.5 Module EAControl()

The following module is added to FAST_Mods.f90.

```
MODULE EAControl

! This MODULE stores variables used in the controllers programmed by Eric Anderson.

USE Precision

! Parameters read in from input file primary.fst
REAL(ReKi) :: TimeDRStart ! Time for turbine to initiate derating
REAL(ReKi) :: TimeDREnd ! Time for turbine to start returning to full rated operation
REAL(ReKi) :: DerateFactor ! Ammount turbine will be derated (fraction of 1)
REAL(ReKi) :: TEmShutdown ! Time to initiate an emergency shutdown of the turbine
REAL(ReKi) :: maxOverspeed ! The maximum safe overspeed (%). An overspeed larger than
this will initiate an emergency shutdown.

! The following variables are used by several control subroutines.
LOGICAL :: EmergencyShutdown = .FALSE. ! Signal instructing the turbine to execute
emergency shutdown. Will remain FALSE unless changed by overspeed or TEmShutdown.
REAL(ReKi) :: GenSpeedF ! Filtered HSS (generator) speed, rad/s.
REAL(ReKi) :: PC_RefSpd ! Desired (reference) HSS speed for pitch controller, rad/s.
REAL(ReKi) :: PC_MinPit ! Minimum pitch setting in pitch controller, rad.
REAL(ReKi) :: VS_Rgn2_K ! Generator torque constant in Region 2 (HSS side),
N-m/(rad/s)^2.
REAL(ReKi) :: VS_RtPwr ! Rated generator generator power in Region 3, Watts. --
chosen to be 5MW divided by the electrical
LOGICAL :: controlDebug ! Flag to turn on debug output for controller
INTEGER(4) :: modCounterPitch = 1 ! Counter to see how often module data is accessed.
INTEGER(4) :: modCounterTorque = 1 ! Counter to see how often module data is accessed.
```

E.6 Additions to input file primary.fst

The following lines were added to the primary.fst input file just after the LINEARIZATION CONTROL section.

```
----- DERATE PARAMETERS -----
150.0      TimeDRStart    -Time for turbine to initiate derating
200.0      TimeDREnd     -Time for turbine to start returning to full rated
operation
0.2        DerateFactor   -Ammount turbine will be derated (fraction of 1)
9999.9     TEmShutdown   -Time to initiate an emergency shutdown of the turbine
15.0       maxOverspeed  -The maximum safe overspeed (%). An overspeed larger than
this will initiate an emergency shutdown.
```

E.7 Additions to FAST_IO.f90

The following lines are added to the FAST_IO.f90 immediately after the FAST LINEARIZATION CONTROL PARAMETERS section.

```
! ----- DERATE PARAMETERS -----  
  
! Skip the comment line.  
CALL ReadCom ( UnIn, PriFile, 'derate parameters' )  
  
! TimeDRStart - Time for turbine to initiate derating  
CALL ReadRVar ( UnIn, PriFile, TimeDRStart, 'TimeDRStart', 'Time for turbine to  
initiate derating' )  
  
! TimeDREnd - Time for turbine to start returning to full rated operation  
CALL ReadRVar ( UnIn, PriFile, TimeDREnd, 'TimeDREnd', 'Time for turbine to start  
returning to full rated operation' )  
  
! DerateFactor - Ammount turbine will be derated  
CALL ReadRVar ( UnIn, PriFile, DerateFactor, 'DerateFactor', 'Ammount turbine will be  
derated' )  
  
! TEmShutdown - Time to initiate an emergency shutdown  
CALL ReadRVar ( UnIn, PriFile, TEmShutdown, 'TEmShutdown', 'Time to initiate emergency  
shutdown' )  
  
! maxOverspeed - The maximum safe overspeed (%).  
CALL ReadRVar ( UnIn, PriFile, maxOverspeed, 'maxGenOS', 'The maximum safe overspeed  
(%)' )  
  
! controlDebug - Flag to turn on debug output for controller.  
CALL ReadLVar ( UnIn, PriFile, controlDebug, 'controlDebug', 'Flag to turn on debug &  
output for controller' )
```

REFERENCES

- [1] Steve Sawyer and Klaus Rave. Global wind report—annual market update 2015. Technical report, Global Wind Energy Council, 2015.
- [2] J.M. Jonkman, S. Butterfield, W. Musial, and G. Scott. Definition of a 5-mw reference wind turbine for offshore system development. Technical report, National Renewable Energy Laboratory, Golden, CO, 2009.
- [3] Jonathan Berg, Dale Berg, and Jon White. Fabrication, integration and initial testing of a smart rotor. In *Proceedings of the 50th AIAA Aerospace Sciences Meeting, Nashville, Tennessee*, 2012.
- [4] David Schlipf, Ervin Bossanyi, Carlo Enrico Carcangiu, Tim Fischer, Timo Maul, and Michele Rossetti. Lidar assisted collective pitch control. Technical report, ProjectUpwind, 2011.
- [5] B.J. Jonkman and M. Buhl. *Turbsim User's Guide*. National Renewable Energy Laboratory, Golden, CO, 2012.
- [6] National Renewable Energy Laboratory. National wind technology center, m2 tower data, april 25, 2013. Available: http://www.nrel.gov/midc/nwtc_m2/, 2013.
- [7] E. Williams, J. Hensley, and E. Salerno. Awea us wind industry annual market report year ending 2016. Technical report, American Wind Energy Association, Washington, DC, 2016.
- [8] Renewables 2011 global status report. Technical report, REN21, 2011.
- [9] T. Burton, N. Jenkins, D. Sharpe, and E. Bossanyi. *Wind Energy Handbook, Second Edition*. John Wiley and Sons, 2011.
- [10] Torben Juul Larsen, Helge A Madsen, and Kenneth Thomsen. Active load reduction using individual pitch, based on local blade flow measurements. *Wind Energy*, 8(1):67–80, 2005.
- [11] E.A. Bossanyi. Developments in individual blade pitch control. In *Proceedings of “the Science of Making Torque from Wind” Conference*, pages 486–497, 2004.
- [12] C.P. Van Dam, Dale E Berg, and Scott J Johnson. Active load control techniques for wind turbines. Technical report, Sandia National Laboratories, 2008.
- [13] General Electric. Ge energy 1.5 mw wind turbine brochure, 2009.

- [14] A. Manjock. Evaluation report, design codes fast and adams for load calculation of onshore wind turbines. Technical report, Germanischer Lloyd WindEnergie GmbH, 2005.
- [15] K. Z. Østergaard, P. Brath, and J. Stoustrup. Estimation of effective wind speed. *Journal of Physics: Conference Series* 75, 2007.
- [16] E.L. Van der Hooft and T.G. Van Engelen. Estimated wind speed feed forward control for wind turbine operation optimisation. In *Proceedings of European Wind Energy Conference in London, UK*, 2004.
- [17] David Schlipf. Lidar assisted control of wind turbines. *Struttgart Wind Energy. Universitat Stuttgart*, 2014.
- [18] AD Platt and ML Buhl Jr. Wt _ perf user guide for version 3. 05. 00. Technical report, National Renewable Energy Laboratory, Golden, CO, 2012.
- [19] Geoffrey Ingram Taylor. The spectrum of turbulence. In *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, volume 164, pages 476–490. The Royal Society, 1938.
- [20] David JC Dennis and Timothy B Nickels. On the limitations of taylor's hypothesis in constructing long structures in a turbulent boundary layer. *Journal of Fluid Mechanics*, 614:197–206, 2008.
- [21] V.W. Goldschmidt, M.F. Young, and E.S. Ott. Turbulent convective velocities (broadband and wavenumber dependent) in a plane jet. *Journal of Fluid Mechanics*, 105:327–345, 1981.
- [22] Juan C Del Alamo and Javier Jiménez. Estimation of turbulent convection velocities and corrections to taylor's approximation. *Journal of Fluid Mechanics*, 640:5–26, 2009.
- [23] Callum Atkinson, Nicolas Alexander Buchmann, and Julio Soria. An experimental investigation of turbulent convection velocities in a turbulent boundary layer. *Flow, Turbulence and Combustion*, 94(1):79–95, 2015.
- [24] Isaac Van der Hoven. Power spectrum of horizontal wind speed in the frequency range from 0.0007 to 900 cycles per hour. *Journal of Meteorology*, 14(2):160–164, 1957.
- [25] Chad W Higgins, Martin Froidevaux, Valentin Simeonov, Nikki Vercauteren, Caitlin Barry, and Marc B Parlange. The effect of scale on the applicability of taylor's frozen turbulence hypothesis in the atmospheric boundary layer. *Boundary-layer meteorology*, 143(2):379–391, 2012.

- [26] David Schlipf, Tim Fischer, Carlo Enrico Carcangiu, Michele Rossetti, and Ervin Bossanyi. Load analysis of look-ahead collective pitch control using lidar. In *Proceedings of the 10th German Wind Energy Conference DEWEK*, Bremen, Germany, 2010.
- [27] Ahmet Arda Ozdemir, Peter Seiler, and Gary J Balas. Benefits of preview wind information for region 2 wind turbine control. In *51st AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, page 317, Grapevine, TX, 2013.
- [28] Fiona Dunne, Lucy Y Pao, Alan D Wright, Bonnie Jonkman, Neil Kelley, and Eric Simley. Adding feedforward blade pitch control for load mitigation in wind turbines: Non-causal series expansion, preview control, and optimized fir filter methods. In *Proc. 49th AIAA Aerospace Sciences Meeting*, Orlando, FL, 2011.
- [29] David Schlipf and Martin Kühn. Prospects of a collective pitch control by means of predictive disturbance compensation assisted by wind speed measurements. In *Proceedings of the 9th German Wind Energy Conference DEWEK 2008*, Bremen, Germany, 2008.
- [30] David Schlipf, Stefan Kapp, Jan Anger, Oliver Bischoff, Martin Hofsäß, Andreas Rettenmeier, and Martin Kühn. Prospects of optimization of energy production by lidar assisted control of wind turbines. In *EWEA 2011 conference proceedings*, Brussels, Belgium, 2011.
- [31] David Schlipf, Dominik Johannes Schlipf, and Martin Kühn. Nonlinear model predictive control of wind turbines using lidar. *Wind Energy*, 16(7):1107–1129, 2013.
- [32] Andrew Scholbrock, Paul Fleming, Lee Fingersh, Alan Wright, David Schlipf, Florian Haizmann, Fred Belen, et al. Field testing lidar based feed-forward controls on the nrel controls advanced research turbine. In *51th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition*, Grapevine, TX, 2013.
- [33] Jason M Jonkman and Marshall L Buhl Jr. *FAST User's Guide*. National Renewable Energy Laboratory, Golden, CO, 2005.
- [34] Jason Jonkman. Nrel national wind technology center forum: Blade pitch control. <https://wind.nrel.gov/forum/wind/viewtopic.php?f=30I&t=1048>, 2014.
- [35] Fiona Dunne, David Schlipf, Lucy Pao, Alan Wright, Bonnie Jonkman, Neil Kelley, and Eric Simley. Comparison of two independent lidar-based pitch control designs. In *50th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition*, page 1151, Nashville, TN, 2012.
- [36] *Wind Turbine—Part 1: Design Requirements, IEC 61400-1*. International Electrotechnical Commission, Geneva, Switzerland, ed. 3 edition, 2005.

- [37] Ameet S Deshpande and Rhonda R Peters. Wind turbine controller design considerations for improved wind farm level curtailment tracking. In *2012 IEEE Power and Energy Society General Meeting*, San Diego, CA, 2012.
- [38] Sari Fink, C Mudd, K Porter, and Brett Morgenstern. Wind energy curtailment case studies. Technical report, National Renewable Energy Laboratory, Golden, CO, 2009.
- [39] Jarmo Aho, Andrew Buckspan, Jason Laks, Paul Fleming, Yunho Jeong, Fiona Dunne, Matthew Churchfield, Lucy Pao, and Kenneth Johnson. A tutorial of wind turbine control for supporting grid frequency through active power control. In *American Control Conference (ACC), 2012*, pages 3120–3131, Montréal, Canada, 2012. IEEE.
- [40] Jacob Aho, Andrew Buckspan, Lucy Pao, Paul Fleming, et al. An active power control system for wind turbines capable of primary and secondary frequency control for supporting grid reliability. In *Proc. 51st AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition*, Grapevine, TX, 2013.
- [41] Benjamin Biegel, Daria Madjidian, Vedrana Spudic, Anders Rantzer, and Jakob Stoustrup. Distributed low-complexity controller for wind power plant in derated operation. In *2013 IEEE International Conference on Control Applications (CCA)*, pages 146–151, Hyderabad, India, 2013. IEEE.
- [42] Phillip W Richards, D Todd Griffith, and Dewey H Hodges. Operating strategies and design recommendations for mitigating local damage effects in off-shore turbine blades. Technical report, Sandia National Laboratories, Albuquerque, NM, 2014.
- [43] Daniel Todd Griffith. Structural health and prognostics management for off-shore wind plants; final report of sandia r&d activities. Technical report, Sandia National Laboratories (SNL-NM), Albuquerque, NM (United States), 2015.
- [44] Mate Jelavić, Vlaho Petrović, Mario Barišić, and Ivana Ivanović. Wind turbine control beyond the cut-out wind speed. In *Annual Conference and Exhibition of European Wind Energy Association (EWEA2013)*, Vienna, Austria, 2013.
- [45] Vlaho Petrović and Carlo Luigi Bottasso. Wind turbine optimal control during storms. *Journal of Physics: Conference Series*, 524, 2014.
- [46] Vlaho Petrović and Carlo Luigi Bottasso. Wind turbine envelope riding. In *33rd Wind Energy Symposium-AIAA SciTech 2015*, Kissimmee, FL, 2015.
- [47] I. Carlén. Some simple concepts for dynamic wind sector management of wind farms. Technical report, Teknikgruppen AB, Täby, Sweden, 2010.

- [48] Susan A Frost, Kai Goebel, and Léo Obrecht. Integrating structural health management with contingency control for wind turbines. *International Journal of Prognostics and Health Management, Special Issue on Wind Turbine PHM*, page 11, 2013.
- [49] Raymond Chow and C Pv Dam. Verification of computational simulations of the nrel 5 mw rotor with a focus on inboard flow separation. *Wind Energy*, 15(8):967–981, 2012.
- [50] Raymond Chow and van CP Dam. Computational investigations of blunt trailing-edge and twist modifications to the inboard region of the nrel 5 mw rotor. *Wind Energy*, 16(3):445–458, 2013.
- [51] M.J. Churchfield, S. Lee, and P. Moriarty. *Overview of the simulator for offshore wind farm application (SOWFA)*. National Renewable Energy Laboratory, Golden, CO, 2012.
- [52] Sang Lee, Matthew Churchfield, Patrick Moriarty, J Jonkman, and J Michalakes. Atmospheric and wake turbulence impacts on wind turbine fatigue loadings. In *50th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, page 540, Nashville, TN, 2012.
- [53] Paul Fleming, Pieter Gebraad, Jan-Willem van Wingerden, Sang Lee, Matt Churchfield, Andrew Scholbrock, John Michalakes, Kathryn Johnson, and Pat Moriarty. The sowfa super-controller: A high-fidelity tool for evaluating wind plant control approaches. In *Proceedings of the EWEA Annual Meeting*, Vienna, Austria, 2013.
- [54] Matthew J Churchfield, S Lee, and PJ Moriarty. Adding complex terrain and stable atmospheric condition capability to the simulator for on/offshore wind farm applications (sowfa). In *First Symposium on OpenFOAM in Wind Energy*, Oldenburg, Germany, 2013.
- [55] Joseph Smagorinsky. General circulation experiments with the primitive equations: I. the basic experiment. *Monthly weather review*, 91(3):99–164, 1963.
- [56] Jens Nøjrkær Sołyrkensen and Wen Zhong Shen. Numerical modeling of wind turbine wakes. *Journal of Fluids Engineering*, 124(2):393–399, 2002.
- [57] JD Mirocha, B Kosovic, ML Aitken, and JK Lundquist. Implementation of a generalized actuator disk wind turbine model into the weather research and forecasting model for large-eddy simulation applications. *Journal of Renewable and Sustainable Energy*, 6(1):013104, 2014.
- [58] J.M. Jonkman. Overview of the fast servo-elastic module. In *NREL Wind Turbine Modeling Workshop*, Golden, CO, 2013. National Renewable Energy Laboratory.

- [59] Pieter G Buning, Dennis C Jespersen, Thomas H Pulliam, WM Chan, Jeffrey P Slotnick, SE Krist, and Kevin J Renze. *Overflow User's Manual, Version 1.8aa*. NASA Langley Research Center, Hampton, VA, 2003.
- [60] Robert H Nichols and Peter G Buning. *User's Manual for OVERFLOW 2.1, Version 2.1t*. University of Alabama and NASA Langley Research Center, 2008.
- [61] Dennis Jespersen, Thomas Pulliam, Pieter Buning, Dennis Jespersen, Thomas Pulliam, and Pieter Buning. Recent enhancements to overflow. In *35th Aerospace Sciences Meeting and Exhibit*, page 644, Reno, NV, 1997.
- [62] Robert Nichols, Robert Tramel, and Pieter Buning. Solver and turbulence model upgrades to overflow 2 for unsteady and high-speed applications. In *24th AIAA Applied Aerodynamics Conference*, page 2824, San Francisco, CA, 2006.
- [63] Richard M Beam and RoF Warming. An implicit factored scheme for the compressible navier-stokes equations. *AIAA journal*, 16(4):393–402, 1979.
- [64] Florian R Menter. Two-equation eddy-viscosity turbulence models for engineering applications. *AIAA journal*, 32(8):1598–1605, 1994.
- [65] Robert H Nichols and Bonnie D Heikkinen. Validation of implicit algorithms for unsteady flows including moving and deforming grids. In *43rd AIAA Aerospace Sciences Meeting and Exhibit*, Reno, NV, 2005.
- [66] Joseph L Steger, F Carroll Dougherty, and John A Benek. A chimera grid scheme. *Advance in Grid Generation*, vol. 5, 1983.
- [67] William M Chan. Overset grid technology development at nasa ames research center. *Computers & Fluids*, 38(3):496–503, 2009.
- [68] William M Chan, Stuart E Rogers, Steven M Nash, Pieter G Buning, and Robert L Meakin. *User's Manual for Chimera Grid Tools, version 1.8*. NASA Ames Research Center, Moffett Field, CA, 2002.
- [69] William Chan. The overgrid interface for computational simulations on overset grids. In *32nd AIAA Fluid Dynamics Conference and Exhibit*, page 3188, St. Louis, MO, 2002.
- [70] Robert Meakin. Object x-rays for cutting holes in composite overset structured grids. In *15th AIAA computational fluid dynamics conference*, page 2537, Anaheim, CA, 2001.
- [71] William Chan, Reynaldo Gomez, Stuart Rogers, and Pieter Buning. Best practices in overset grid generation. In *32nd AIAA Fluid Dynamics Conference and Exhibit*, page 3191, St. Louis, MO, 2002.

- [72] J. Bardina, P. Huang, T. Coakley, J. Bardina, P. Huang, and T. Coakley. Turbulence modeling validation. In *28th Fluid dynamics conference*, page 2121, Snowmass Village, CO, 1997.
- [73] Robert Meakin. Automatic off-body grid generation for domains of arbitrary size. In *15th AIAA Computational Fluid Dynamics Conference*, page 2536, Anaheim, CA, 2001.
- [74] Luis Martinez, Stefano Leonardi, Matthew Churchfield, and Patrick Moriarty. A comparison of actuator disk and actuator line wind turbine models and best practices for their use. In *50th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, page 900, 2012.
- [75] Paul A Fleming, Pieter MO Gebraad, Sang Lee, Jan-Willem van Wingerden, Kathryn Johnson, Matt Churchfield, John Michalakes, Philippe Spalart, and Patrick Moriarty. Evaluating techniques for redirecting turbine wakes using sowfa. *Renewable Energy*, 70:211–218, 2014.
- [76] Cristina L Archer, Sina Mirzaeisefat, and Sang Lee. Quantifying the sensitivity of wind farm performance to array layout options using large-eddy simulation. *Geophysical Research Letters*, 40(18):4963–4970, 2013.
- [77] M Maché, H Mouslim, and L Mervoyer. From meso-scale to micro scale les modelling: Application by a wake effect study for an offshore wind farm. In *ITM Web of Conferences*, volume 2. EDP Sciences, 2014.
- [78] Andreas Søndergaard Pedersen and Christian Sigge Steiniche. Safe operation and emergency shutdown of wind turbines. Master’s thesis, Aalborg University, Aalborg, Denmark, 2012.
- [79] Paul Fleming, Pieter Gebraad, Matt Churchfield, Sang Lee, Kathryn Johnson, John Michalakes, Jan-Willem van Wingerden, and Patrick Moriarty. *Sowfa+ super controller user’s manual*. National Renewable Energy Laboratory, Golden, CO, 2013.
- [80] Niels Troldborg. *Actuator line modeling of wind turbine wakes*. PhD thesis, Technical University of Denmark, Kongens Lyngby, Denmark, 2009.
- [81] Luis A Martínez-Tossas, Matthew J Churchfield, and Charles Meneveau. Large eddy simulation of wind turbine wakes: detailed comparisons of two codes focusing on effects of numerics and subgrid modeling. In *Journal of Physics: Conference Series*, volume 625, page 012024. IOP Publishing, 2015.
- [82] Matthew J Churchfield, Sang Lee, Patrick J Moriarty, Yujia Hao, Matthew A Lackner, Rebecca Barthelmie, Julie K Lundquist, and Gregory Oxley. A comparison of the dynamic wake meandering model, large-eddy simulation, and field data at the egmond aan zee offshore wind plant. In *33rd Wind Energy Symposium*, page 0724, Kissimmee, FL, 2015.

- [83] Niels Troldborg, Frederik Zahle, Pierre-Elouan Réthoré, and Niels N Sørensen. Comparison of wind turbine wake properties in non-sheared inflow predicted by different computational fluid dynamics rotor models. *Wind Energy*, 18(7):1239–1250, 2015.
- [84] Niels Troldborg, Jens N Sorensen, and Robert Mikkelsen. Numerical simulations of wake characteristics of a wind turbine in uniform inflow. *Wind Energy*, 13(1):86–99, 2010.
- [85] H Aa Madsen, Gunner Chr Larsen, Torben J Larsen, Niels Troldborg, and R Mikkelsen. Calibration and validation of the dynamic wake meandering model for implementation in an aeroelastic code. *Journal of Solar Energy Engineering*, 132(4), 2010.
- [86] G.J. Hayman. Mlife theory manual for version 1.00. Technical report, National Renewable Energy Laboratory, Golden, CO, 2012.
- [87] G.J. Hayman and M.L. Buhl Jr. *Mlife users guide for version 1.00*. National Renewable Energy Laboratory, Golden, CO, 2012.
- [88] Arvid Palmgren. The service life of ball bearings. *Zeitschrift des Vereines Deutscher Ingenieure*, 68(14):339–341, 1924.
- [89] Milton A Miner. Cumulative damage in fatigue. *Journal of applied mechanics*, 12(3):159–164, 1945.
- [90] G. Stewart, M. Lackner, L. Haid, D. Matha, J. Jonkman, and A. Robertson. Assessing fatigue and ultimate load uncertainty in floating offshore wind turbines due to varying simulation length. *Safety, Reliability, Risk and Life-Cycle Performance of Structures and Infrastructures*, page 239, 2013.