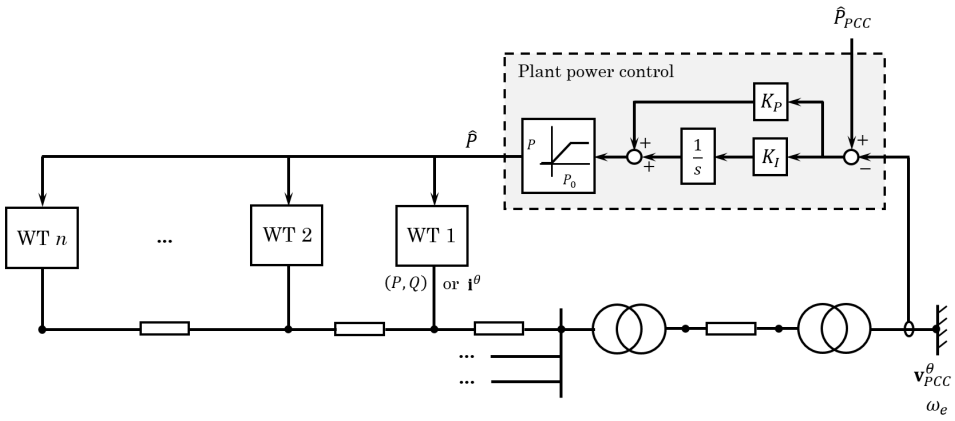
2019:00342 - Unrestricted

Report

An Electromechanical Model of the TotalControl Reference Wind Power Plant

Author(s)
Karl Merz, Konstanze Kölle, Andrzej Holdyk



SINTEF Energy Research AS
Power Conversion and Transmission
March 22, 2019

SINTEF Energi AS
SINTEF Energy Research AS

Address:
Postboks 4761 Torgarden
7465 Trondheim
NORWAY

www.sintef.no

Enterprise Number: NO 939 350 675 MVA

KEYWORDS:
Wind energy
Systems engineering
Wind plant control

Report

An Electromechanical Model of the TotalControl Reference Wind Power Plant

VERSION
1.0

DATE
March 20, 2019

AUTHOR(S)
Karl Merz, Konstanze Kölle, Andrzej Holdyk

CLIENT(S)
TotalControl

CLIENT'S REFERENCE
D1.05

PROJECT
TotalControl

NUMBER OF PAGES AND ATTACHMENTS
55

ABSTRACT
An electromechanical model of the TotalControl Reference Wind Power Plant has been built. This is based on SINTEF's STAS software, for the system dynamics and control of wind power plants. An overview is given of the theory behind the STAS software, and the architecture of the system model. A wind turbine control system has been designed, which is well-suited to the STAS model and the upcoming research on plant control algorithms in the TotalControl project. The STAS model consists of linear/nonlinear equation pairs, and it is shown how these can be used to obtain Newton-Raphon solutions for operating points, as well as precise gradients with the complex step approach. A procedure for scaling stochastic atmospheric turbulence is presented, that greatly accelerates design and optimization studies involving turbulent wind fields. The detailed electrical design of the TotalControl Reference Wind Power Plant is described, and this is applied together with the wind turbine module to construct a holistic electromechanical model of the plant. The dynamics of the plant is characterized in terms of transfer functions.

PREPARED BY
Karl Merz

SIGNATURE

CHECKED BY
Harald Svendsen

SIGNATURE

APPROVED BY
Knut Samdal

SIGNATURE

REPORT NUMBER
2019:00342

ISBN
978-82-14-06830-6

CLASSIFICATION
Unrestricted

CLASSIFICATION THIS PAGE
Unrestricted

Document History

VERSION	DATE	VERSION DESCRIPTION
1.0	19.03.2019	Original document

SINTEF

Contents

1 Background

A wind power plant is an arrangement of wind turbines feeding into a common electric collection grid. The plant-scale dynamics emerge from interactions between individual wind turbines

1. through power flows, terminal voltages, and resonant modes in the electric grid;
2. as the turbines remove kinetic energy from the atmospheric flow, and introduce vorticity, wakes, and turbulence that are convected through the plant; and
3. in response to coordinated control signals sent from the plant operator, often in response to events or stress in the electric grid, or in an attempt to actively influence the properties of the atmospheric flow.

The present electromechanical model encompasses all parts of the plant except for the atmospheric flow; the wind speed at each turbine is considered as an input. The electromechanical model extends from the local aerodynamics at the blades, through the support structures, to the foundation in the sea bed; and from the generator, through the electrical components, and out into the grid.

The scope is limited to frequencies of relevance for plant-wide dynamics, primary wind turbine structural loads, and the design of high-level turbine and plant control systems: usually below 1 Hz, on occasion up to perhaps 10 Hz, though always well below the AC electrical frequency of the grid. From the electrical standpoint, then, we are concerned with the small-signal stability and subsynchronous oscillations of the lowest frequency grid modes; and particularly the way these are influenced by different wind plant control strategies. From the mechanical standpoint, we are interested in how the interactions between the grid and plant controls – and, with the help of an external model, the atmosphere – impact key quantities like structural loads and power generation.

The electromechanical model of the TotalControl reference wind power plant (TC-RWP) is created using SINTEF's STAS program. This is a set of Octave/Matlab scripts[1] that constructs a unified state-space model of a wind power plant, including the wind turbines, electrical systems, and hierarchical controls. The outstanding characteristic of STAS is that the model is numerically smooth, and generates, at any given initial condition, both a nonlinear state-space model and a high-numerical-precision linearization. The numerical smoothness allows precise gradients to be obtained, which is useful for Newton-Raphson solutions of operating points, as well as multidisciplinary optimization and optimal control. Having obtained a linear state-space model, one can apply the powerful tools of linear systems theory, such as modal dynamics, model reduction, and synthesis and tuning of multivariable controls.

A number of technical challenges were tackled during development of the STAS program and TC-RWP model. These are brought into focus in the chapters below. Chapter 2 describes the development of a unified state-space model, the equations of motion, and how the modules are linked together. A special wind turbine control algorithm, described in Chapter 3, was developed for use within the STAS state-space framework. The linear/nonlinear equation pairs allow for a Newton-Raphson solution of operating points; Chapter 4 includes some tricks used to aid convergence. STAS supports complex step gradients, and Chapter 5 gives a brief description of this technique. Chapter 6 presents a method for scaling atmospheric turbulence spectra, such that it is not necessary to generate a new set of spectra as operating variables like the rotor speed are perturbed. A detailed electrical design for the TC-RWP is described in Chapter 7, containing sufficient information to construct a dynamic model for small-signal stability analysis. Finally, Chapter 8 brings it all together, albeit briefly, computing the closed-loop transfer functions of the TC-RWP to an operator power command.

[1] Octave shares Matlab's scripting language; Octave has been used to develop the present results, but STAS can also be run in a Matlab environment.

2 Unified state-space electromechanical model

2.1 Introduction

The electromechanical model of the TC-RWP is built up from a hierarchy of modules; Fig. 1 gives some idea of the way things are defined and interconnected at the plant and turbine levels. At the highest level there is a wind turbine module and an electrical grid module. The wind turbine is in turn composed of aeroelastic, electrical, and control modules, while the electrical grid consists of components like cables and transformers. The lowest level of the hierarchy consists of single components, like a blade element, PI control function, or electric cable.

Each module accepts interface, external, and state variables at time t as inputs, and returns the nonlinear dynamic function

$$\mathbf{N}(\mathbf{x},\mathbf{p})\frac{d\mathbf{x}}{dt} = \mathbf{f}(\mathbf{x},\mathbf{u},\mathbf{z},\mathbf{p}), \qquad \begin{bmatrix}\mathbf{y}\\ \mathbf{z}\end{bmatrix} = \begin{bmatrix}\mathbf{g}_y(\mathbf{x},\mathbf{u},\mathbf{z},\mathbf{p})\\ \mathbf{g}_z(\mathbf{x},\mathbf{u},\mathbf{z},\mathbf{p})\end{bmatrix} \tag{1}$$

and its linearization with respect to the dynamic variables,[2]

$$\begin{aligned}\mathbf{N}_0\frac{d\Delta\mathbf{x}}{dt} &= \left(\left.\frac{\partial\mathbf{f}}{\partial\mathbf{x}}\right|_0 - \left.\frac{\partial\mathbf{N}}{\partial\mathbf{x}}\right|_0 \left.\frac{d\Delta\mathbf{x}}{dt}\right|_0\right)\Delta\mathbf{x} + \left.\frac{\partial\mathbf{f}}{\partial\mathbf{u}}\right|_0\Delta\mathbf{u} + \left.\frac{\partial\mathbf{f}}{\partial\mathbf{z}}\right|_0\Delta\mathbf{z}\\ &= \mathbf{A}\,\Delta\mathbf{x} + \mathbf{B}_u\Delta\mathbf{u} + \mathbf{B}_z\Delta\mathbf{z}\end{aligned} \tag{2}$$

$$\begin{bmatrix}\Delta\mathbf{y}\\ \Delta\mathbf{z}\end{bmatrix} = \left.\frac{\partial\mathbf{g}}{\partial\mathbf{x}}\right|_0\Delta\mathbf{x} + \left.\frac{\partial\mathbf{g}}{\partial\mathbf{u}}\right|_0\Delta\mathbf{u} + \left.\frac{\partial\mathbf{g}}{\partial\mathbf{z}}\right|_0\Delta\mathbf{z} = \begin{bmatrix}\mathbf{C}_y\\ \mathbf{C}_z\end{bmatrix}\Delta\mathbf{x} + \begin{bmatrix}\mathbf{D}_{yu}\\ \mathbf{D}_{zu}\end{bmatrix}\Delta\mathbf{u} + \begin{bmatrix}\mathbf{D}_{yz}\\ \mathbf{D}_{zz}\end{bmatrix}\Delta\mathbf{z}. \tag{3}$$

Here $\mathbf{x}$ are the state variables, $\mathbf{u}$ are external inputs, $\mathbf{z}$ are interface variables, and $\mathbf{p}$ are static parameters. With the exception of Chapter 5, the focus is on the dynamic variables, and the dependence on $\mathbf{p}$ will often not be indicated explicitly.

The $\mathbf{N}$ matrix contains the mass matrix in the rows associated with the structural accelerations; otherwise, it is the identity matrix. It is invertible, and generally well-conditioned, but we postpone its inversion, for reasons discussed in Section 2.7.

2.2 Aeroelastic module

The aeroelastic module, STAS Aeroelastic, encompasses the local blade aerodynamics and the structural components: blades, low-speed shaft, nacelle, tower, and foundation. The aeroelastic module is further decomposed into an aerodynamics module and a structural module. The aerodynamics module handles the dynamic relationships between the incoming wind, the turbine wake, blade structural motions, and the aerodynamic forces along the blades. The structural module accounts for the deformation of the turbine structure, including rotor rotation, under the applied forces.

The interface variables are shown in Figure 2. The aerodynamic module takes the incoming wind speed in global coordinates $\mathbf{V}^g$ and structural positions $\mathbf{q}$ and velocities $\dot{\mathbf{q}}$ as inputs, and returns the aerodynamic forces on the blades $\mathbf{F}_a$, as well as the induced velocity $\mathbf{V}_i$. The structural module takes forces $\mathbf{F}$ and $\mathbf{F}_a$ as inputs, and returns $\mathbf{q}$ and $\dot{\mathbf{q}}$. The external force $\mathbf{F}$ includes the remaining environmental loads like ocean waves, actuator forces at the yaw bearing and pitch bearings, and the generator torque on the driveshaft and its reaction in the nacelle.

The complete derivation of the state-space equations implemented in STAS Aeroelastic is given by Merz (2018). Highlights are provided in Sections 2.2.1 and 2.2.2 below.

[2] Gradients with respect to static parameters $\mathbf{p}$ are at present handled by the complex step method, Chapter 5.

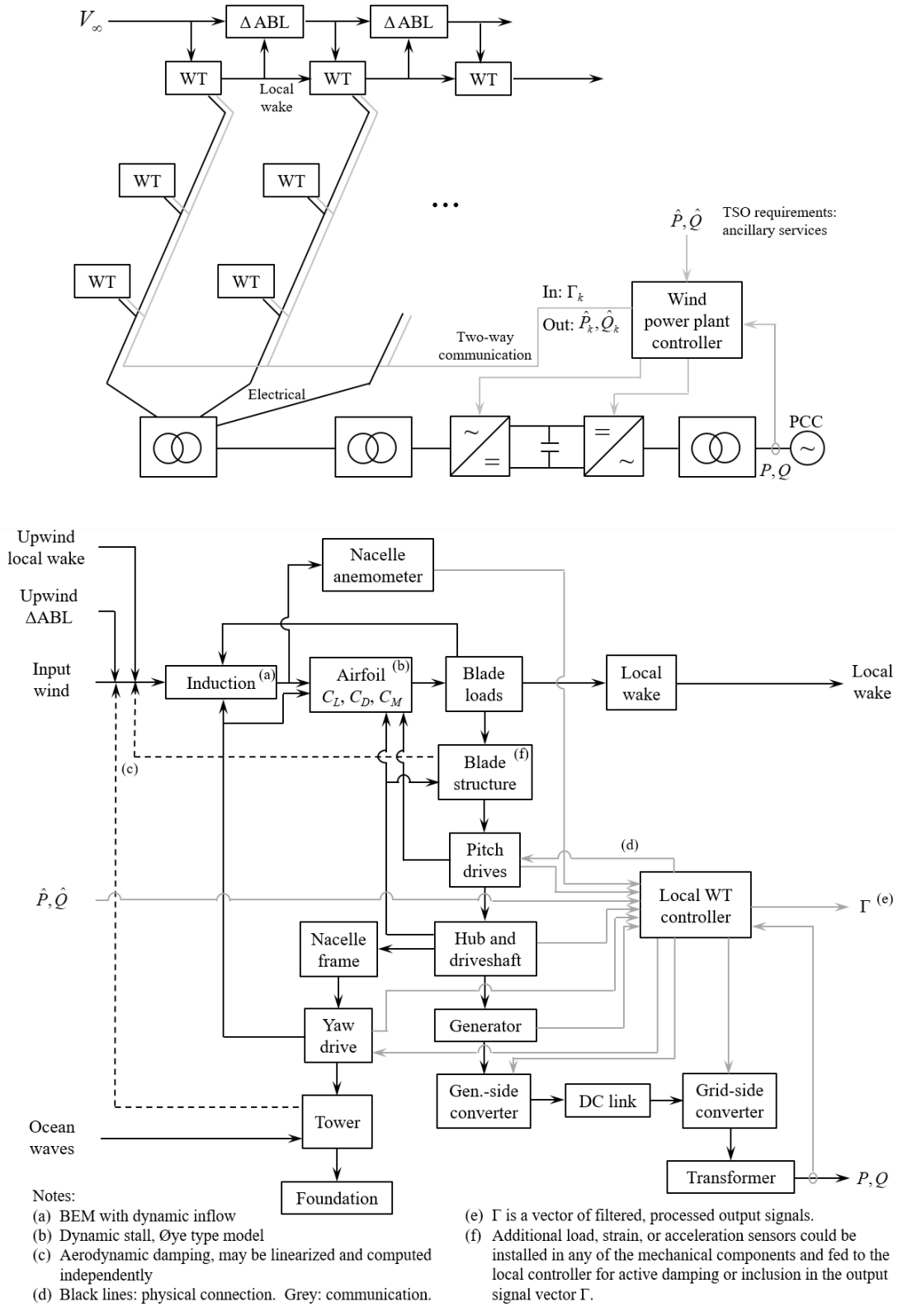


Figure 1: A block diagram of a generic wind power plant and wind turbine, illustrating the system architecture and interconnections.

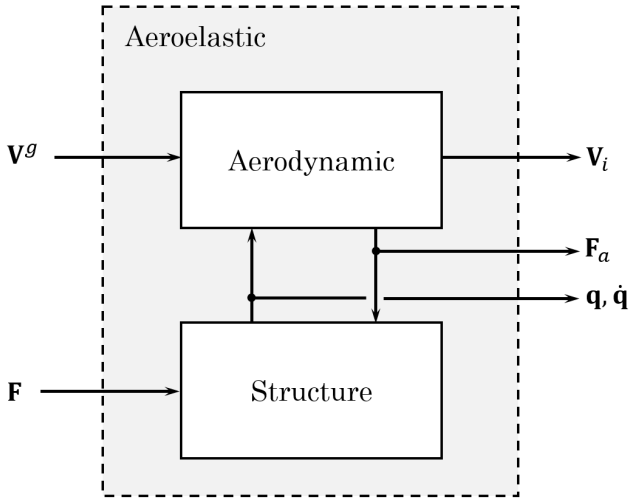


Figure 2: The interface of the aeroelastic module, and between the aerodynamic and structural sub-modules.

2.2.1 Structural dynamics

The structural model in STAS Aeroelastic is a multibody, corotational finite-element beam representation of the wind turbine. The turbine consists of a number of bodies: foundation, tower, nacelle, driveshaft, and blades. Each body can move rigidly in space, and deform elastically. Fig. 3 shows the arrangement of the bodies and their associated coordinate systems.

The equations of motion are developed using the Lagrange equations,

$$\frac{d}{dt}\frac{\partial E_K}{\partial \dot{\mathbf{q}}} + \frac{1}{2}\frac{\partial \dot{E}_D}{\partial \dot{\mathbf{q}}} + \frac{\partial E_P}{\partial \mathbf{q}} = \frac{\partial E_K}{\partial \mathbf{q}} + \frac{\partial W}{\partial \mathbf{q}}. \tag{4}$$

Here E_K is the kinetic energy, E_D is dissipated energy, E_P is potential energy, and W is the work done by applied forces. The $\mathbf{q}$ are the structural degrees-of-freedom: three positions and three rotation parameters for each node.

In a corotational finite element formulation (Felippa and Haugen 2005), the elements follow the deformation of the structure as rigid bodies, and themselves deform elastically by only a small amount. The variables that describe the location of the element in space are

1. the undeformed position and orientation – pose, for short – of a reference node on the body, relative to the global coordinate frame,
2. the deformed pose of this reference node, with respect to its undeformed pose,
3. the undeformed pose of the element's boundary nodes with respect to the reference node, and
4. the deformed pose of the nodes with respect to their undeformed pose.

The first and third of these are constant, while the second and fourth represent the degrees-of-freedom $\mathbf{q}$ of the structure. The essence of developing the equations of motion is to formulate coordinate transforms, and their time derivatives, in order to express the global position, velocity, and acceleration of the elements' pose. The detailed derivation of the equations of motion requires its own report (Merz 2018).

Whatever the specific formulation, the equations of motion of the structure can be put into the form

$$\mathbf{M}(\mathbf{q})\frac{d^2\mathbf{q}}{dt^2} + [\mathbf{G}(\mathbf{q},\dot{\mathbf{q}}) + \mathbf{C}(\mathbf{q})]\frac{d\mathbf{q}}{dt} - \mathbf{H}(\mathbf{q},\dot{\mathbf{q}}) + \mathbf{K}(\mathbf{q}) = \mathbf{Q}(\mathbf{q})\,\mathbf{F}, \tag{5}$$

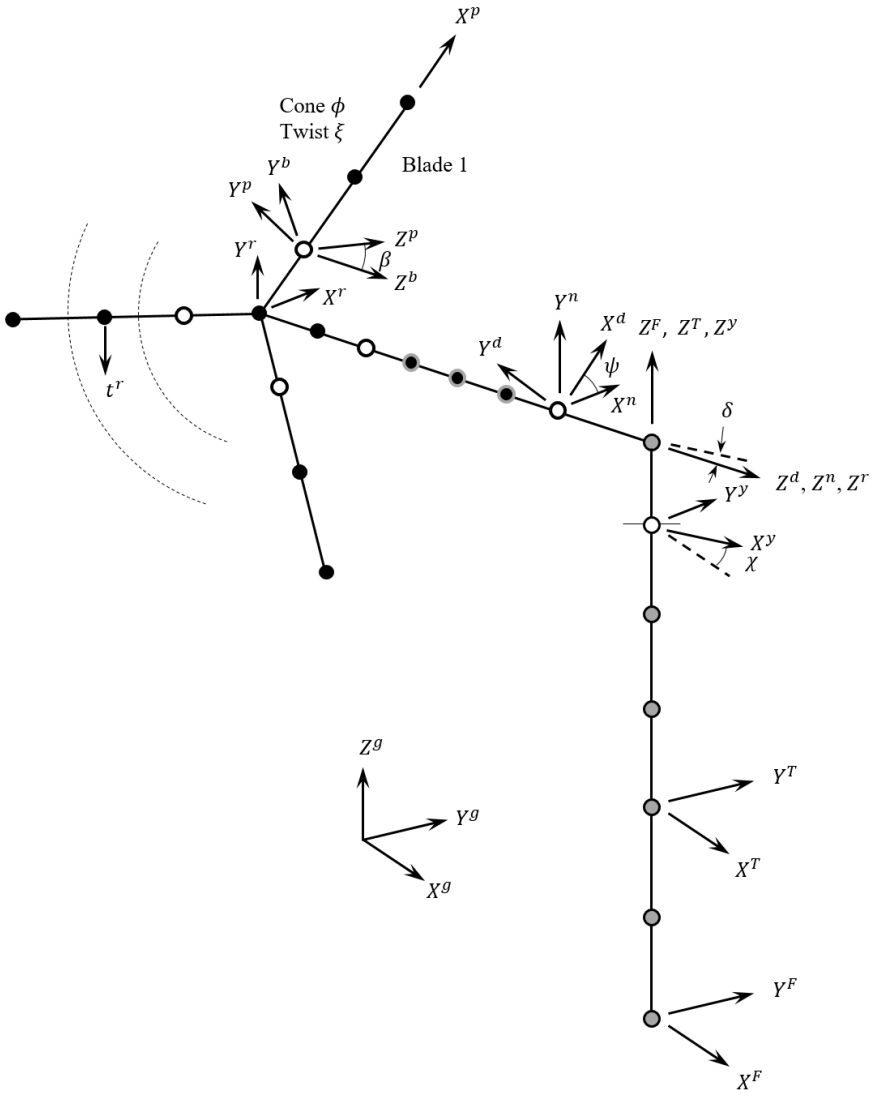


Figure 3: A sketch of the finite beam element model of the wind turbine structures, showing the bodies and key coordinate systems, and a representation of the nodes and elements.

where $\mathbf{M}$ is the mass matrix, $\mathbf{C}$ is the damping matrix, $\mathbf{G}$ is the gyroscopic matrix, $\mathbf{H}$ is a vector containing centrifugal forces, $\mathbf{K}$ is a vector of internal stiffness forces, $\mathbf{F}$ is a vector of externally applied nodal forces, and $\mathbf{Q}$ is a matrix converting the nodal forces into the generalized coordinates of $\mathbf{q}$.

The equations (5) can be linearized, and put into the well-recognized form

$$\tilde{\mathbf{M}}_0 \frac{d^2\Delta\mathbf{q}}{dt^2} + \tilde{\mathbf{C}}_0 \frac{d\Delta\mathbf{q}}{dt} + \tilde{\mathbf{K}}_0 \Delta\mathbf{q} = \mathbf{Q}_0 \Delta\mathbf{F}, \tag{6}$$

where only $\tilde{\mathbf{M}}$ and $\tilde{\mathbf{Q}}$ bear a direct resemblance to their counterparts in (5); the other terms are composites formed from the derivatives of the various nonlinear matrices and vectors in (5).

An eigenmode decomposition can be performed,

$$\mathbf{\Phi}^T \tilde{\mathbf{M}}_0 \mathbf{\Phi} \frac{d^2\Delta\boldsymbol{\eta}}{dt^2} + \mathbf{\Phi}^T \tilde{\mathbf{C}}_0 \mathbf{\Phi} \frac{d\Delta\boldsymbol{\eta}}{dt} + \mathbf{\Phi}^T \tilde{\mathbf{K}}_0 \mathbf{\Phi} \Delta\boldsymbol{\eta} = \mathbf{\Phi}^T \mathbf{Q}_0 \Delta\mathbf{F}, \tag{7}$$

converting to modal degrees-of-freedom η. Although the structural mode shapes are associated with the particular operating point at which (7) is linearized, the lower-frequency mode shapes tend to be relatively consistent at different operating points. Thus it is reasonable to apply a selected $\mathbf{\Phi}$ to the nonlinear equations (5), as

$$\mathbf{\Phi}^T \mathbf{M} \mathbf{\Phi} \frac{d^2\boldsymbol{\eta}}{dt^2} + \mathbf{\Phi}^T \left[\mathbf{G} + \mathbf{C}\right] \mathbf{\Phi} \frac{d\boldsymbol{\eta}}{dt} - \mathbf{\Phi}^T \mathbf{H} + \mathbf{\Phi}^T \mathbf{K} = \mathbf{\Phi}^T \mathbf{Q}\, \mathbf{F}. \tag{8}$$

In STAS the structural eigenmodes are computed for each body in isolation, so that each mode shape is associated with a particular body's deformation: blade 1 flapwise, tower fore-aft, drivetrain torsion, and so on.

2.2.2 Aerodynamics

The aerodynamics module implements a version of the blade element momentum (BEM) method (Merz 2018). The basics of this method can be found in other references (Burton *et al.* 2001, Hansen 2008), and Hansen (2004a, 2004b) employs a similar set of linearized equations. The jist of the BEM method is to estimate the change in the flow field due to the vortex wake – not by actually modelling the wake, but rather by assuming an ideal streamwise flow pattern, and performing a control volume analysis, a much-used approach in fluid mechanics.

The BEM method implemented in STAS is a full vector version, based on the momentum equation

$$\mathbf{F}^r = -2\rho A f W \mathbf{V}_{iq}, \quad W := \sqrt{[(\mathbf{V}^r_{/r})_z + f(\mathbf{V}^r_{iq})_z]^2 + (\mathbf{V}^r_{/r})^2_x + (\mathbf{V}^r_{/r})^2_y}, \tag{9}$$

where $\mathbf{F}^r$ is a vector of the aerodynamic forces on the blades, ρ is the air density, f is Prandtl's finite-blade tip-loss factor, $\mathbf{V}^r_{/r}$ is the wind velocity relative to the rotorplane, and $\mathbf{V}^r_{iq}$ is the quasi-steady induced velocity in rotorplane coordinates. The spanwise momentum balance is neglected, as it has no influence on the aerodynamic forces, according to the BEM method.

Low-order state-space filters, essentially first- and second-order low-pass filters, are used to model the dynamics of circulation lag and dynamic stall, as well as the influence of wake vortex dynamics (dynamic inflow), at each blade element. The equations are described in Merz (2018), and come from Snel and Schepers (1995), Leishman (2002), and Hansen (2008).

The number of states in the aerodynamic model can be reduced by two methods: slaving the aerodynamic states to the structural modes (Sønderby 2013), or using splines, as a function of the spanwise coordinate along the blades (Merz 2016). In the analyses of this report, spline reduction with six spanwise control points was employed.

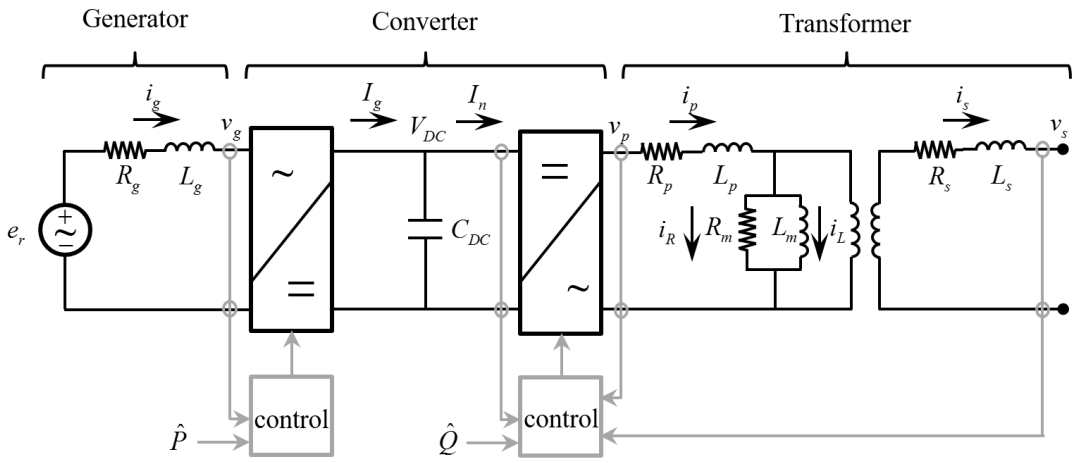


Figure 4: An equivalent circuit representation of the wind turbine electrical components.

2.3 Electrical module

The state-space electrical equations are described by Merz (2019b). The equations are based on elementary equivalent-circuit models of turbine and grid electrical components, and are adapted from Anaya-Lara et al. (2009), D'Arco *et al.* (2015), and Merz and Pedersen (2018). An equivalent circuit of the wind turbine electrical components is sketched in Fig. 4. The converter controls are given special consideration, as the hierarchy of measurements and feedback controls governs how rapidly the electrical system is able to respond to power set-point commands and disturbances.

Generically, the electrical states consist of currents through inductors, and voltages across capacitors. In the *abc* frame, the current equations are of the form

$$\mathbf{L}\frac{d\mathbf{i}}{dt} = -\mathbf{R}\mathbf{i} + \mathbf{v}_1 - \mathbf{v}_2, \tag{10}$$

where $\mathbf{L}$ is a 3-by-3 inductance matrix, $\mathbf{i}$ is the current, $\mathbf{R}$ is a diagonal resistance matrix, and $\mathbf{v}_1$ and $\mathbf{v}_2$ are the voltages at adjacent nodes. The voltage equations are

$$\mathbf{C}\frac{d\mathbf{v}}{dt} = \mathbf{i}_1 - \mathbf{i}_2, \tag{11}$$

where $\mathbf{C}$ is a 3-by-3 capacitance matrix, $\mathbf{v}$ is the voltage across the capacitor, and $\mathbf{i}_1$ and $\mathbf{i}_2$ are the net currents flowing into the capacitor. The equations are transformed to the *d*-*q* frame, employing the power-equivalent form of the transform,

$$T_a^\theta = \sqrt{23}\begin{bmatrix} \cos\theta & \cos(\theta - 2\pi/3) & \cos(\theta - 4\pi/3) \\ -\sin\theta & -\sin(\theta - 2\pi/3) & -\sin(\theta - 4\pi/3) \end{bmatrix}, \tag{12}$$

$$T_\theta^a = \sqrt{23}\begin{bmatrix} \cos\theta & -\sin\theta \\ \cos(\theta - 2\pi/3) & -\sin(\theta - 2\pi/3) \\ \cos(\theta - 4\pi/3) & -\sin(\theta - 4\pi/3) \end{bmatrix}. \tag{13}$$

The transformed equations, which have two rather than three states, since the three phases are assumed to be balanced,[3] are

$$\mathbf{T}_a^\theta \mathbf{L}\mathbf{T}_\theta^a \frac{d\mathbf{i}^\theta}{dt} = -\mathbf{T}_a^\theta \left(\mathbf{R}\mathbf{T}_\theta^a + \omega_e \mathbf{L}\frac{d\mathbf{T}_\theta^a}{d\theta}\right)\mathbf{i}^\theta + \mathbf{v}_1^\theta - \mathbf{v}_2^\theta \tag{14}$$

[3]It is fairly simple to adapt the equations to include the 0-component in addition to the *d* and *q* components if it were desired for some reason to use STAS for looking at unbalanced fault cases.

and

$$\mathbf{T}_a^\theta \mathbf{C} \mathbf{T}_\theta^a \frac{d\mathbf{v}^\theta}{dt} = -\omega_e \mathbf{T}_a^\theta \mathbf{C} \frac{d\mathbf{T}_\theta^a}{d\theta} \mathbf{v}^\theta + \mathbf{i}_1^\theta - \mathbf{i}_2^\theta. \tag{15}$$

Over the frequency band of interest, below the fundamental electrical frequency, the converters on the generator and network sides of the DC link provide a controllable terminal voltage. Taking the generator as an example, the equation describing the electrical dynamics is

$$\mathbf{T}_a^\theta \mathbf{L}_g \mathbf{T}_\theta^a \frac{d\mathbf{i}_g^\theta}{dt} = -\mathbf{T}_a^\theta \left(\mathbf{R}_g \mathbf{T}_\theta^a + \omega_g \mathbf{L}_g \frac{d\mathbf{T}_\theta^a}{d\theta} \right) \mathbf{i}_g^\theta - \mathbf{v}_g^\theta - \omega_g \mathbf{T}_a^\theta \frac{d\mathbf{T}_\theta^a}{d\theta} \boldsymbol{\lambda}_r^\theta, \tag{16}$$

which is essentially a version of (14). In (16) the term containing $\boldsymbol{\lambda}_r$ is the emf, while the terminal voltage $\mathbf{v}_g^\theta$ is controlled by the converter. A strategy of feedback linearization is employed, with the control law

$$\mathbf{v}_g^\theta = -L_g \left[\mathbf{K}_P (\hat{\mathbf{i}}_g^\theta - \bar{\mathbf{i}}_g^\theta) + \boldsymbol{\Psi}_g + \overline{\omega}_g \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \bar{\mathbf{i}}_g^\theta + K_F \overline{\omega}_g \boldsymbol{\lambda}_r^\theta \right] \tag{17}$$

attempting to cancel the nonlinearities in the generator dynamics and replace them with a desirable control response. Here $\boldsymbol{\Psi}_g$ is the integrated current error, including a tunable gain K_I, and K_P and K_F are also tunable gains. The network-side voltage control is implemented with the same type of strategy.

2.4 Actuator module

The STAS Actuator module (Merz 2019a) provides the link between the commanded blade pitch of the turbine controller, and the torque applied to the slew ring at the blade root. That is, the input to the module is a blade pitch command $\hat{\beta}$, and the output from the module is a torque T_b applied to the blade pitch degree-of-freedom. The present implementation includes a standard second-order filter to represent the actuator dynamics. In addition, there are two features of note. First, a spring and damper connect the actuator output with the slew ring. These can be tuned in order to model actuator flexibility and resonance. Second, the saturation of the pitch angle and rate is implemented with a smoothed profile, for instance Fig. 5, such that the linearization is locally continuous and numerically precise.

2.5 Control module

The interface of the control module is shown in Figure 6. Inputs of the control module are the power command sent by the plant operator $\hat{P}_o$, the measured rotor speed $\overline{\Omega}$, and the measured pitch angle $\overline{\beta}$. These main inputs may be augmented by the current electrical power $\overline{P}_e$, the blade root moments $\overline{\mathbf{M}}$ and tower velocities $\overline{\mathbf{V}}_\mathbf{n}$. Outputs are the pitch command $\hat{\beta}$ and the power command $\hat{P}$ which are determined by the main control functions. These commands may be altered by auxiliary control functions such as for tower damping.

In general, any controller such as the DTU Basic Wind Energy Controller (Hansen and Henriksen 2013) could be used. This would be acceptable for simulations in the time domain. However, a numerically smooth model is needed to fully exploit the functionalities of the STAS program, and this in most cases requires a custom implementation of the controller.

Chapter 3 describes a controller specifically designed for the STAS framework, in particular with a thought towards linear and probabilistic analyses.

2.6 Linking modules

Figure 7 shows a generic system consisting of two state-space modules that are to be coupled into a unified state-space; the extension to additional modules is straightforward, by recursive application of the two-module coupling procedure. This procedure relies on the condition that either $\mathbf{z}_{12}$ or $\mathbf{z}_{21}$ is

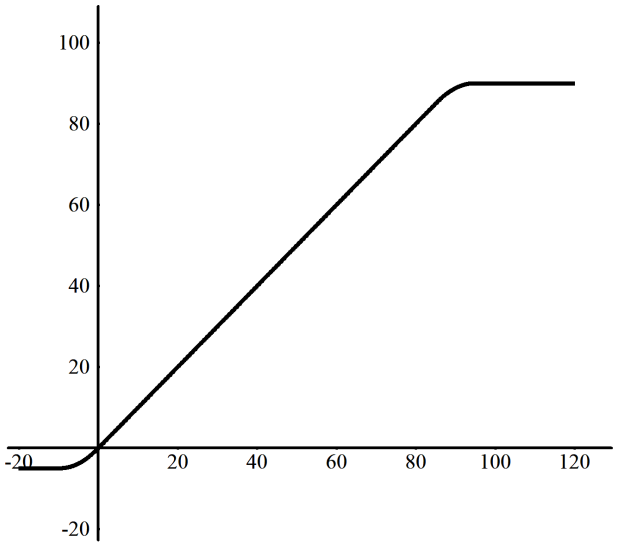


Figure 5: An example of pitch angle saturation at -5° and 90°.

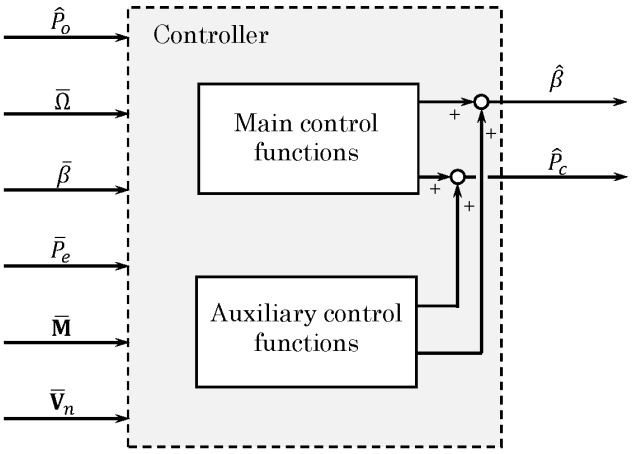


Figure 6: The interface of the control module.

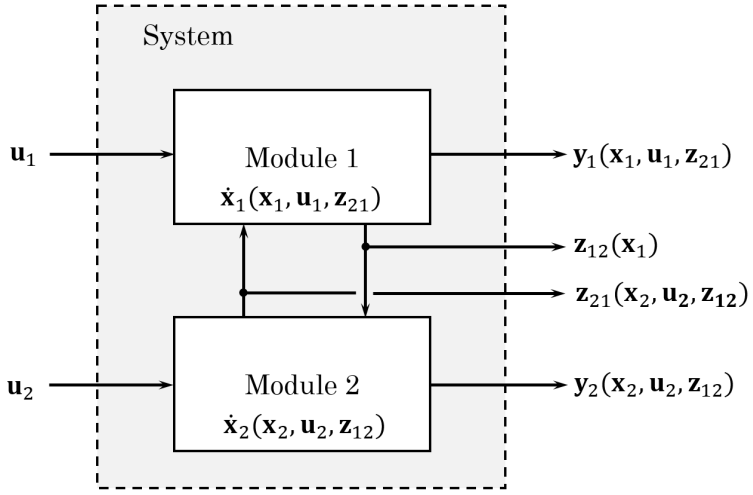


Figure 7: A generic system composed of two coupled state-space modules.

a function of only the states of its respective system, and not on the other set of interface variables. The present wind power plant model satisfies this condition.[4]

For the nonlinear state equations, the calculation is initialized by calling one of the modules – say, the first – with the initial states $\mathbf{x}_1$ and arbitrary inputs $\mathbf{u_1}$ and $\mathbf{z}_{21}$. Since there is no direct link between the inputs and interface variables, the output $\mathbf{z}_{12}$ will correctly match the initial condition of the states. Then, at each timestep, the second module is called to obtain the dynamic equation

$$\mathbf{N}_2(\mathbf{x}_2)\frac{d\mathbf{x}_2}{dt} = \mathbf{f}_2(\mathbf{x}_2, \mathbf{u}_2, \mathbf{z}_{12}) \tag{18}$$

and outputs $\mathbf{y}_2$ and $\mathbf{z}_{21}$. Finally, the first module is called again to obtain the dynamic equation

$$\mathbf{N}_1(\mathbf{x}_1)\frac{d\mathbf{x}_1}{dt} = \mathbf{f}_1(\mathbf{x}_1, \mathbf{u}_1, \mathbf{z}_{21}) \tag{19}$$

and outputs $\mathbf{y}_1$.

In the linear case, the combined equations are written in the form of (2) and (3). Omitting for the time being the $\mathbf{y}$ output variables,

$$\begin{bmatrix} \mathbf{N}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{N}_2 \end{bmatrix}_0 \frac{d}{dt}\begin{bmatrix} \Delta\mathbf{x}_1 \\ \Delta\mathbf{x}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{A}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_2 \end{bmatrix}\begin{bmatrix} \Delta\mathbf{x}_1 \\ \Delta\mathbf{x}_2 \end{bmatrix} + \begin{bmatrix} \mathbf{B}_{u1} & \mathbf{0} \\ \mathbf{0} & \mathbf{B}_{u2} \end{bmatrix}\begin{bmatrix} \Delta\mathbf{u}_1 \\ \Delta\mathbf{u}_2 \end{bmatrix} + \begin{bmatrix} \mathbf{0} & \mathbf{B}_{z1} \\ \mathbf{B}_{z2} & \mathbf{0} \end{bmatrix}\begin{bmatrix} \Delta\mathbf{z}_{12} \\ \Delta\mathbf{z}_{21} \end{bmatrix} \tag{20}$$

$$\begin{bmatrix} \Delta\mathbf{z}_{12} \\ \Delta\mathbf{z}_{21} \end{bmatrix} = \begin{bmatrix} \mathbf{C}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{C}_2 \end{bmatrix}\begin{bmatrix} \Delta\mathbf{x}_1 \\ \Delta\mathbf{x}_2 \end{bmatrix} + \begin{bmatrix} \mathbf{D}_{u1} & \mathbf{0} \\ \mathbf{0} & \mathbf{D}_{u2} \end{bmatrix}\begin{bmatrix} \Delta\mathbf{u}_1 \\ \Delta\mathbf{u}_2 \end{bmatrix} + \begin{bmatrix} \mathbf{0} & \mathbf{D}_{z1} \\ \mathbf{D}_{z2} & \mathbf{0} \end{bmatrix}\begin{bmatrix} \Delta\mathbf{z}_{12} \\ \Delta\mathbf{z}_{21} \end{bmatrix}. \tag{21}$$

The modules are linked by solving (21) for $\Delta\mathbf{z}$,

$$\Delta\mathbf{z} = (\mathbf{I} - \mathbf{D}_z)^{-1}(\mathbf{C}\,\Delta\mathbf{x} + \mathbf{D}_u\,\Delta\mathbf{u}), \tag{22}$$

and substituting into (20),

$$\mathbf{N}\frac{d\mathbf{x}}{dt} = \left[\mathbf{A} + \mathbf{B}_z(\mathbf{I} - \mathbf{D}_z)^{-1}\mathbf{C}\right]\Delta\mathbf{x} + \left[\mathbf{B_u} + \mathbf{B}_z(\mathbf{I} - \mathbf{D}_z)^{-1}\mathbf{D}_u\right]\Delta\mathbf{u}. \tag{23}$$

[4]One can always add, with care, high-frequency filters to make this so, although such an augmentation was not necessary with the present model.

2.7 Eliminating degrees-of-freedom

It is often desired to eliminate a selection of degrees-of-freedom from (1) through (3). For instance, one may wish to activate and deactivate different control functions; to make certain structural components rigid; or to study the modal dynamics of an isolated module, without programming customized wrapper code. In particular, the case of switching on and off different control functions is relevant here. It is convenient to build an initial model with *all* the control functions – that is, a fixed system architecture, such that degree-of-freedom n is always the same variable – and then, on the fly, disconnect those branches that are not in use.

The $\mathbf{A}$ matrix in (2) must be of full rank – or from a practical standpoint, well-conditioned numerically – in order to perform a Newton-Raphson solution for steady-state operating points (Chapter 4). Disconnecting some variables from others can easily lead to a situation where $\mathbf{A}$ is singular; all it takes is a single integrator without a feedback path to itself. The offending states must then be eliminated from the equations.

Consider (1), with the states partitioned into retained and slave variables, giving

$$\begin{bmatrix} \mathbf{N}_{rr} & \mathbf{N}_{rs} \\ \mathbf{N}_{sr} & \mathbf{N}_{ss} \end{bmatrix} \frac{d}{dt} \begin{bmatrix} \mathbf{x}_r \\ \mathbf{x}_s \end{bmatrix} = \begin{bmatrix} \mathbf{f}_r \\ \mathbf{f}_s \end{bmatrix}. \tag{24}$$

Eliminating a selection of degrees-of-freedom is as simple as partitioning the equations as in (24) and retaining only

$$\mathbf{N}_{rr}(\mathbf{x}_r, \mathbf{x}_{s0}) \frac{d\mathbf{x}_r}{dt} = \mathbf{f}_r(\mathbf{x}_r, \mathbf{x}_{s0}, \mathbf{u}, \mathbf{z}). \tag{25}$$

Note that (25) is not the same as the case where the partitioning and elimination is performed after taking the inverse, that is, striking rows from the equation

$$\frac{d}{dt} \begin{bmatrix} \mathbf{x}_r \\ \mathbf{x}_s \end{bmatrix} = \mathbf{N}^{-1} \begin{bmatrix} \mathbf{f}_r \\ \mathbf{f}_s \end{bmatrix}. \tag{26}$$

2.8 Multi-blade coordinate transform

A multi-blade coordinate (MBC) transform transforms the dynamics of the rotating wind turbine rotor into the fixed frame of the support structures, which, for balanced three-bladed rotors, eliminates the dependence on the rotor azimuth angle. The MBC transform is the same in principle as the d-q transform (12) and (13) employed in three-phase electrical systems. The details of the implementation are slightly different though. For the rotor, the transform is defined as

$$\mathbf{T}_B^{\psi} = \frac{1}{3} \begin{bmatrix} 1 & 1 & 1 \\ 2\cos\psi_1 & 2\cos\psi_2 & 2\cos\psi_3 \\ 2\sin\psi_1 & 2\sin\psi_2 & 2\sin\psi_3 \end{bmatrix}, \qquad \mathbf{T}_{\psi}^{B} = \begin{bmatrix} 1 & \cos\psi_1 & \sin\psi_1 \\ 1 & \cos\psi_2 & \sin\psi_2 \\ 1 & \cos\psi_3 & \sin\psi_3 \end{bmatrix} \tag{27}$$

for each trio of degrees-of-freedom – structural displacements and velocities, blade pitch commands, and so on – that are associated with the spinning rotor blades. The state equations transform as

$$\mathbf{T}_B^{\psi} \mathbf{N} \mathbf{T}_{\psi}^{B} \frac{d\mathbf{x}^{\psi}}{dt} = \mathbf{T}_B^{\psi}\, \mathbf{f}\left(\mathbf{T}_{\psi}^{B}\mathbf{x}^{\psi}, \mathbf{T}_{\psi}^{B}\mathbf{u}^{\psi}\right) - \Omega \mathbf{T}_B^{\psi} \mathbf{N} \frac{d\mathbf{T}_{\psi}^{B}}{d\Psi} \mathbf{x}^{\psi}. \tag{28}$$

3 Wind turbine control system for linear and probabilistic analyses

3.1 Introduction

A wind power plant is controlled with a hierarchy of functions. A practical control architecture consists of a central supervisory controller that sends a power command to each wind turbine.[5] The wind

[5] Alternatives that give direct authority over the turbines' pitch actuators and generator to a central controller, or on the other side allow clusters of turbines to intercommunicate and act independently as "intelligent agents" (Zhao *et*

turbine controller implements the commanded power, to the extent possible, while providing other necessary functions like keeping the rotor speed within limits, and communicating status back to the plant supervisory controller. It is therefore the *wind turbine* controller that directly governs the turbine dynamics; and the plant-level dynamics, though coordinated by the supervisory controller, emerge from the interactions between individual turbines. Also, the limitations in the rate and accuracy of power command tracking are mainly a consequence of the turbine-level controls. Thus the design of a good plant supervisory controller begins with a good wind turbine controller.

The primary function of the wind turbine controller is to keep the rotor speed within limits, while generating as much power as possible, within the capacity of the electrical system. Wind turbines are also required to provide grid support functions, involving, in one form or another, tracking active and reactive power commands sent by the plant operator.[6] Beyond this, a suite of secondary control functions damp drivetrain and support structure resonant modes, and reject moments due to asymmetry in the wind over the rotor plane.

The STAS framework places special requirements of numerical smoothness and precise gradients on the implementation of the control logic. In order to meet these requrements, a dedicated implementation of a wind turbine controller needed to be developed. At the same time, we have tried to look forward to future developments of the STAS program, in particular towards the probabilistic dynamics of wind power plants. This calls for an architecture with a minimal number of paths with nonlinear elements, and a minimal number of state variables sandwiched in between nonlinear functions. Our rotor speed controller here represents a first iteration, a starting point, and may be further streamlined in this regard. The key feature, though, is that it is built around a hard switch in the error signal fed to a single PI controller. If numerical smoothness and a hard switch seems like a contradiction – well, it is. The controller consists of two numerically-smooth systems, with a hard switch selecting between them.

The present control architecture draws on several references describing complete, industrial-type wind turbine controllers: Leithead and Connor (2000), Bossanyi (2000), van der Hooft *et al.* (2003), Hansen *et al.* (2005), Jonkman *et al.* (2009), Hansen and Henriksen (2013), Mulders and van Wingerden (2018), and Merz and Pedersen (2018). Figure 8 shows how the turbine control functions fit into the overall system architecture of a wind power plant. Of particular note are the use of an embedded model of the wind turbine as a state observer, in order to estimate the local wind speed and rotor thrust; and that the entire hierarchy of controls are explicitly modelled, including generator current control and, inside the electrical module, voltage control of the converters.

The thrust-hold, tower damping, and individual blade pitch control functions were not active in this study, and are omitted hereafter from the discussion.

3.2 Rotor speed control and power command tracking

Figure 9 shows the architecture of the rotor speed and power command tracking control function. This function is designed to fulfil the following high-level objectives:

1. When the wind speed is sufficient to provide the commanded power, then do so, and employ the blade pitch system to hold the rotor speed at its target value.

2. When the wind speed is insufficient to provide the commanded power, then provide the maximum possible power.

The transition between these regimes is handled by a hard switch, designated by the logical variable μ in Fig. 8. This does not need to be a hard switch; the DTU Basic Wind Energy Controller (Hansen

al. 2012), are theoretically possible. However, a practical controller must respect the system architecture of present wind power plants, and the division of responsibility between OEMs, developers, and operators.

[6]The operator may be a human setting a curtailed level of power, or an automatic control system acting on frequency or voltage droop.

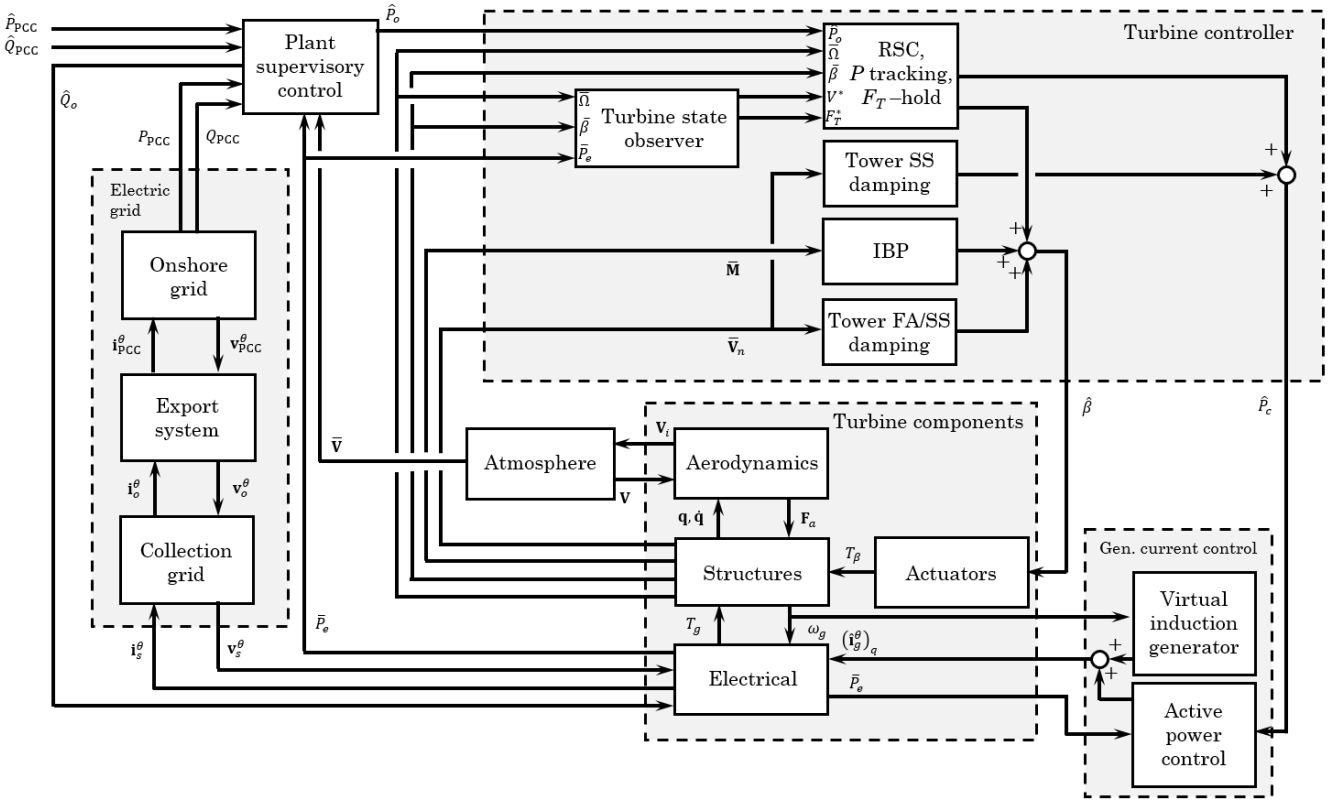


Figure 8: The system architecture of a wind power plant, with particular emphasis on the turbine control functions.

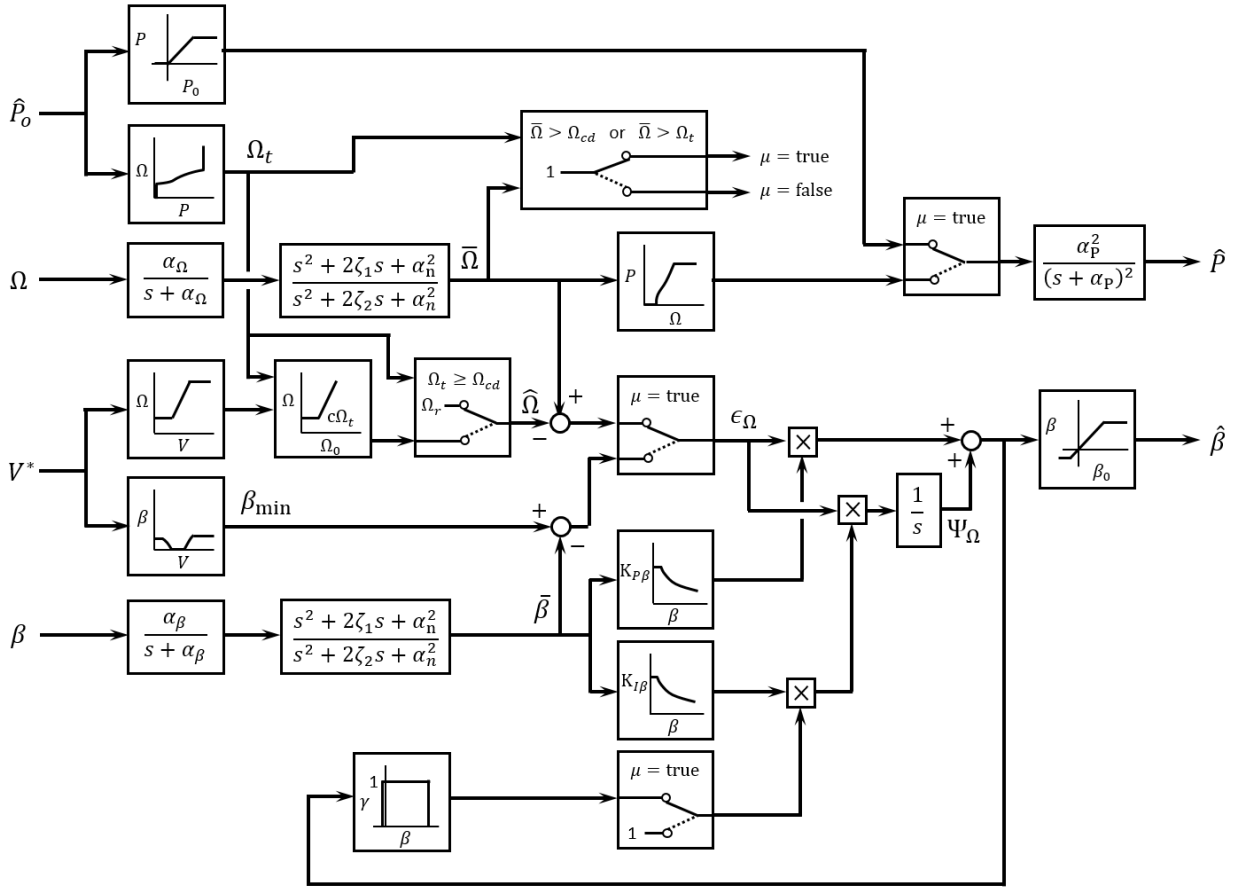


Figure 9: The architecture of the rotor speed control function.

and Henriksen 2013) in particular is an example of a controller that uses smooth scaling of saturation bounds in order to achieve the same ends. Why the hard switch, then? In the future, we envision performing probabilistic analyses where the system on each side of the hard switch is approximated as linear – in fact, where the hard switch is the only remaining nonlinearity. Here a hard switch has the great benefit that it has only two settings, rather than a continuum: this has the potential of greatly reducing the dimensionality of a probabilistic description of the system. Having said this, probabilistic analysis is not our present concern: and so in this chapter we shall evaluate the controller in terms of its performance according to the usual criteria.

Let us walk the paths through the controller in turn, using a tuning for the DTU 10 MW wind turbine (DTU-WT) as an example. The raw inputs are an operator power command $\hat{P}_o$ – in Fig. 8 this is seen to come from the wind plant supervisory controller – the rotor speed Ω, blade pitch β, and some estimate of the effective wind speed, V^*. Outputs are a power command[7] $\hat{P}$, and a collective pitch angle $\hat{\beta}$, "collective" referring to the fact that the command is applied to all three blades equally.

The most important feature of the controller is the switch, so we will deal with that first. The main element between the input $\hat{P}_o$ and the switch is an inverted power-speed curve. The relationship between rotor speed Ω and electric power P_e under normal operation is prescribed, as shown in Fig. 10 for the case of the DTU-WT in particular. This can be divided into operating regions, which we here denote (a) through (d) for convenience.[8] Region (b) is a region of operation at the design tip-speed

[7]This could just as well be a torque command; power and torque differ by a factor of Ω, and a measurement of this is available to the controller.

[8]These are also known as, respectively, Regions 1.5, 2, 2.5, and 3.

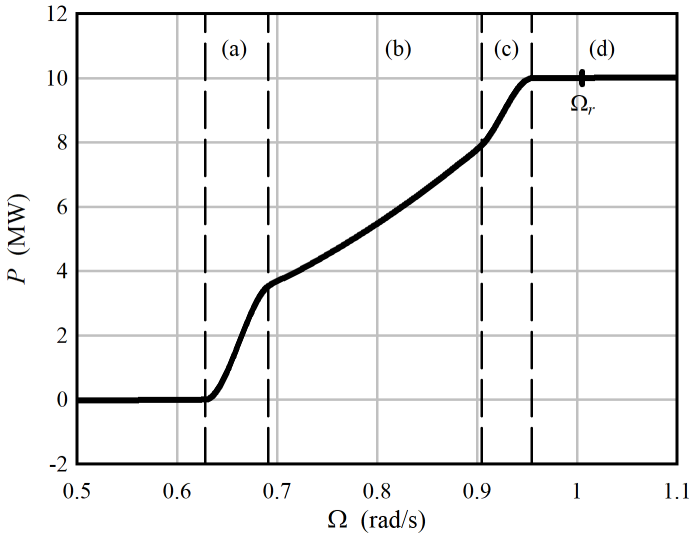


Figure 10: The schedule of electrical power as a function of rotor speed, for the DTU-WT.

ratio,

$$P_e = C_P^* \eta \frac{\pi}{64} \overline{\rho} \frac{D^5}{(\lambda^*)^3} \Omega^3 = K_\Omega(\eta, \overline{\rho})\ \Omega^3, \tag{29}$$

where C_P^* is the power coefficient at the design tip speed ratio λ^*, A is the rotor area, D is the rotor diameter, $\overline{\rho}$ is the measured air density, and η is the efficiency of the drivetrain, from the aerodynamic power to the electrical power, at whichever location (generator, transformer) this is being measured. Region (d) is the rated power. Regions (a) and (c) are transitional, both serving the purpose to constrain the rotor speed. Region (a) in effect prevents the rotor speed from dropping below its lower bound. Region (c) is set such that the power will reach its maximum at a speed somewhat below the rated speed Ω_r. This provides some margin for the pitch controller, such that the speed can depart slightly from the rated speed without fluctuations in the electrical power. In the STAS implementation, the curve of Fig. 10 is input as a list of tabulated values, and piecewise cubic Hermite splines are used to generate a smooth curve with a continuous first derivative.

Now, the curve in Fig. 10 can be inverted,[9] and with the operator power command $\hat{P}_o$ as input, the resulting Ω_t (Fig. 9) defines the rotor speed at which the transition from maximum-C_P to power command tracking should take place. The rotor speed Ω_{cd}, at the boundary between Regions (c) and (d), serves as an upper bound on this transition speed. When the measured rotor speed $\overline{\Omega}$ is above the transition speed, then the turbine is put into power command tracking mode and the pitch controller is active. When $\overline{\Omega}$ is below the transition speed, then the pitch is set to its minimum angle and the power is computed directly from $\overline{\Omega}$, using Fig. 10. This is the essence of the switch μ, seen in Fig. 9.

This essentially completes the description of the portion of the controller associated with the power command $\hat{P}$: either $\hat{P}$ tracks the speed-power profile, or it tracks the operator power command, depending on the status of the switch. It remains to say only that the measured speed $\overline{\Omega}$ is low-pass filtered, and also notch filtered at the first tower side-to-side frequency, to prevent this from being excited by the generator.

The pitch controller is a bit more complicated. When the generator is operating according to the (Ω, P) schedule (μ = false), the pitch angle is driven to its minimum value. The minimum pitch angle is a prescribed function of the wind speed, as in Fig. 11 for the DTU-WT. Here the profile at low wind speeds is set such that, in combination with Region (a) of Fig. 10, the rotor follows the maximum

[9]Though not readily visible in Fig. 10, Region (d) and the region to the left of (a) are given a small slope, both to aid Newton-Raphson iteration, and to make the inverse function defined for all P.

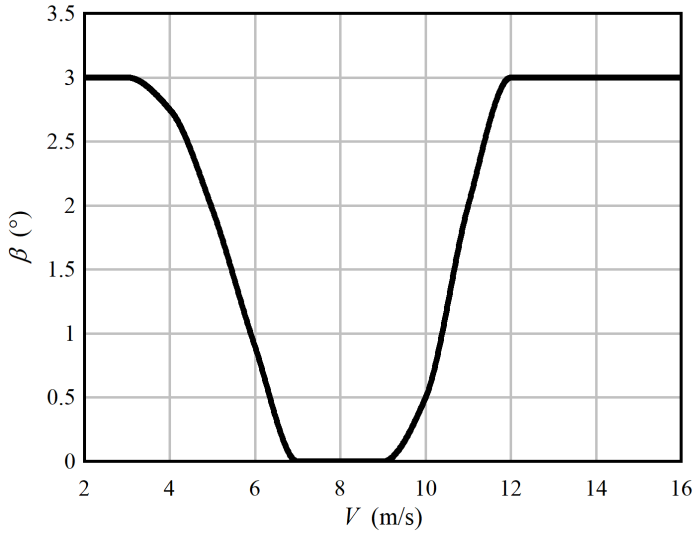


Figure 11: The minimum pitch angle, as a function of wind speed, for the DTU-WT.

off-design C_P. At wind speeds approaching rated (roughly 11.5 m/s), the blades are made to pitch a couple degrees in order to "prime" the aerodynamic conditions prior to activating the pitch controller. There are two reasons for doing this: first, the sensitivity in aerodynamic power to pitch angle, $\partial P/\partial\beta$, is by definition zero at the point of maximum C_P; therefore by pitching a couple degrees and leaving the maximum C_P point, some $\partial P/\partial\beta$ sensitivity is gained, making the pitch controller more effective. Second, pitching early can significantly reduce the peak aerodynamic thrust load on the rotor, which occurs, if no mitigating action is taken, just at the rated wind speed.

Let now the wind speed by sufficiently high that the condition $\overline{\Omega} > \Omega_t$ is met, and μ = true. Now the generator is tracking the operator power command $\hat{P}_o$ – or else the rated power P_r – and the rotor speed is being controlled by the blade pitch. The error between the measured and target speeds, ε_Ω is fed through a gain-scheduled PI controller, and this gives the pitch command. The gains are scheduled as a function of the blade pitch angle, for reasons described by Hansen *et al.* (2005). Figure 12 shows an example of the gains used in the present work for the DTU-WT. These are tuned to give a natural frequency of around 0.1 Hz and damping ratio of about 0.4 under normal operating conditions.

The target rotor speed is set to a value that is according to a prescribed relationship with the wind speed, Fig. 13. When operating in power command tracking mode, the rotor speed is not unique: one can operate along any isoline of $C_P(\lambda, \beta)$. For instance, Fig. 14 at right indicates potential envelopes of operation, if the rotor speed is to be constrained between the nominal and rated values. We choose, however, to operate with a consistent relationship between the rotor speed and wind speed – in other words, Fig. 13 is the nominal power-speed operating schedule of Fig. 10, mapped through the C_P characteristics of the rotor to arrive at a corresponding wind speed.[10] Operating in this manner – as opposed to allowing the rotor speed to fluctuate over wider bounds – eases the transition at the hard switch between control modes.

The final feature to mention regarding the rotor speed controller of Fig. 9 is the anti-windup. If the pitch saturates at either its upper or lower limit, then the integrator is stopped by setting the gain $K_{I\beta}$ to zero. The anti-windup is disabled when in maximum-C_P mode, and the pitch is being driven to its minimum bound. This allows the controller to operate normally, with near-linear dynamics, as β tracks the fluctuating $\beta_{\min}$.

[10]The minimum target speed, for purposes of pitch control, is set somewhat above the nominal cut-in speed of 0.628 rad/s.

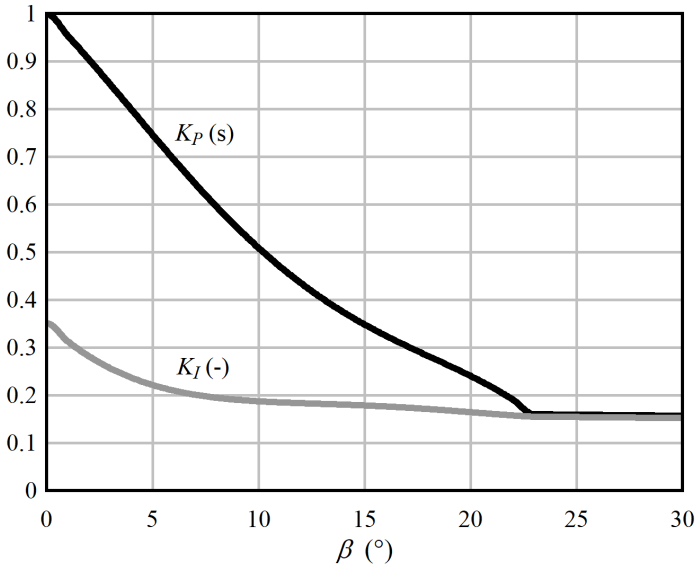


Figure 12: The schedule of proportional and integral gains of the blade pitch controller, for the DTU-WT.

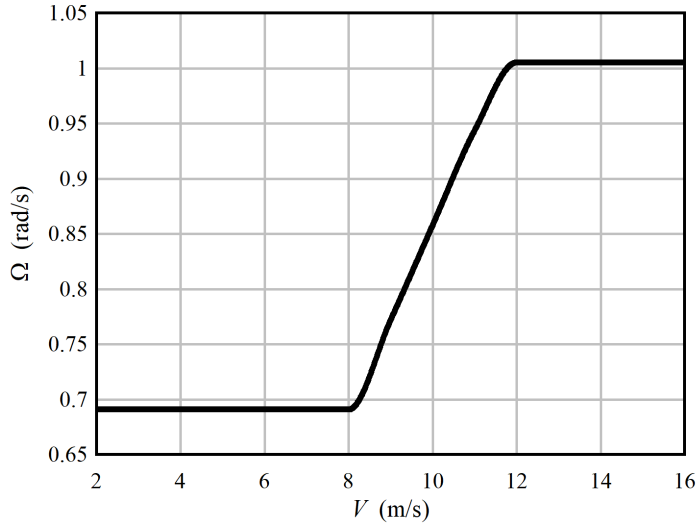


Figure 13: The schedule of rotor speed as a function of wind speed, for the DTU-WT.

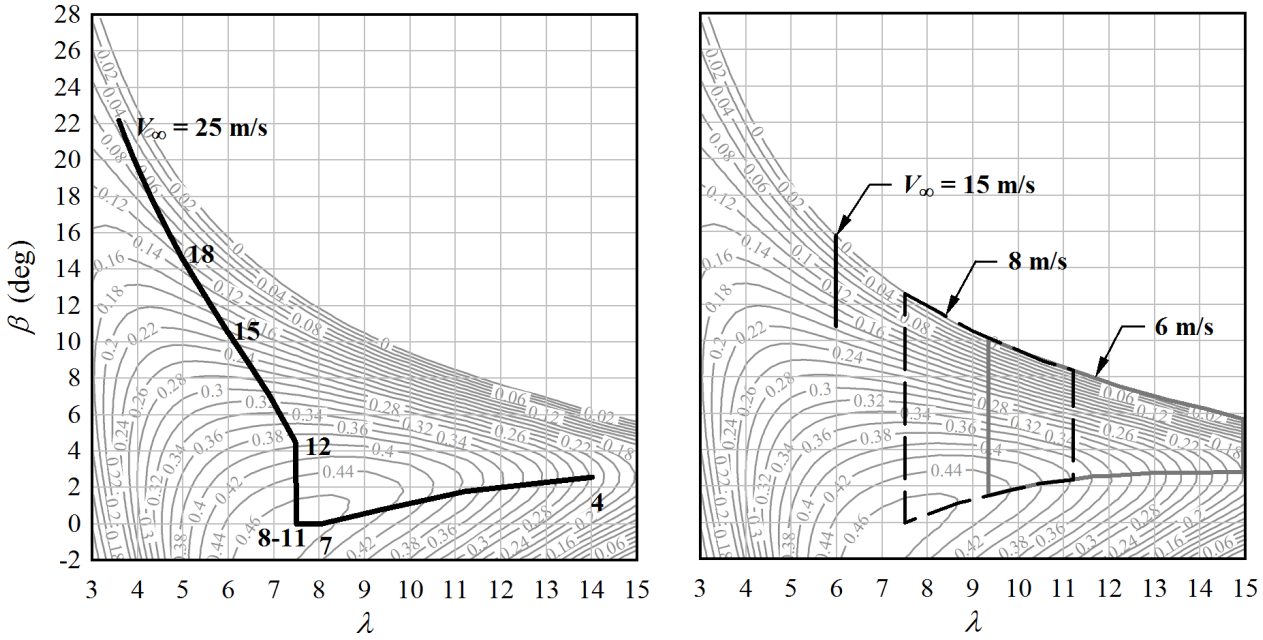


Figure 14: The $C_P(\lambda, \beta)$ characteristics of the DTU-WT. At left, the nominal operating schedule is overlaid; Fig. 13 is extracted along this curve. At right, envelopes of operation between the nominal and maximum rotor speeds are shown, for a selection of wind speeds.

3.3 Active power control and virtual induction generator

Following Fig. 8, the first step along the line for the electrical power command $\hat{P}$, output from the rotor speed controller of Section 3.2, is the generator current controller. This consists of two blocks, as shown in Fig. 15. The primary block, the active power controller, measures the electric power at some convenient point like the generator or wind turbine transformer terminals,[11] and adjusts the operation of the generator until it meets the commanded power. The secondary block makes a synchronous generator mimic an induction generator, with torque changing in proportion to speed, over a particular frequency band. It takes a measured rotor speed as input, and applies an in-phase resistive torque at the frequency of driveshaft resonance; this helps to damp driveshaft and blade edgewise resonant oscillations.[12]

The q-axis generator current is directly proportional to the air gap torque, if the generator is of the permanent magnet synchronous type, so Fig. 15 is essentially a torque controller.

3.4 Wind speed observer

The rotor speed controller of Section 3.2 requires a slowly-varying estimate of the wind speed, in order to set appropriate target values for rotor speed or blade pitch. This is not obtained by measuring the wind directly, due to problems with variability and reliability of anemometer measurements. Rather,

[11] Here we assume the latter: we define the electrical power of the turbine as that output at the point where the turbine is connected to the collection grid. This then compensates for the unknown losses in the converters and transformer by the feedback loop through the active power controller. In certain emergency and fault situations, it might be better to switch back to measuring the power at the generator, although in any case, if the fault is severe then the turbine must enter an emergency mode where the particular algorithms of this chapter no longer apply.

[12] On direct-drive turbines with a very stiff shaft, the "driveshaft torsion" mode may consist primarily of the generator resonating against the inertia of the outer portion of the aerodynamic rotor, with the blades, rather than the driveshaft itself, providing the majority of the flexibility. In any case, driveshaft torsion will be coupled with collective edgewise deflection of the blades, which is poorly damped by aerodynamic forces.

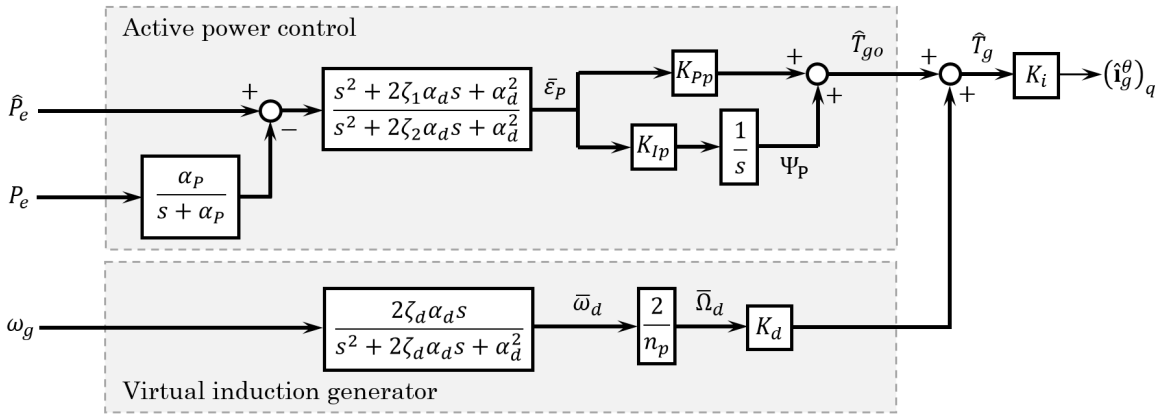


Figure 15: The architecture of the generator current controller, consisting of an active power control loop and a virtual induction generator control loop.

a wind speed observer is constructed from a model of the wind turbine, and this is used together with reliable measured quantities – rotor speed, blade pitch, and electrical power – to estimate an effective rotor-average wind speed for use in control. The same observer can also be used to provide an estimate of the aerodynamic thrust, which is used in the controller's thrust-hold mode; though we leave the control of rotor thrust outside the present scope.

The wind speed observer is based around a rigid, rotating-shaft model of the wind turbine rotor, together with the aerodynamic C_P characteristics of Fig. 14. Based on the equation of motion

$$J\frac{d\Omega}{dt} = T_a - T_g = C_P(V,\Omega,\beta)\ \frac{1}{2}\overline{\rho}A\frac{V^3}{\Omega} - \frac{P_e}{\eta(P_e)\ \Omega}, \tag{30}$$

an observer is constructed,

$$\frac{d}{dt}\begin{bmatrix}\Omega^*\\ V^*\end{bmatrix} = \begin{bmatrix} C_P(V^*,\Omega^*,\overline{\beta})\ \frac{1}{2}\overline{\rho}A\frac{(V^*)^3}{J\Omega^*} - \frac{\overline{P}_e}{\eta(\overline{P}_e)\ J\Omega^*} \\ 0 \end{bmatrix} + \begin{bmatrix}K_W\\ K_V\end{bmatrix}(\overline{\Omega}-\Omega^*). \tag{31}$$

Here A is the rotor swept area, J is the rotor inertia, and Ω^* and V^* are the observed quantities. The archictecture (31), and some alternatives with minor benefits, are described by Østergaard (2007).

The wind speed observer is simple enough to be tuned manually. Tuning automatically as an extended Kalman filter is fine, but not really necessary here. For the DTU-WT, gains of $K_W = 2$ and $K_V = 15$ m give satisfactory performance and stability margins.

3.5 Controller performance

A well-performing controller fulfils its objectives robustly, maintaining performance under severe inputs, and when the dynamic properties of the system do not exactly match those assumed during the design and tuning process. The linear (small-signal) performance and stability characteristics of Regions (b) and (d) of the rotor speed controller have been thoroughly investigated, and are documented in Merz and Pedersen (2018). In short, when the controller is coupled to a full open-loop model of a wind turbine, including aeroelastic and electrical dynamics, a single eigenmode[13] appears that

[13] Actually, when the mode is oscillatory, it is strictly speaking two complex-conjugate eigenmodes, one each at the positive and negative frequencies. It is most convenient and meaningful to think of this as "one mode." It is actually quite fantastic, that although the model contains hundreds of degrees-of-freedom, we can observe a single mode when tuning the pitch control gains!

contains the primary closed-loop response of the pitch control of rotor speed. The properties of this mode can be used to tune the controller and schedule the gains. The gain scheduling of Fig. 12 gives a frequency of around 0.1 Hz, and a damping ratio of around 0.4, which represents a nice compromise between speed of action and stability. As for the variable-speed operating regime (b), it is shown in Merz and Pedersen that the nonlinear control algorithm (29) has good stability characteristics, and near-optimal performance.

Our focus here is on the nonlinear elements of Fig. 9, above all the behavior during switching from one control mode to the other. The performance of a nonlinear dynamic system is difficult to verify, since one cannot extrapolate the response under one condition to another condition, nor apply the principle of superposition to look at the system input-by-input or frequency-by-frequency, in the same way that one can with a linear system. We take the strategy of looking at step functions in the wind speed and power command inputs. Step functions of small amplitude reveal much about the small-signal stability, since the step excites all the resonant modes – it is something of a worst-case input – and the amount of overshoot and damping are immediately evident. A large step forces the system to traverse the entire operating range, at a limiting rate determined by the system, rather than the input.

For this evaluation the rotor speed controller of Section 3.2 was implemented in the wind turbine simulation program Ashes (Thomassen *et al.* 2012). Ashes contains a full aeroelastic model of the turbine, including flexible blades, drivetrain, and tower, and will reveal any resonant interactions between the controller and structures. The pitch actuator response was incorporated into the controller, and was modeled as a low-pass filter. The equations implemented in Ashes are similar in scope to those of the aeroelastic module of STAS; we elected to use Ashes because it is optimized for the time-domain simulation of single-turbine aeroelastic load cases, and for this purpose runs much faster. Also, these results from Ashes provide a basis for comparison against mixed linear/nonlinear models, Section 3.6.

We have used the 10 MW Offshore Reference Turbine from Anaya-Lara *et al.* (2018) as a case. This turbine is based on the DTU 10 MW rotor, and includes a detailed model of a direct-drive 10 MW generator, nacelle structures with a shaft and bearing arrangement for the direct-drive generator, and an offshore monopile support structure in 30 m water depth.

Figures 16 through 19 show a series of cases that exercise the primary functions of the controller, namely stabilizing the rotor speed in various operating regimes; and tracking, or maximizing, power generation. The response of the system to fluctuations in the wind speed is illustrated in Fig. 16, where the wind speed is progressively stepped across the operating range, from cut-in through rated and above; and then back down. The plots at top show the incoming wind speed, as well as the estimate V^* from the wind speed observer. The estimate is seen to track the actual value with a timescale of roughly 30 s to 1 minute, with a slower response at lower wind speeds; this is not surprising, since the primary input to the observer is the rotor speed, and this changes sluggishly in low winds, where the aerodynamic forces are small.

The succeeding rows of plots show respectively the rotor speed, collective blade pitch, and electrical power. These are just as intended: below rated, the pitch passively tracks the target based on the estimated wind speed, while the generator governs the rotor speed, at first tightly in the vicinity of the minimum operating speed (0.63 rad/s), and then tracking the maximum C_P. The response is seen to be relatively rapid, with minimal overshoot, and high damping: the hallmarks of a good controller. Crossing the rated wind speed between 11 and 12 m/s, the control mode switches: μ goes from false to true, referring to Fig. 9. Although the switch is abrupt, this is barely visible in Fig. 16. On switching, the electrical power tracks its rated value $P_r = 10$ MW, while the blade pitch acts to track the rated rotor speed Ω_r. The response under pitch control is also rapid, with minimal overshoot and high damping.

Figure 17 shows an absurdly severe case, where the wind speed jumps instantaneously from 5 to 15 m/s, and then back. On the way up, there is a significant 20% overshoot in the rotor speed; and in fact, in the period between 30 and 34 s, the blade pitch initially takes an incorrect action, pitching

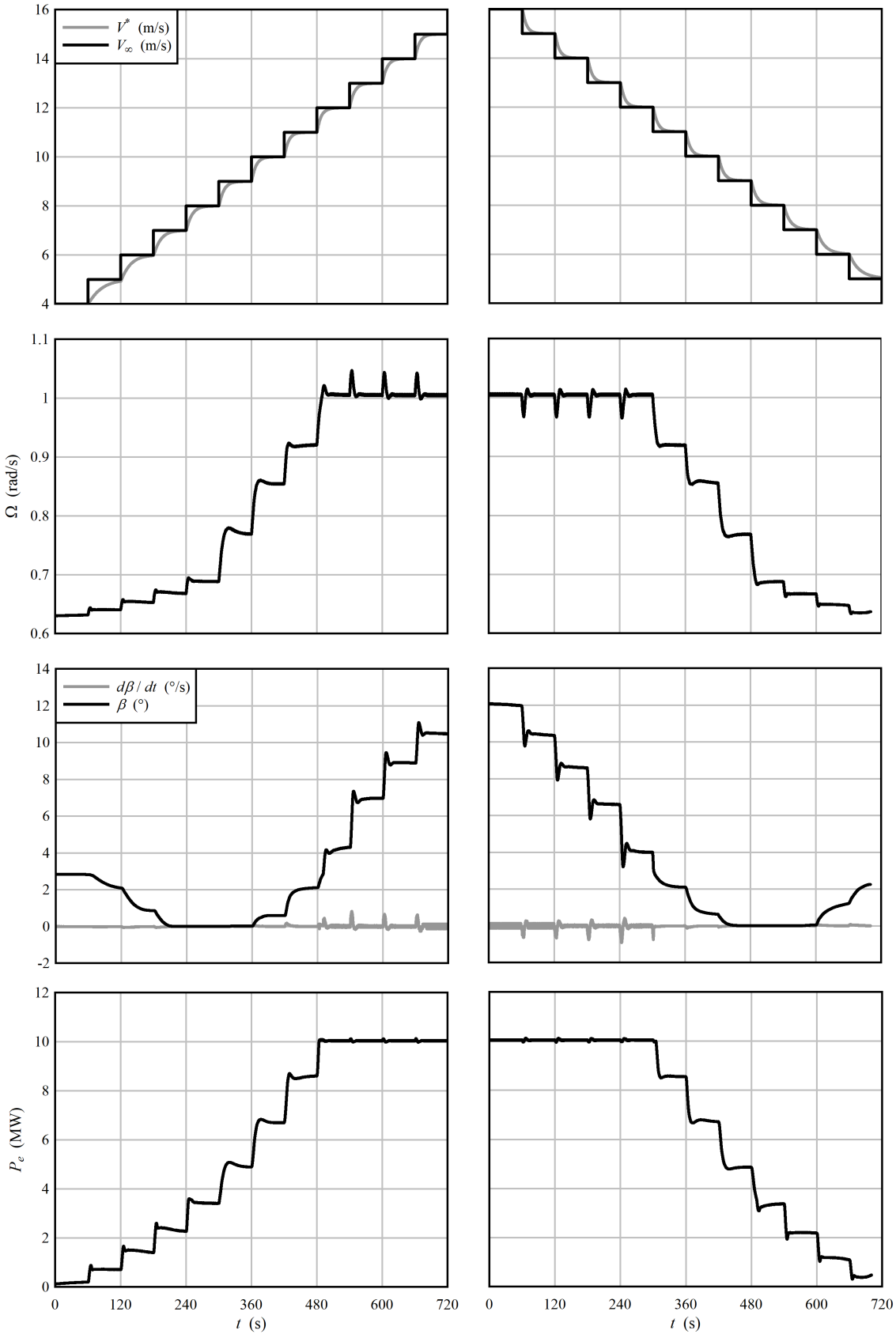


Figure 16: The closed-loop response to step changes in wind speed. Column at left: wind speed increasing, column at right: wind speed decreasing.

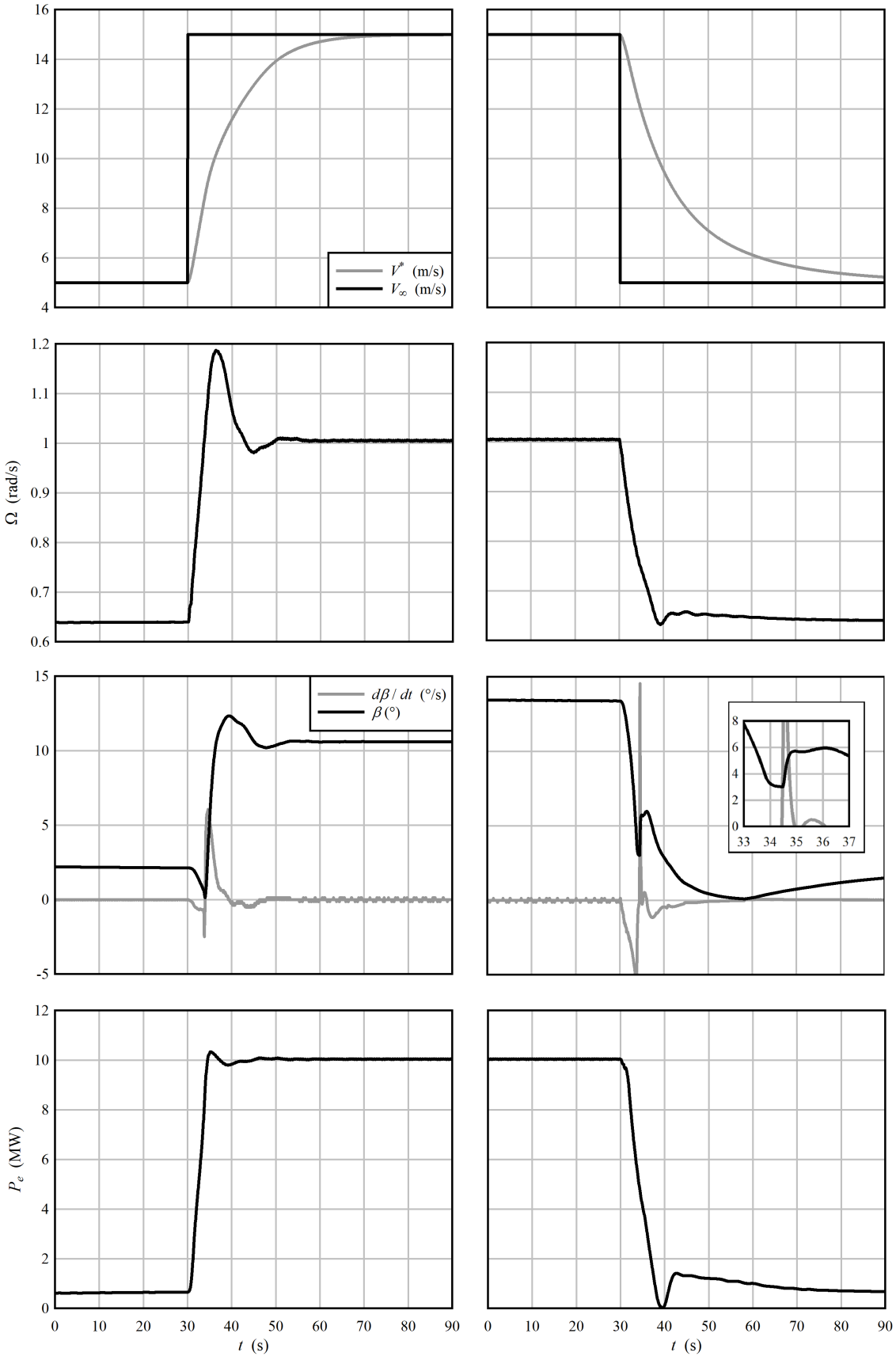


Figure 17: The closed-loop response to a severe and sudden jump in the wind speed. Column at left: wind speed increasing, column at right: wind speed decreasing.

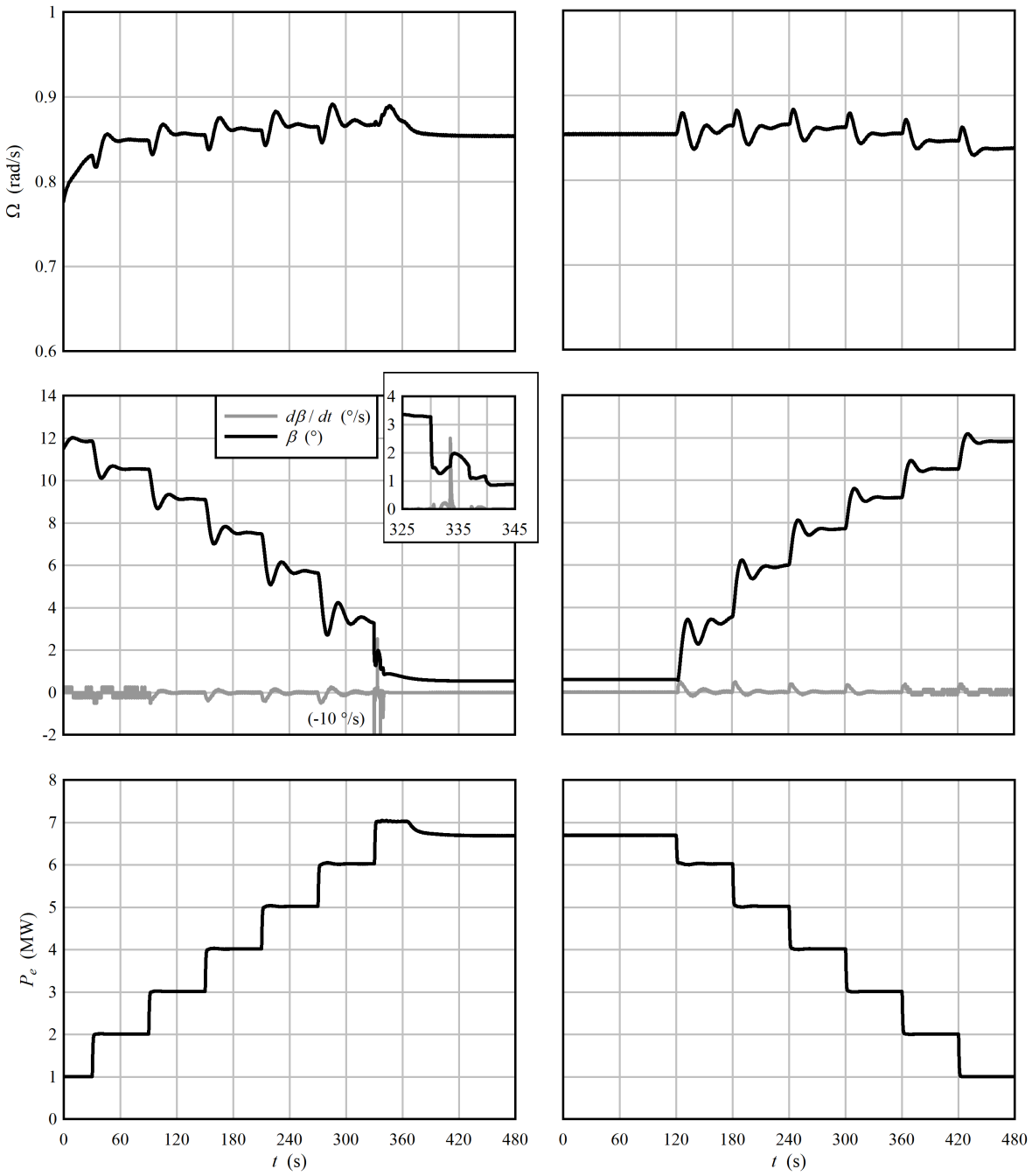


Figure 18: The closed-loop response to step changes in power command. Column at left: power increasing, column at right: power decreasing. The wind speed is 10 m/s.

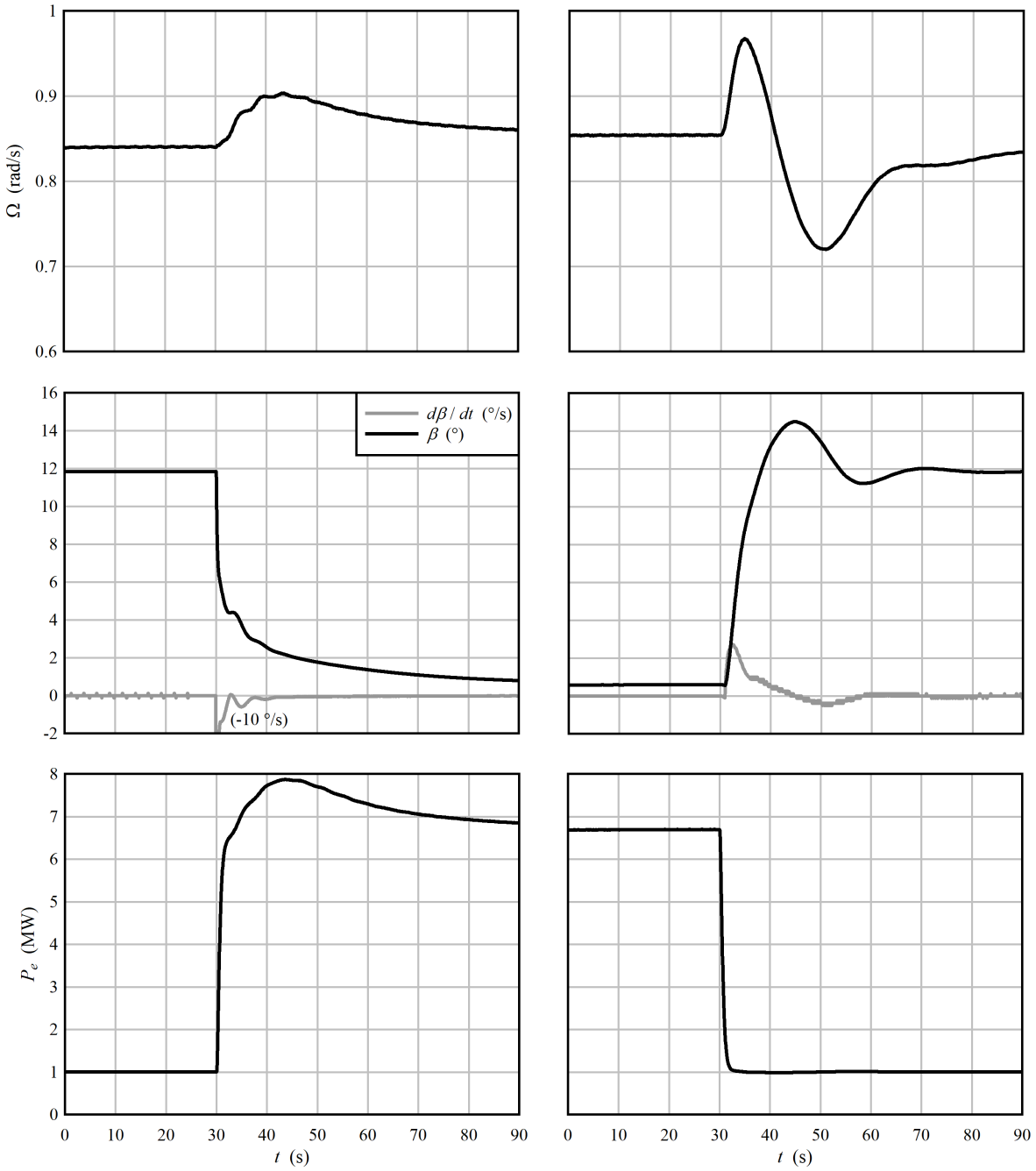


Figure 19: The closed-loop response to a sudden change in the power command from near idle to rated and vice versa. Column at left: power increasing, column at right: power decreasing. The wind speed is 10 m/s.

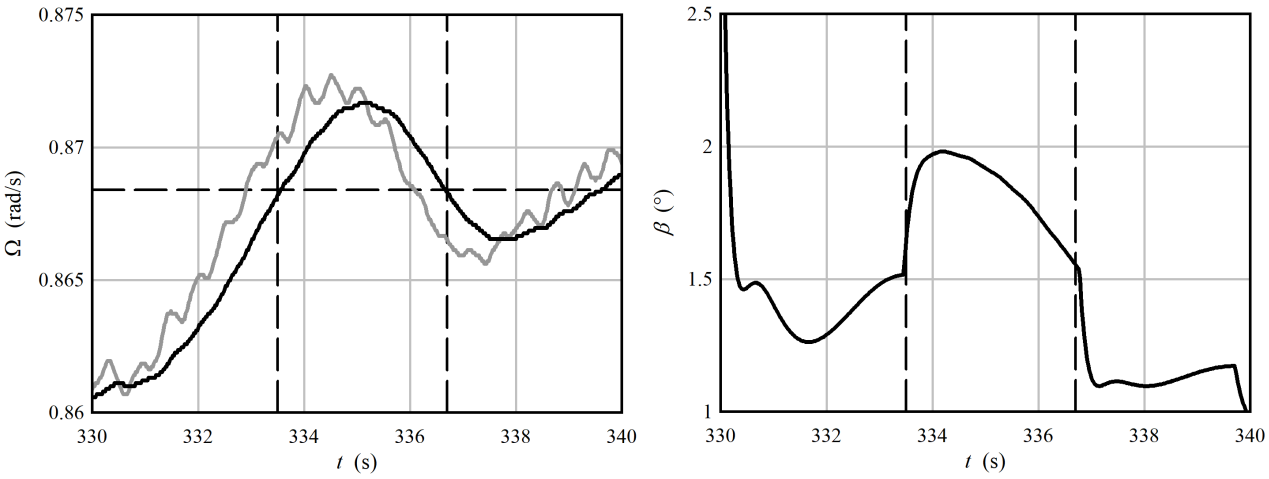


Figure 20: A close look at the behavior of the pitch system under switching initiated by a step in the power command.

towards zero, when it would ideally pitch to feather and begin to arrest the rotor. This initial pitch response follows, as intended, the prescribed function of V^* (Fig. 11); it does not feather until the rotor speed crosses the Ω_{cd} boundary that activates the switch in control modes. In the plots at right, where the wind speed abruptly drops, the blade pitch is also observed to undergo an abrupt reversal. Here the initial response is that of the actuator to the step change in the error ϵ_Ω, via the proportional gain $K_{P\beta}$ pathway. This then recovers to a slower decrease governed primarily by the integral $K_{I\beta}$ pathway, together with the filtering on the measured pitch angle $\overline{\beta}$. We have included a low-pass filter on the pitch command output $\hat{\beta}$, like that on the power command $\hat{P}$, which can limit the pitch rate in such cases. This filter was not active when these results were obtained.

The other primary function of the controller is to track, to the extent possible, an operator power command. The performance in this regard is shown in Fig. 18, where the commanded power has been stepped from 1 MW through 10 MW, in a constant 10 m/s wind. So long as the available power at this wind speed is greater than the commanded $\hat{P}$, then the rotor speed is stabilized nicely by the pitch system. (Some effects of a startup transient are visible at the beginning of the left-hand plot of rotor speed.) The transition rotor speed Ω_t, not shown, steps along with $\hat{P}$, and at 330 s this crosses the measured rotor speed, initiating a switch in the control mode. The response to this which is a rapid reduction in the pitch angle to the minimum, which is indeed the appropriate response. However, immediately afterwards there is another cycle of switching between the two modes.[14]

Figure 20 focuses on the period of time around the switching. On the left is a plot of the rotor speed, both the raw value from the structural model (gray), and the filtered value used by the controller (black). Also shown on this plot is a horizontal line which indicates the new transition speed Ω_t after the step change in the power command. It is clear why there is an additional switching cycle: the updated value of Ω_t – which is purely a function of the power command $\hat{P}$ – lands just above the actual rotor speed; so that when the blades pitch out, and the rotor accelerates a little bit, the switch is re-crossed. Two vertical lines in the left-hand plot indicate where the measured speed crosses Ω_t, which is where the switching occurs. The consequence of the switch is an increase in the pitch rate, although the travel of about 0.5° is small.

In summary, the controller performs its objectives of tracking power commands and limiting the rotor speed nicely. The consequence of employing a hard switch between control regimes is that an

[14]This is always the danger with a hard switch, and hysteresis could be added to eliminate the problem without much further thought. But this is not an option here, where we aim to minimize the complexity of the nonlinear logic in the controller.

increase in the pitch rate may occur at the time of the switch.[15] Whether this is acceptable or not depends on the purpose of the analysis. It could be problematic if a given study were focused on the pitch rate as a primary metric of performance. On the other hand, if overall pitch travel (the integral of the absolute rate) were used as a metric of pitch activity, then the performance would be essentially unaffected. It should also be noted that the present control architecture is but one of several available, and alternatives can also, without much difficulty, be made numerically smooth and incorporated into the STAS software.

3.6 A further look at control mode switching, and efficient nonlinear analysis

We envision a wind power plant controller that acts on real-time probabilistic estimates of the future state. Extrapolating probability into the future requires either simplifying assumptions about the nature of the probability distributions – a linear system with Gaussian probabilities is the most common example – or else some way to limit the dimensionality of the joint probabilities that must be evaluated, be it working with low-order approximations of the system, or exploiting special features of the system architecture.

Our hypothesis is that a fair approximation of the stochastic response of a wind turbine, and by extension a wind power plant, can be obtained with a mixed linear/nonlinear dynamic system: in particular, one with a large number of states that evolve according to linear equations, and a small number of isolated nonlinear elements. The extreme version of this concept would be a switched linear system – that is, a single two-position switch selecting between two linear system configurations. This has been the motivation for developing the controller in its present form.

So let us take a look at how we can arrive at such a switched linear system. The starting point is a Newton-Raphson solution of a steady-state operating point, Chapter 4. After the solution for the state vector

$$\mathbf{x}_0 = \begin{bmatrix} \mathbf{x}_{s0} \\ \mathbf{x}_{c0} \end{bmatrix}$$

is completed, the linearized matrices $\mathbf{N}$, $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, and $\mathbf{D}$ from the last iteration are stored. These are the *closed-loop* matrices: embedded within them is a linearized version of the controller at the operating point. We no longer want this controller, and want to swap it out for its nonlinear sibling. Beginning with

$$\begin{bmatrix} \mathbf{N}_{ss} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \frac{d}{dt} \begin{bmatrix} \Delta\mathbf{x}_s \\ \Delta\mathbf{x}_c \end{bmatrix} = \begin{bmatrix} \mathbf{A}_{ss} & \mathbf{A}_{sc} \\ \mathbf{A}_{cs} & \mathbf{A}_{cc} \end{bmatrix} \begin{bmatrix} \Delta\mathbf{x}_s \\ \Delta\mathbf{x}_c \end{bmatrix} + \begin{bmatrix} \mathbf{B}_s \\ \mathbf{B}_c \end{bmatrix} \Delta\mathbf{u}, \tag{32}$$

$$\Delta\mathbf{y} = \begin{bmatrix} \mathbf{C}_s & \mathbf{C}_c \end{bmatrix} \begin{bmatrix} \Delta\mathbf{x}_s \\ \Delta\mathbf{x}_c \end{bmatrix} + \mathbf{D}\Delta\mathbf{u} \tag{33}$$

the equations associated with the controller states are removed, and the perturbations to the states $\mathbf{x}_c$ are recast as inputs,

$$\mathbf{N}_{ss} \frac{d\Delta\mathbf{x}_s}{dt} = \mathbf{A}_{ss}\,\Delta\mathbf{x}_s + \begin{bmatrix} \mathbf{B}_s & \mathbf{A}_{sc} \end{bmatrix} \begin{bmatrix} \Delta\mathbf{u} \\ \Delta\mathbf{x}_c \end{bmatrix} \tag{34}$$

or

$$\frac{d\Delta\mathbf{x}_s}{dt} = \mathbf{N}_{ss}^{-1}\mathbf{A}_{ss}\,\Delta\mathbf{x}_s + \mathbf{N}_{ss}^{-1} \begin{bmatrix} \mathbf{B}_s & \mathbf{A}_{sc} \end{bmatrix} \begin{bmatrix} \Delta\mathbf{u} \\ \Delta\mathbf{x}_c \end{bmatrix} = \tilde{\mathbf{A}}\Delta\mathbf{x}_s + \tilde{\mathbf{B}} \begin{bmatrix} \Delta\mathbf{u} \\ \Delta\mathbf{x}_c \end{bmatrix}, \tag{35}$$

and

$$\Delta\mathbf{y} = \mathbf{C}_s\Delta\mathbf{x}_s + \begin{bmatrix} \mathbf{D} & \mathbf{C}_c \end{bmatrix} \begin{bmatrix} \Delta\mathbf{u} \\ \Delta\mathbf{x}_c \end{bmatrix} = \mathbf{C}_s\Delta\mathbf{x}_s + \tilde{\mathbf{D}} \begin{bmatrix} \Delta\mathbf{u} \\ \Delta\mathbf{x}_c \end{bmatrix}. \tag{36}$$

[15] Compared with the NREL rotor speed controller (Jonkman *et al.* 2009), what we have done is to place the burden associated with the hard switch on the pitch system, as opposed to the generator. When the switch occurs, our controller may give a step command do the pitch system, whereas the NREL controller may give a step command to the generator. There are tradeoffs either way, and in either case the transition could be smoothed if we were to allow a more complex switching logic. When it comes to the study of *plant control*, we prefer to retain full authority over the generator power: the blade pitch is a local effect, whereas the generator power has a global influence.

This is to be coupled to the nonlinear controller

$$\frac{d\mathbf{x}_c}{dt} = \mathbf{f}(\mathbf{x}_c, \mathbf{x}_s, \mathbf{u}) \tag{37}$$

for time-domain simulation. The procedure is straightforward. We start each timestep with an $\Delta\mathbf{x}_s$ and $\mathbf{x}_c$. The rate of change $d\Delta\mathbf{x}_s/dt = d\mathbf{x}_s/dt$ is obtained from the linear equation using $\Delta\mathbf{x}_c = \mathbf{x}_c - \mathbf{x}_{c0}$ as an input, and $d\mathbf{x}_c/dt$ is obtained from the nonlinear equation using $\mathbf{x}_s = \mathbf{x}_{s0} + \Delta\mathbf{x}_s$ as an input. Then any worthy time-integration technique can be employed to march the equations in time.

There is one catch, though: the aeroelastic part of the system tends to have some very large eigenfrequencies, associated with electrical modes and stiff structural elements. This calls for a singular value decomposition or eigendecomposition; we choose the latter. Computing the eigenvalues $\boldsymbol{\lambda}$ and eigenvector matrix $\boldsymbol{\Phi}$, which contains a normalized eigenvector in each of its columns, a transformation of variables

$$\mathbf{x}_s = \boldsymbol{\Phi}\mathbf{q} \tag{38}$$

is made. Applying this to (35),

$$\frac{d\Delta\mathbf{q}}{dt} = \boldsymbol{\Lambda}\Delta\mathbf{q} + \boldsymbol{\Psi}\tilde{\mathbf{B}}\begin{bmatrix}\Delta\mathbf{u}\\ \Delta\mathbf{x}_c\end{bmatrix}, \tag{39}$$

where $\boldsymbol{\Psi} := \boldsymbol{\Phi}^{-1}$, and $\boldsymbol{\Lambda}$ is a diagonal matrix, with the eigenvalues along its diagonal. The output equation (36) becomes

$$\Delta\mathbf{y} = \mathbf{C}_s\boldsymbol{\Phi}\Delta\mathbf{q} + \tilde{\mathbf{D}}\begin{bmatrix}\Delta\mathbf{u}\\ \Delta\mathbf{x}_c\end{bmatrix}. \tag{40}$$

Many of the eigenvalues $\boldsymbol{\lambda}$ are complex, occuring as complex conjugate pairs. It is most convenient if the state matrices are real. To accomplish this, an additional transform of variables is made. A transform is applied to each pair of complex conjugate modes,

$$\mathbf{q} = \mathbf{Y}\mathbf{z}, \qquad \mathbf{Y} = \frac{1}{2}\begin{bmatrix}1 & i\\ 1 & -i\end{bmatrix} \qquad \mathbf{Y}^{-1} = \begin{bmatrix}1 & 1\\ -i & i\end{bmatrix}, \tag{41}$$

and $\mathbf{Y}$ is the identity matrix for real, first-order modes (Stevens and Lewis 2003). This gives the real state equation

$$\frac{d\Delta\mathbf{z}}{dt} = \mathbf{Y}^{-1}\boldsymbol{\Lambda}\mathbf{Y}\Delta\mathbf{z} + \mathbf{Y}^{-1}\boldsymbol{\Psi}\tilde{\mathbf{B}}\begin{bmatrix}\Delta\mathbf{u}\\ \Delta\mathbf{x}_c\end{bmatrix} = \mathbf{A}^\lambda\Delta\mathbf{z} + \mathbf{B}^\lambda\begin{bmatrix}\Delta\mathbf{u}\\ \Delta\mathbf{x}_c\end{bmatrix} \tag{42}$$

and output equation

$$\Delta\mathbf{y} = \mathbf{C}_s\boldsymbol{\Phi}\mathbf{Y}\Delta\mathbf{z} + \tilde{\mathbf{D}}\begin{bmatrix}\Delta\mathbf{u}\\ \Delta\mathbf{x}_c\end{bmatrix} = \mathbf{C}^\lambda\Delta\mathbf{z} + \tilde{\mathbf{D}}\begin{bmatrix}\Delta\mathbf{u}\\ \Delta\mathbf{x}_c\end{bmatrix}. \tag{43}$$

Now, (42) is no longer diagonalized; but the only coupling is between the complex conjugate pairs. If we consider complex conjugate pairs to be "one mode," which is the most natural way to look at things from an engineer's perspective, then we can still say that the modes are decoupled.

We want to simulate (42) in time, over the frequency band of relevance for the controller (37), and things like the rotor speed and blade pitch that are directly influenced by the controller. These things tend to change slowly, and so we need only the low-frequency or slowly-acting modes in (42). At the same time, we want to retain the correct steady-state response, which is only reproduced exactly if all the modes are retained. The solution is to split (42) into dynamic and static modes,

$$\begin{bmatrix}d\Delta\hat{\mathbf{z}}/dt\\ \mathbf{0}\end{bmatrix} = \begin{bmatrix}\hat{\mathbf{A}}^\lambda & \mathbf{0}\\ \mathbf{0} & \mathbf{A}_q^\lambda\end{bmatrix}\begin{bmatrix}\Delta\hat{\mathbf{z}}\\ \Delta\mathbf{z}_q\end{bmatrix} + \begin{bmatrix}\hat{\mathbf{B}}^\lambda\\ \mathbf{B}_q^\lambda\end{bmatrix}\begin{bmatrix}\Delta\mathbf{u}\\ \Delta\mathbf{x}_c\end{bmatrix}, \tag{44}$$

and solve for

$$\Delta\mathbf{z}_q = (\mathbf{A}_q^\lambda)^{-1}\mathbf{B}_q^\lambda\begin{bmatrix}\Delta\mathbf{u}\\ \Delta\mathbf{x}_c\end{bmatrix} \tag{45}$$

directly. Only $\hat{\mathbf{z}}$, associated with low-frequency modes, is integrated in time. Reassembling $\mathbf{z}$ from $\hat{\mathbf{z}}$ and $\mathbf{z}_q$, the transform

$$\Delta \mathbf{x}_s = \mathbf{\Phi Y}\, \Delta \mathbf{z} \tag{46}$$

gives the aeroelastic and electrical states that are needed as inputs to the controller (37).

Some caution is needed in picking the cut-off $|\lambda|$. This should be set at a frequency where the controller greatly attenuates the input-to-output signal. The reason is that one loses the low-pass behavior of those modes that are represented as steady-state; taking the example of a simple low-pass filter,

$$\frac{dx}{dt} = -\alpha x + \alpha u$$

becomes

$$x = u,$$

which transmits all frequencies equally.

To put the above theory to the test, let us look at a few examples where the controller steps across a control mode transition: first, the typical case where the wind speed increases or decreases across the rated value; and then, a case where a change in the operator power command initiates a transition.

Figure 21 shows the response to a step change in the wind speed from 11 to 12 m/s, which crosses the transition to rated power. Comparing the STAS results with a linear and nonlinear controller, it is evident that the significant nonlinear effects are indeed within the controller, and are associated with the switch in control modes. Since the controller switches modes, we have no reason to expect that the purely linear analysis will be valid; and indeed it is not. Focusing on the nonlinear simulations, these have a peculiar profile, and we will proceed step-by-step with an explanation. At the start of the time period shown in Fig. 21, the turbine is in steady-state operation (or almost so) at a wind speed of 11 m/s. At $t = 30$ s, the wind speed steps to 12 m/s. This increases the aerodynamic torque, and the rotor begins to accelerate. The generator power follows the rotor speed according to the $\hat{P}$-$\overline{\Omega}$ curve (Fig. 10), and the blade pitch follows the $\beta_{\min}$-V^* curve (Fig. 11). As the rotor speed crosses $\Omega_{cd} = 0.95$ rad/s ($t \approx 32$ s), the control mode switches. The target rotor speed is set to $\Omega_r = 1.005$ rad/s, and the power command to 10 MW. Although there is a step change in the error ϵ_Ω as it switches from $(\beta_{\min} - \beta)$ to $(\overline{\Omega} - \hat{\Omega})$, the blade pitch continues to follow $\beta_{\min}$, since the pitch angle is saturated at its minimum value.[16] At $t \approx$ 43-48 s, depending on the model, the rotor speed crosses Ω_r: now the blade pitch comes out of saturation and acts to arrest the rotor speed. After this the response is essentially linear, the rotor speed control mode having a frequency of about 0.1 Hz and damping ratio of around 0.4.

When it comes to the discrepancies between the STAS/nonlinear controller and Ashes results in the upper two plots, it is not certiain what portion of this is due to the linearization and reduction of the aeroelastic model, and what portion is due to other, non-pertinent differences in the STAS and Ashes models.[17] Comparisons between STAS and Ashes often show comparable results with some minor discrepancies – as is the case here. Known differences between the models include

- the Ashes controller was implemented as a discrete-time DLL with linear interpolation of the various schedules (Figs. 10 through 14), while the STAS controller is formulated in Octave as a set of continuous-time state-space equations, with smoothed spline interpolation of all schedules;
- the Ashes model was tuned to match the parked structural frequencies predicted by STAS, but it was not possible to match all the rotor and support structure frequencies precisely;

[16] In other words, since $\overline{\Omega} < \hat{\Omega}$, the blades "want" to pitch out and accelerate the rotor. But since the pitch is saturated, and the integrator anti-windup is in effect, nothing happens.

[17] It is not practical to conduct a full nonlinear STAS simulation with the present code, since the highest eigenvalues are so high that explicit integration methods require a prohibitively small timestep, while implicit methods are also slow, due to the need to update the linearized model at each step.

- STAS includes a detailed model of the electrical powertrain and feedback control based on the measured current at the transformer terminals, whereas Ashes models the electrical system with a constant user-defined efficiency;
- the driveshaft and nacelle structures in STAS are modelled with flexible elements, although these are then reduced to a handful of modes; while Ashes includes drivetrain torsional flexibility but a rigid nacelle;
- STAS employs the $4fa(1-fa)$ form of the axial induction equation, while Ashes applies the $4fa(1-a)$ form.[18]

In any case, we can say that there are differences between the two nonlinear simulations, but they follow a similar sort of profile.

Figure 22 shows the case where the wind speed drops from 12 to 11 m/s, and the controller switches back to the below-rated operating regime. Here the wind speed step occurs at $t = 30$ s, and the rotor begins to decelerate. The blades pitch to arrest the deceleration, and hit the minimum pitch: if one looks closely in the vicinity of $t = 32$ s, a "knee" in the pitch response is visible. At $t = 33$-35 s, the boundary Ω_{cd} is crossed, and the control mode switches. In the case of the STAS simulation, the controller integrator has not yet reached its saturated value, and so there is a "blip" in the blade pitch when the control mode switches.[19] The small-amplitude oscillations visible after the control-mode switch are associated with the tower-frequency notch filter on the measured pitch angle.[20]

We next consider a case of power-stepping, from Fig. 18. Operating at a curtailed power of 6 MW in winds of 10 m/s, a step-change in the power command is given. Stepping down to 5 MW, there is no switch in the control mode, only an adjustment of the blade pitch to restrain the rotor speed. Stepping up to 7 MW, on the other hand, sets the transition speed (Fig. 10) above the operating speed, which initiates a switch in the control mode.

Figure 23 shows the case where the power is stepped down. Here the dominant oscillatory mode is related to the interaction between the wind speed observer, which is setting the target rotor speed Ω_t, and the rotor speed controller acting on the speed error ϵ_Ω. Though this mode is present in both the linear and nonlinear analyses, the damping is lower in the nonlinear case, where the model accounts for things like the gain scheduling and the updated observer aerodynamics at the new operating condition. The change in the reference speed in the STAS models is erroneous; it is associated with an error in the observed wind speed V^*. The nonlinear aspects of the aerodynamics when changing the pitch from 3° to 5° may play a role here.[21]

Finally, Fig. 24 shows the case where the operator power command is stepped up from 6 to 7 MW in a 10 m/s wind. Upon receiving the command, Ω_t moves above the present rotor speed, and the controller immediately switches from the pitch-control to the P-Ω schedule mode, and the blades pitch towards their minimum value. Here the responses of the two nonlinear simulations differ. In the case of STAS, the rotor begins the transition at a somewhat higher speed, and accelerates back across the transition speed, returning to pitch control. Ashes, on the other hand, accelerates less, and oscillates across the transition speed multiple times (Fig. 20). By $t = 40$ s, the turbines in both simulations are operating back in pitch-control mode, with the power set to 7 MW. The maximum available power, however, is just below this: so the rotor gradually decelerates, crossing Ω_t and initiating a

[18] See Burton *et al.* (2001) for a discussion on this topic; f is Prandtl's finite-blade tip loss factor, and a is the axial induction factor $-V_{i,z}/V_\infty$.

[19] The unwanted pitch response can possibly be fixed by linking the anti-windup of the integral pathway to its own output, rather than the total pitch command; this will be tested and implemented in a future iteration of the controller design.

[20] The notch filter removes the pitch response at the tower frequency of about 0.24 Hz, and introduces this moderately-damped resonant mode at 0.2 Hz.

[21] In fact, it might be better to keep the wind speed observer as a linear model, so that the observer best matches the (linearized) plant dynamics.

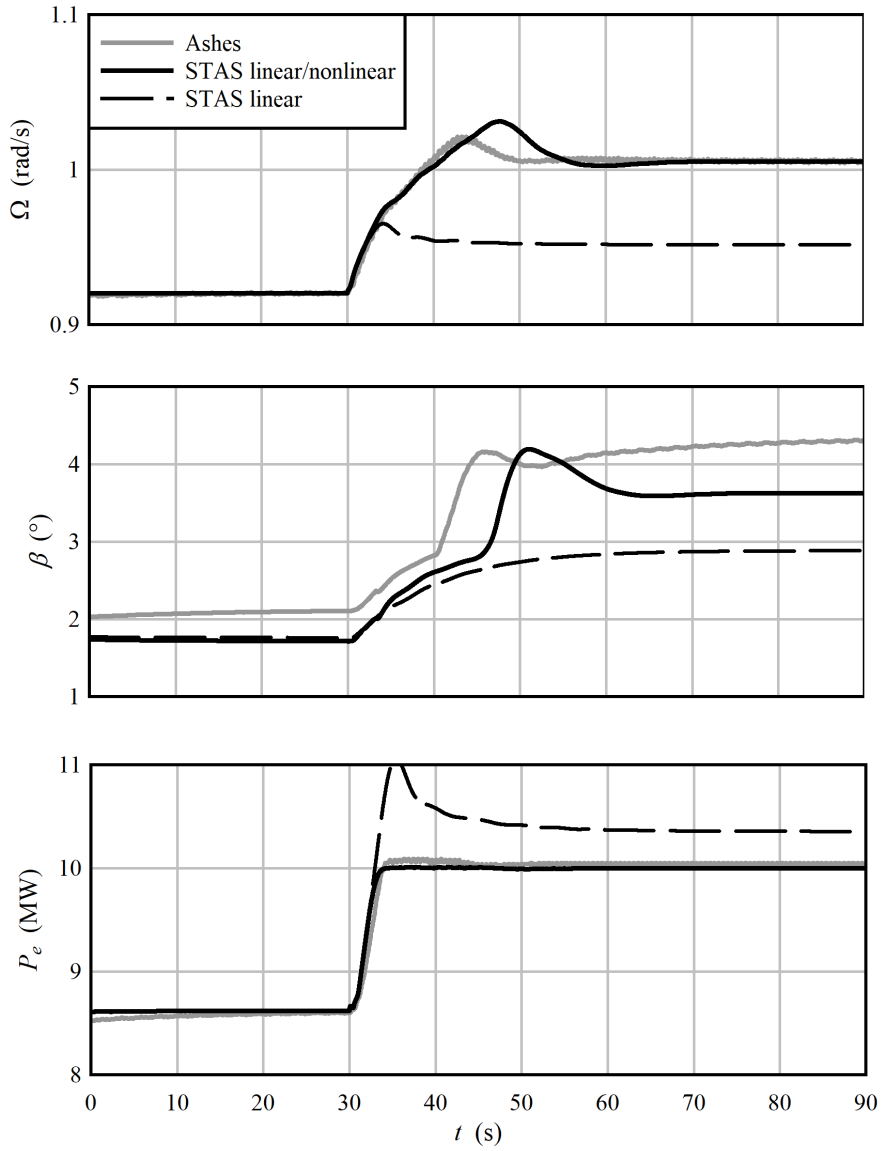


Figure 21: A step in the wind speed from 11 to 12 m/s, comparing an Ashes nonlinear aeroelastic simulation against reduced-order STAS models, one in which the aeroelastic model was linear and the controller was nonlinear, and another with a fully linearized model. In STAS the frequency cutoff was set to 2 Hz; that is, eigenvalues with a magnitude less than 4π were simulated dynamically, while those above were solved quasi-statically.

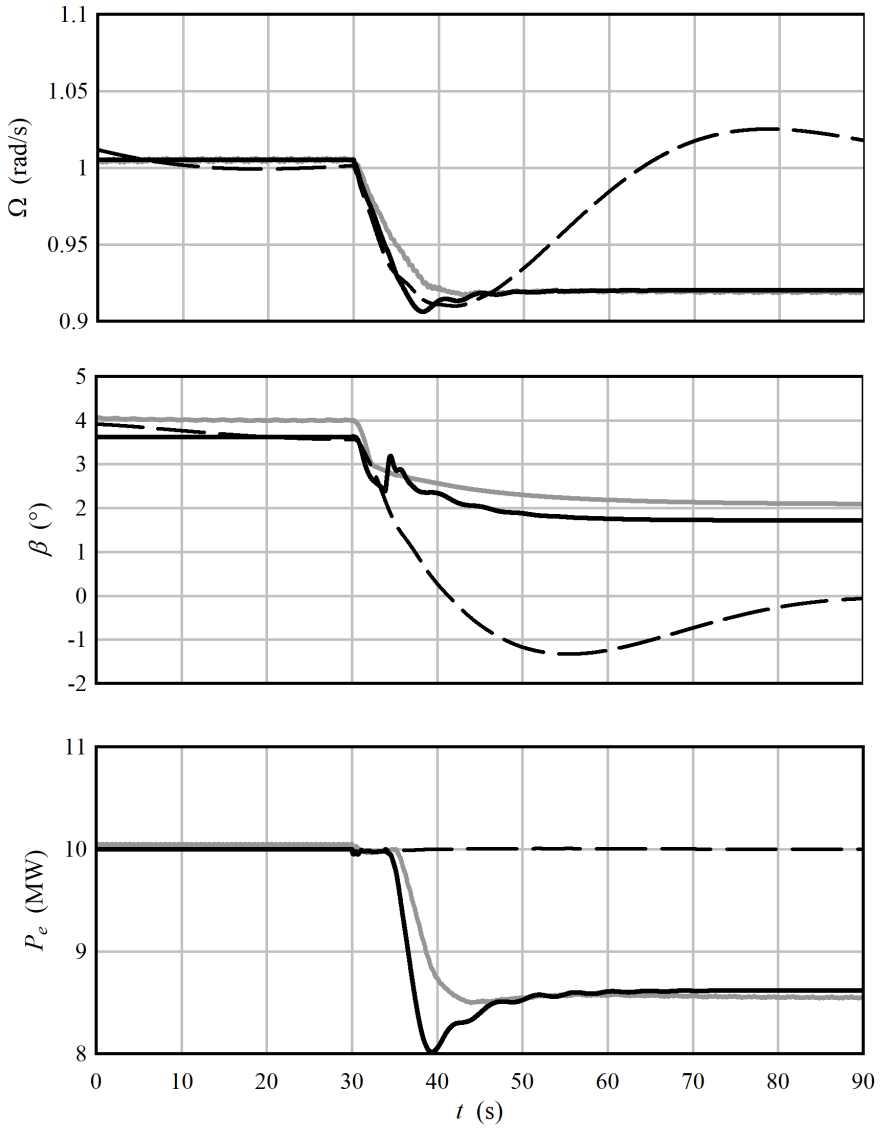


Figure 22: A step in the wind speed from 12 to 11 m/s. The legend is the same as in Fig. 21.

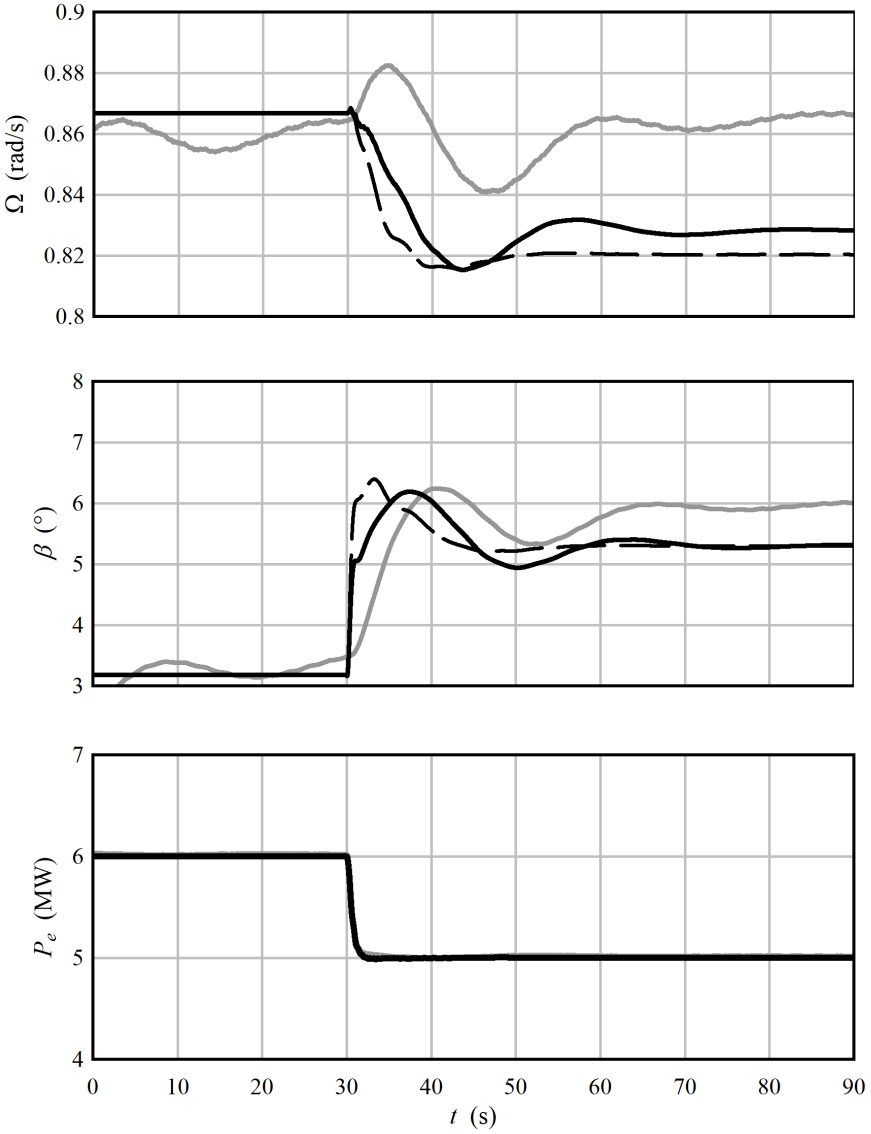


Figure 23: A step in the power command from 6 to 5 MW, comparing the response from different calculations. The Ashes results are a close-up of the step in Fig. 18; it had not quite reached steady state during the previous step, so some small differences in the initial condition are visible. The legend is the same as in Fig. 21.

final control-mode switch, after which the power and speed drop and plateau according to the P-Ω table relationship.

To summarize: we have developed a wind turbine controller with a particular architecture, based around a two-position switch that selects between blade pitch and generator power control of the rotor speed. The intent was an architecture that lends itself to a minimalist representation of nonlinearity, for incorporation into a mixed linear/nonlinear dynamic model. Ultimately, this may be useful for probabilistic analysis of wind power plant dynamics. We tested the controller performance in a fully nonlinear time-domain simulation program, and also conducted a proof-of-concept study for a mixed linear/nonlinear model, where the nonlinearity is limited to the controller. The results indicate that the controller does an excellent job of governing the power generation and rotor speed, but has some unwarranted pitch action in the vicinity of the control-mode switch. Whether this is a concern depends on how much weight is given to the pitch rate when evaluating system performance. Additional iterations of the controller design should be conducted in order to improve its switching characteristics. That said, the results of Figs. 21 through 24 hint that there is promise in this concept of minimizing and isolating the nonlinear elements of a wind power plant, and modelling most of the system by linear equations. The topic is deserving of a deeper and more complete investigation.

4 Newton-Raphson solution of steady-state operating points

4.1 Introduction

The dynamic equations (1)

$$\mathbf{N}(\mathbf{x},\mathbf{p})\frac{d\mathbf{x}}{dt} = \mathbf{f}(\mathbf{x},\mathbf{u},\mathbf{z},\mathbf{p}), \qquad \begin{bmatrix}\mathbf{y}\\ \mathbf{z}\end{bmatrix} = \begin{bmatrix}\mathbf{g}_y(\mathbf{x},\mathbf{u},\mathbf{z},\mathbf{p})\\ \mathbf{g}_z(\mathbf{x},\mathbf{u},\mathbf{z},\mathbf{p})\end{bmatrix}$$

can be linearized about any operating point. Typically, though, we wish to study the dynamics for perturbations about an equilibrium point: those particular values of $\mathbf{x}$ for which $\mathbf{f} = \mathbf{0}$. An equilibrium solution is the starting point for most any application of linear systems theory, be it modal decomposition, frequency-domain analysis, or multivariable control synthesis.[22] The nonlinear equations $\mathbf{f} = \mathbf{0}$ are solved using the Newton-Raphson method.

There is a smart way and a naïve way to solve for the equilibrium value of $\mathbf{x}$. The naïve way (Section 4.2) is to assemble the complete system equations, and then apply the Newton-Raphson procedure at the global system level. The smart way (Section 4.3) is to take advantage of the relatively simple interconnections between modules, and apply a hierarchical solution where the Newton-Raphson method is applied separately for each subsystem, and then globally for the interface variables.

The thing is, the naïve method was found to be functional: given a reasonable starting guess for the states, the equations of Chapter 2 can be solved simultaneously using the Newton-Raphson method. All the results in this report, with the exception of Chapter 8, were obtained using a global solution method, although some tricks (Section 4.4) were employed to get around difficulties with convergence.

4.2 Solution for one module, or the unified system

STAS uses a robust version of the Newton-Raphson method, suggested by Press *et al.* (2007), to solve for $\mathbf{f} = \mathbf{0}$. On the k^{th} iteration, the residual is established as $\mathbf{f}(\mathbf{x}_k, \mathbf{u}, \mathbf{z}_k)$ for one module, considering $\mathbf{z}_k$ as an input; or $\mathbf{f}(\mathbf{x}_k, \mathbf{u}, \mathbf{z}_k(\mathbf{x}_k)) = \mathbf{f}(\mathbf{x}_k, \mathbf{u})$ for the unified system. A linearization is conducted with

[22] Linearization far from equilibrium is necessary algorithmically, for gradient-based optimization, the present Newton-Raphson technique, and the like; but linear dynamic features of a system, such as the modes and frequency response, are not physically meaningful if the nonlinear aspects of the system are changing faster than the relevant modal or response frequencies.

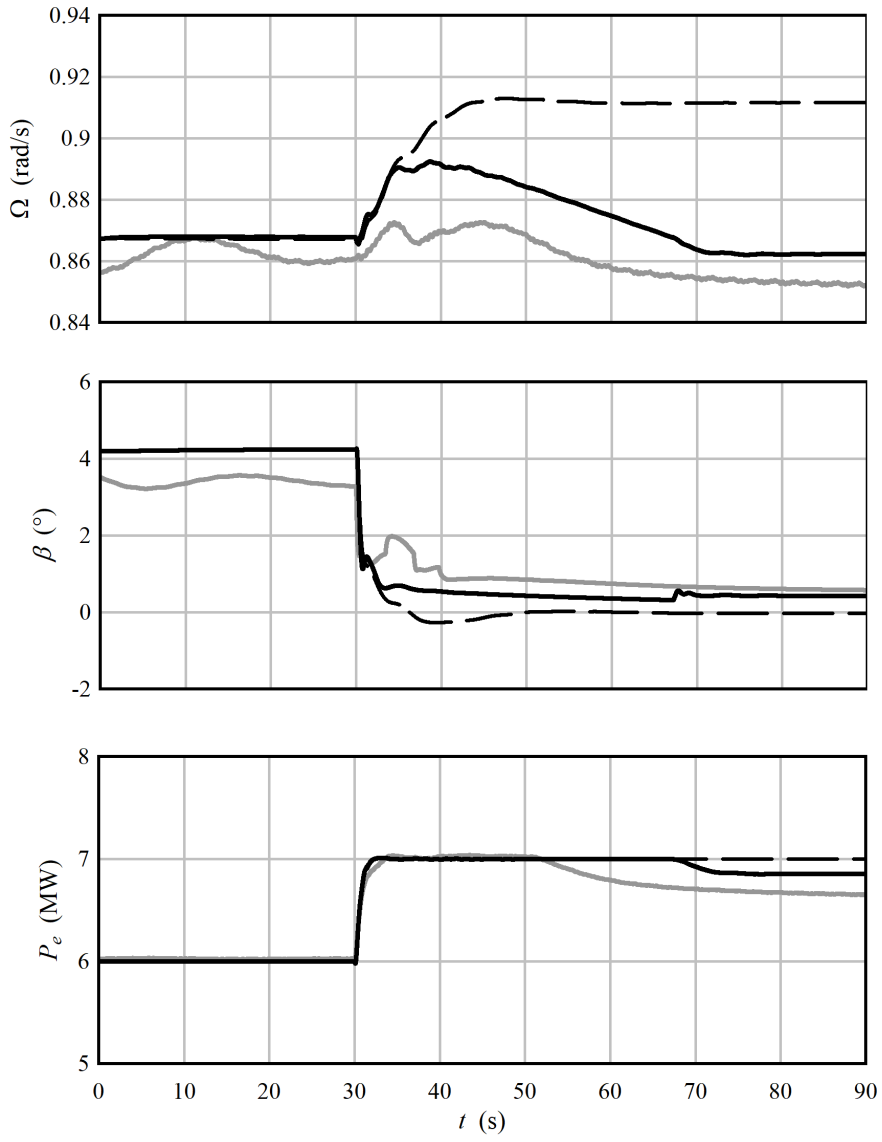


Figure 24: A step in the power command from 6 to 7 MW, where the controller switches from one mode to the other. The legend is the same as in Fig. 21.

respect to $\mathbf{x}$, and an estimate

$$\hat{\mathbf{x}}_{k+1} = \mathbf{x}_k - \left(\left. \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \right|_{\mathbf{x}=\mathbf{x}_k} \right)^{-1} \mathbf{f}(\mathbf{x}_k) \tag{47}$$

is made by linear projection for the value of $\mathbf{x}$ that satisfies $\mathbf{f} = \mathbf{0}$. To make the algorithm more robust, the squared magnitude of the residual

$$R = \mathbf{f}^T \mathbf{f}$$

is required to decrease on each step. Assuming no numerical problems, (47) is guaranteed to be a descent direction for R. The function is evaluated at the projected solution, $\hat{\mathbf{f}}_{k+1}(\hat{\mathbf{x}}_{k+1}, \mathbf{u})$, and the metric $R_{k+1} = \hat{\mathbf{f}}^T \hat{\mathbf{f}}$ is compared against the value R_k from the previous iteration. If $R_{k+1} < R_k$, then the estimate is accepted and the algorithm proceeds to the next iteration of (47). Otherwise, the value of λ is progressively halved, such that the estimated solution retreats towards the starting point. Since (47) is a descent direction, at some step length there will be a point where $R_{k+1} < R_k$, and the solution then proceeds to the next iteration.

Experience shows that this robust Newton-Raphson method works quite well if a reasonable guess is made for the initial values of $\mathbf{x}$. It can, like all gradient-based methods, get stuck at local minima in $\mathbf{f}$. Near a local minimum, the gradient $\partial \mathbf{f}/\partial \mathbf{x}$ does not provide a valid indication of the direction in which $\mathbf{f} = \mathbf{0}$. Section 4.4 suggests some techniques to try when problems with convergence are encountered.

4.3 Hierarchical solution for multiple modules

It is often the case that multiple complicated modules, each of which may have many internal states, are connected by a small number of interface variables. An example is the connection between the mechanical and electrical systems of a wind turbine: the driveshaft passes the rotor speed to the generator, and the generator passes the air gap torque back to the driveshaft.

Consider the case shown in Fig. 7. Say that we are able to obtain the solution $(\mathbf{x}_1, \mathbf{y}_1, \mathbf{z}_{12})$ for $\mathbf{f}_1 = \mathbf{0}$ and $(\mathbf{x}_2, \mathbf{y}_2, \mathbf{z}_{21})$ for $\mathbf{f}_2 = \mathbf{0}$ separately, given values for the interface variables $\mathbf{z} = [\mathbf{z}_{12}\ \mathbf{z}_{21}]$. Denote as $\mathbf{z}_k$ the best estimate for $\mathbf{z}$ at iteration k, and $\tilde{\mathbf{z}}_k$ the corresponding values output from each function. That is, $\tilde{\mathbf{z}}_{12,k}(\mathbf{z}_{21,k})$ and $\tilde{\mathbf{z}}_{21,k}(\mathbf{z}_{12,k})$ are output by respectively modules 1 and 2, and the output values satisfy the criteria that $\mathbf{f}_1 = \mathbf{0}$ and $\mathbf{f}_2 = \mathbf{0}$. We then define a residual,

$$\mathbf{r} = \begin{bmatrix} \tilde{\mathbf{z}}_{12,k}(\mathbf{z}_{21,k}) - \mathbf{z}_{12,k} \\ \tilde{\mathbf{z}}_{21,k}(\mathbf{z}_{12,k}) - \mathbf{z}_{21,k} \end{bmatrix}, \tag{48}$$

and apply (47) to update the estimate for $\mathbf{z}$. This requires computing, for Module 1,

$$\frac{d\tilde{\mathbf{z}}_{12}}{d\mathbf{z}_{21}} = \frac{\partial \tilde{\mathbf{z}}_{12}}{\partial \mathbf{z}_{21}} + \frac{\partial \tilde{\mathbf{z}}_{12}}{\partial \mathbf{x}_1} \frac{\partial \mathbf{x}_1}{\partial \mathbf{z}_{21}}, \tag{49}$$

where $\partial \mathbf{x}_1/\partial \mathbf{z}_{21}$ in the latter term needs to be computed along the profile $\mathbf{f}_1 = \mathbf{0}$; and similarly for Module 2.

4.4 Tricks to aid convergence

Solving large sets of nonlinear equations is difficult; Press *et al.* (2007) give a good explanation of the reasons why. A good starting guess means everything. A wind turbine, in particular, is loaded with summits and valleys – picture a C_P diagram, Fig. 14, or the lift and drag coefficients of an airfoil – where landing on the "wrong side" gives faulty gradient information to the Newton-Raphson method, which all but guarantees that no solution, or at best the wrong solution, will be found.

We found the aerodynamic equations to be the most challenging, since an unlucky Newton-Raphson step can put the angle-of-attack into the stalled zone, where gradients lead away from the true solution. For this reason the initial guesses for the pitch angle and rotor speed were critical. It was best to set the pitch to a couple degrees more than what was expected for the final steady-state value: this places the starting point a bit farther from stall. One idea to help improve the situation might be to simply use $C_L = 2\pi\alpha$, instead of the tabulated value, during an initial sequence of the Newton-Raphson solution.

Starting from a manually-input starting guess, typically with zero structural displacements (since manually inputting guesses for each modal amplitude of each component would be laborious), no solution could be obtained if the initial step size $-(\partial\mathbf{f}/\partial\mathbf{x})^{-1}\mathbf{f}$ were set to its full value. The reason likely has to do with how the residual for some variables is improved at the expense of others. If the residual in the structure was improved, then a Newton-Raphson step might be accepted, even if the aerodynamics or rotor speed were thrown way off. In such cases it was vital to take several small steps at the beginning of the sequence – say, a relative step size of 0.01 – and gradually increase this towards 1.

Locking and freeing different variables is an option. Often one variable dominates the residual. It was occasionally observed that locking all the other variables, and zero'ing the dominant variable's residual alone, could put the solution on the right track. Also, there were cases – including the first successful solution we managed to obtain – where we observed that the rotor speed was diverging towards unrealistic values early in the sequence. Here it helped to make a best guess for the rotor speed based on the controller's operating schedule, and then lock the rotor speed for an initial Newton-Raphson solution. Then, the rotor speed was freed, and the process restarted from the previous solution.

The control regiime for a given operating point is usually easy to determine. A flag was added to the controller to force it to stay in one mode or the other, without switching. This prevented the solution from getting stuck adjacent to the control-mode switching boundary.

Once a first solution was obtained, for some operating condition, then by using this as an initial guess, adjacent operating points could be easily solved. Applying this recursively, a family of solutions could be "grown" across the relevant operating space. Solutions in the vicinity of "elbows" in the power-speed schedule were difficult to obtain, and it helped to progressively step towards such points.

Solutions in the power-command-tracking mode were easier to obtain than those in the variable-power, variable-speed mode, because the former keeps tighter control over the rotor speed. The residuals in the generator power and rotor speed tended to prevent the angle-of-attack from diverging into stall. One strategy that had some success was to begin with a solution in the power-command-tracking mode near to the control-mode switch, and then use this as a starting point for a solution on the other side of the switch. On each solution, the rotor speed controller was locked into the appropriate control regime, to prevent the switch from interfering with the Newton-Raphson solution.

Due to the smoothness of the equations, it was possible to iterate the Newton-Raphson solution until the residual reached a convergence tolerance near machine precision.

5 Complex step gradients

STAS can compute precise gradients with respect to static input parameters $\mathbf{p}$ – or *second* derivatives with respect to dynamic variables $\mathbf{x}$, $\mathbf{u}$, or $\mathbf{z}$ – using the complex step method (Martins *et al.* 2003). Let a complex variable $z = x + i\delta$ have a real part and a small complex part. Taking the Taylor's expansion of a function $f(x)$ about x_0, and evaluating it at $x_0 + i\delta$, that is, taking a complex step,

$$f(z) \approx f(x_0) + \left.\frac{\partial f}{\partial x}\right|_0 (i\delta) + \frac{1}{2}\left.\frac{\partial^2 f}{\partial x^2}\right|_0 (i\delta)^2 + \ldots \tag{50}$$

Comparing the two sides of the equation, noting that $\Re\{f(z)\} = f(x_0)$ to second order in δ,

$$\left.\frac{\partial f}{\partial x}\right|_0 \approx \frac{\Im\{f(z)\}}{\delta}. \tag{51}$$

Complex numbers are native to Octave and are handled by all the functions employed in the STAS program; obtaining a gradient is as simple as defining a single input quantity – say, a parameter p – with a minute complex component, for instance

$$p = \Re\{p\} + i\delta,$$

with $\delta = \sqrt{\epsilon}$, and ϵ the machine precision, `eps` in Octave. Then, the derivative of a function with respect to the parameter is

$$\frac{\partial \mathbf{f}}{\partial p} = \frac{\Im\{\mathbf{f}\}}{\delta}. \tag{52}$$

The advantage of the complex step calculation over finite difference is that it is more precise numerically – it does not involve taking the difference between two closely-spaced evaluations of $\mathbf{f}$.

Packages such as CasADi (web.casadi.org) are available to compute gradients analytically, based on the nonlinear equations. This involves defining the model variables within a specialized computational framework, which complicates and constrains the implementation of the equations. We have elected so far to implement all equations in plain Octave script, though we do not rule out a future CasADi version of STAS. The complex step method is as precise numerically as analytical gradients, but is likely to be slower, especially for large problems: so it is a question of speed versus generality.

6 Scaling atmospheric turbulence spectra for rapid stochastic analysis

6.1 Introduction

The STAS program generates a linear state-space model of a wind turbine, of the form (1). Let this be linked as in (23) to form the unified state space

$$\frac{d\mathbf{x}}{dt} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}. \tag{53}$$

The $\mathbf{x}$ vector is composed of states associated with the structures, aerodynamics, electrical systems, and control systems of the wind turbine, while the $\mathbf{u}$ vector contains the components of turbulence acting at each of a number of aerodynamic elements along the blades and the tower. For input with a given frequency f, the dynamic response is

$$\mathbf{H}(f) := \frac{\partial \mathbf{x}}{\partial \mathbf{u}} = (i2\pi f\mathbf{I} - \mathbf{A})^{-1}\,\mathbf{B}. \tag{54}$$

The transfer function matrix $\mathbf{H}$ is in general complex, indicating the phase between the input turbulence and the states.

A matrix of rotationally-sampled turbulence spectra – that is, the cross-spectra of turbulence seen by points on the rotating blades – is generated by the STAS Wind module, according to the methods of Section 6.2. This spectral matrix $\mathbf{S}_u(f)$ of turbulence is transformed to a spectral matrix $\mathbf{S}_x(f)$ of states by

$$\mathbf{S}_x(f) = \mathbf{H}(f)^{*T}\ \mathbf{S}_u(f)\ \mathbf{H}(f). \tag{55}$$

Based upon the spectral matrix of states, various stochastic quantities can be computed, such as accumulated cycles and extreme values.

Optimization of wind turbine components and systems – including optimal control – generally requires a cost function that incorporates, directly or indirectly, stochastic quantities such as extreme

and fatigue loads, power fluctuations, and the like. Equations (54) and (55) are embedded in an optimization loop that is iterated hundreds or thousands of times.

Generation of the turbulence spectra $\mathbf{S}_u$ is computationally intensive. Since $\mathbf{S}_u$ is a function of the position in space of the aerodynamic elements, as well as the rotor speed, perturbing the blade geometry or the turbine's operating state requires a modification to $\mathbf{S}_u$.

A methodology has been developed to scale the turbulence spectra for perturbations in rotor diameter and rotor speed. For operation at a given windspeed and turbulence intensity, a single set of spectra are generated upfront. The set includes the derivatives of the spectra with respect to windspeed and rotor diameter, where the derivatives are obtained by the complex step method, accurate to machine precision. It is not feasible to directly take the derivative of the spectra with respect to the rotor speed, due to the way in which the rotational frequency scales; but by nondimensionalizing the problem, the perturbations in the spectral *amplitudes* with windspeed can be related to those with rotor speed. The new spectral amplitudes are determined, and then, reintroducing physical units, the spectra are "stretched" to the appropriate frequency axis.

Scaling the turbulence spectra in this manner greatly reduces the computational time of optimization.

6.2 Turbulence model in STAS Wind

The cross-correlation function for isotropic turbulence, following the Von Karman spectrum, is[23]

$$Q_{ss}(s,0) = \frac{2\sigma_u^2}{\Gamma(1/3)}\left(\frac{s}{2.68L_u}\right)\ K_{1/3}\left(\frac{s}{1.34L_u}\right), \tag{56}$$

with derivative

$$\frac{dQ_{ss}(s,0)}{ds} = -\left(\frac{2\sigma_u^2}{\Gamma(1/3)}\right)\left(\frac{1}{1.34L_u}\right)\left(\frac{s}{2.68L_u}\right)^{1/3}\ K_{-2/3}\left(\frac{s}{1.34L_u}\right). \tag{57}$$

Here s is the separation between two points, σ_u is the standard deviation of turbulent velocity fluctuations, L_u is the Von Karman length scale, Γ is the Gamma function, K is the modified Bessel function of the second kind. The spatial correlations between turbulence components at any two points are (Davidson 2004)

$$Q_{ii}(s,0) = Q_{ss}(s,0) + \frac{s}{2}\frac{dQ_{ss}(s,0)}{ds} - \frac{s_i^2}{2s}\frac{dQ_{ss}(s,0)}{ds} \tag{58}$$

for velocity components in the same direction, where i is either the x, y, or z component, and

$$Q_{ij}(s,0) = -\frac{s_i s_j}{2s}\frac{dQ_{ss}(s,0)}{ds} \tag{59}$$

for orthogonal components. The spectra are obtained from the Fourier transform of the correlations.

Figure 25 relates the turbulence components at points r_1 and r_2 on the blades, separated by a time τ, to the components separated by spatial vector s at time 0. This gives

$$s_x = r_2\cos\Omega\tau - r_1, \quad s_y = r_2\sin\Omega\tau, \quad s_z = V_\infty\tau, \tag{60}$$

and

$$s := |\mathbf{s}| = \sqrt{s_x^2 + s_y^2 + s_z^2}. \tag{61}$$

From (56) to (61), the parameters that describe the problem are the windspeed V_∞; the turbulence intensity I, such that $\sigma_u = IV_\infty$; the turbulence length scale L_u; the rotor radius R, assuming that the radial coordinates r_1 and r_2 scale with R; and the rotor speed Ω. In physical (mks) units, during a dynamic analysis at particular environmental conditions, V_∞, I, and L_u are fixed, while R and Ω are design parameters.

[23] Connell (1982), Kristensen and Frandsen (1982). Merz (2015) provides the necessary derivatives, and describes the implementation in the STAS program. The Von Karman spectrum is presented and discussed by Burton et al. (2001).

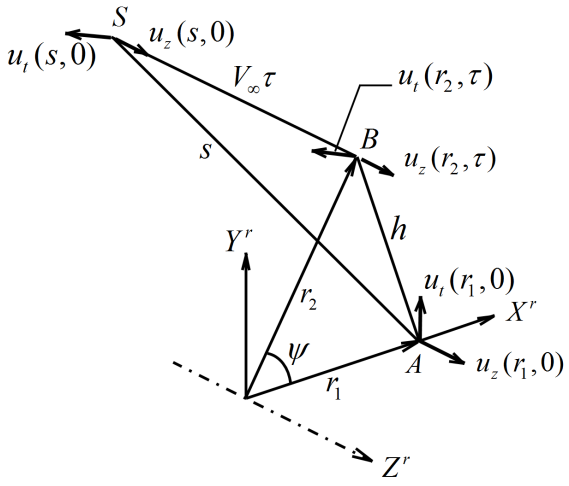


Figure 25: A sketch of the geometry for the rotationally-sampled turbulence problem.

6.3 Scaling the turbulence spectra

For perturbations in R and Ω, it would seem reasonable to propose

$$\mathbf{S}_u(R+\Delta R,\Omega+\Delta\Omega)\approx\mathbf{S}_u(R,\Omega)+\frac{\partial\mathbf{S}_u}{\partial R}\Delta R+\frac{\partial\mathbf{S}_u}{\partial\Omega}\Delta\Omega. \tag{62}$$

This works for perturbations in R of a reasonable size, but not for Ω. The reason is that the frequencies at which peaks in the spectrum occur – these peaks existing due to rotational sampling of turbulent eddies by the spinning blades – scale with the rotor speed. As Ω varies, extrapolating from an instantaneous snapshot of the gradients does not properly represent the translation of the spectral peaks.

A consideration of the blade rotating through the field of turbulence shows that *the same points in the turbulence field are sampled if either the rotor speed is increased, or the windspeed is decreased by an equivalent fraction.* This suggests that we can rather apply

$$\frac{\partial\mathbf{S}_u}{\partial\Omega}=\frac{\partial\mathbf{S}_u}{\partial u}\frac{\partial u}{\partial\Omega}=\frac{\partial\mathbf{S}_u}{\partial u}\frac{V_\infty}{\Omega} \tag{63}$$

to obtain the expected sampling effect with rotor speed.

In STAS Wind, the derivatives $\partial\mathbf{S}_u/\partial R$ and $\partial\mathbf{S}_u/\partial u$ are computed by the complex step method. In the former case, the rotor radius input to the model is perturbed according to

$$\tilde{R}=R+i\delta, \tag{64}$$

such that the radial coordinates become

$$\tilde{r}_k=\frac{r_k}{R}\tilde{R}. \tag{65}$$

Upon computing $\mathbf{S}_u$, the derivative sought is

$$\frac{\partial\mathbf{S}_u}{\partial R}=\frac{\Im_\delta\{\mathbf{S}_u\}}{\delta}, \tag{66}$$

which is accurate to $O(\delta^2)$. The notation $\Im_\delta$ is used, since $\mathbf{S}_u$ is itself a matrix of complex spectra, and the two uses of complex values must not be blended. Cerviño and Bewley (2003) provide guidance on this matter; in short, one should work separately with the real and imaginary parts of the complex spectra – each of which then becomes complex due to the perturbation (64).

The derivative $\partial \mathbf{S}_u / \partial u$ is computed by perturbing the windspeed input to the calculation as

$$\tilde{V}_\infty = V_\infty + i\delta, \tag{67}$$

while adjusting the turbulence intensity by

$$\tilde{I} = \frac{I V_\infty}{\tilde{V}_\infty} \tag{68}$$

such that the strength of the turbulent eddies remains fixed. The modification (68) to the turbulence intensity is needed, since we are not actually perturbing the wind-speed, but rather using this as an indirect step towards a perturbation of the rotor speed. Finally, on obtaining

$$\frac{\partial \mathbf{S}_u}{\partial u} = \frac{\Im_\delta\{\mathbf{S}_u\}}{\delta}, \tag{69}$$

the spectra (62) are calculated, and the frequency axis is stretched by $(\Omega + \Delta\Omega)/\Omega$ so as to correctly place the multiples of the rotor frequency. Stretching the frequency axis requires that the spectral magnitudes are reduced by the same fraction,

$$\mathbf{S}_u \frac{\Omega}{\Omega + \Delta\Omega}, \tag{70}$$

preserving the area under the spectral curves (variance).

6.4 Illustrative results

Figure 26 shows the collective axial turbulence spectra seen by the outboard portion of a rotating rotor, here the DTU 10 MW turbine (Bak 2013). The original radius and rotor speed were each increased by 10%; quite a lot, in the context of a single design optimization. New spectra were generated, exactly by the equations of Section 6.2, and approximately by scaling the original spectra according to the procedures of Section 6.3.

The portion of the scaled spectra associated with stochastic turbulence matches closely. The scaled spectra do not quite capture the spikes associated with the increased severity in wind shear with radius. This is a minor effect; and it is the worst case, at the blade tip, that is shown in the plot.

7 Electrical design of the TotalControl Reference Wind Power Plant

7.1 Introduction

The TotalControl Reference Wind Power Plant (TC-RWP; Andersen 2018) is an archetypical layout of wind turbines, intended for use in developing dynamic models and control algorithms for offshore wind power plants. The layout, including electrical cabling, is shown in Fig. 27. The size of the plant is a compromise between being large enough such that the dynamics, in particular the atmospheric flow, is representative of real offshore sites; yet small enough that it is feasible to simulate the atmospheric flow through the plant using high-resolution CFD (RANS and LES) models.

An offshore wind power plant based on the Borssele site in the Dutch North Sea (B-RWP) is also provided for use in the TotalControl project. The layout and electrical cabling is shown in Fig. 28. The B-RWP is essentially interchangeable with the IEA Wind Task 37 Offshore Reference Wind Power Plant (IEA-RWP)[24]: the same location, water depth, and wind climate; the same number and type of wind turbines; the same cable types, offshore substation, and electrical export system. The difference between the B-RWP and IEA-RWP is the particular layout of the turbines. The B-RWP has

[24] A description of the IEA-RWP is to be released soon as an IEA report.

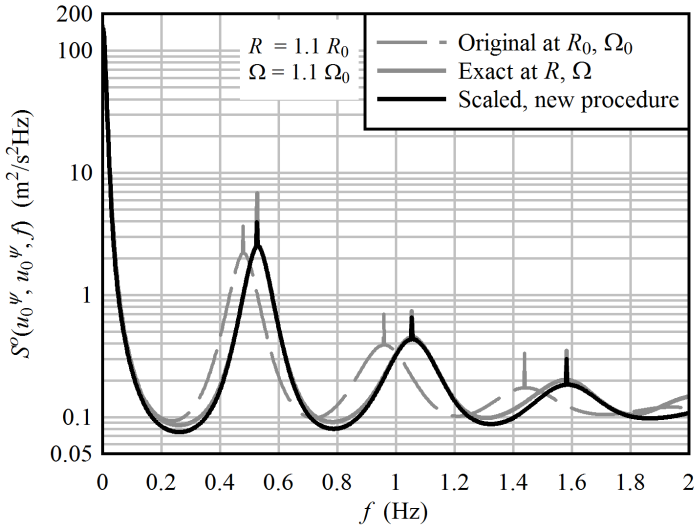


Figure 26: Exact and scaled rotationally-sampled turbulence spectra, for a 10% increase in both rotor radius and speed. The mean windspeed is 13 m/s, the turbulence intensity is 16.5%, and the length scale is 180 m.

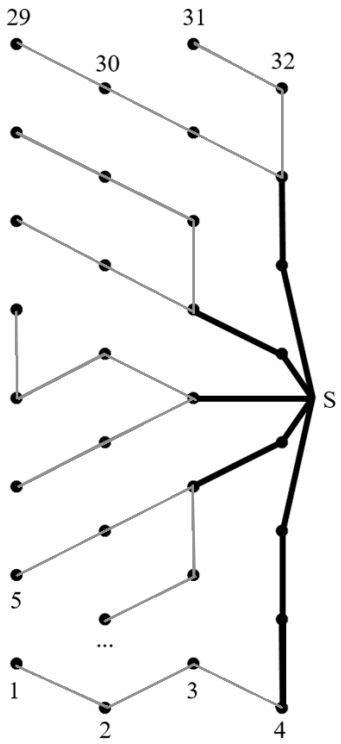


Figure 27: Layout and cable connections of the TotalControl Reference Wind Power Plant. Dimensions are given in terms of s/D, the ratio of the spacing to the turbine rotor diameter.

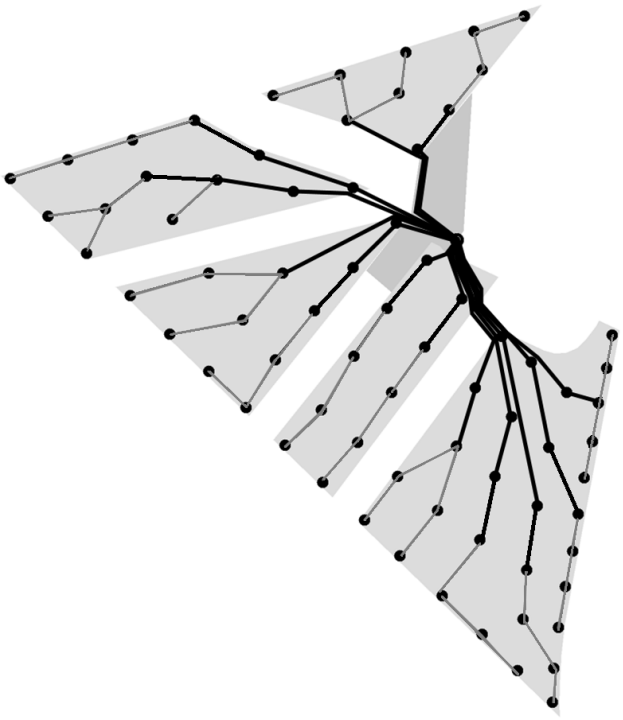


Figure 28: Layout and cable connections of the Borssele Reference Wind Power Plant. This plant is the same as the IEA Wind Task 37 Reference Wind Power Plant, but with an irregular layout.

been designed with an irregular, perimeter-weighted layout, which obeys the actual design constraints for the Borssele site; while the IEA-RWP has a regular staggered layout, akin to the TC-RWP in Fig. 27. An irregular layout is emphasized here, in order to draw attention to the particular research challenges associated with the atmospheric flow and control of such layouts; and to provide a case that is as realistic as possible for demonstration of TotalControl wind plant control algorithms.

7.2 Electrical design of the TC-RWP

The electrical layout of the TC-RWP, Fig. 27, has been conducted according to the EERA DTOC inter-array design procedure (Endegnanew et al. 2013). It consists of two strings of 7 turbines, and three strings of 6 turbines. Branching reduces the overall length of the thickest (500 mm^2) cable.

The inter-array grid voltage is 66 kV, which is foreseen as the standard for the next generation of offshore wind power plants, with turbines approaching a 10 MW rating. The electrical substation contains two 66/220 kV transformers rated at 180 MVA each. The AC system frequency is 50 Hz.

This design is based only on static calculations; no load flow or dynamic studies were performed. Therefore, the plant's electrical design is realistic but not optimized. The wind turbine transformer rating is 11 MVA, and the current per phase is 96 A at rated power. The following cable cross-sections were chosen based on the (static) current rating shown in Table I: 95 mm^2 and 630 mm^2. The cables can transport power of up to three and seven turbines, respectively.

The following assumptions were made while choosing submarine cables:

- Laying depth: 1.5 m
- Soil resistivity: 10 Ωm

It was assumed that the collection grid cables are of a dry design[25] due to the availability of design

[25]"Dry" submarine cables have an impervious moisture barrier. "Wet" designs, on the other hand, do not have such

Table I: Current rating for three-core submarine cables with copper or aluminum conductors and steel wire armor (ABB 2010).

10-90 kV XPLE 3-core cables			100-300 kV XPLE 3-core cables		
Cross-section	Copper	Aluminum	Cross-section	Copper	Aluminum
mm^2	A	A	mm^2	A	A
95	300	235	300	530	430
120	340	265	400	590	485
150	375	300	500	655	540
185	420	335	630	715	600
240	480	385	800	775	660
300	530	430	1000	825	720
400	590	485			
500	655	540			
630	715	600			
800	775	660			
1000	825	720			

information. However, the trend in the industry is towards using a wet design, as this is, in general, simpler. At the time this work was conducted, it was not possible to obtain publicly-available design information for a wet design cable.

The following assumptions were made while choosing land cables:

- Flat formation
- Distance to landfall: 30 km
- Length from landfall to connection with the onshore grid: 30 km
- Laying depth: 1 m
- Distance between phases: 0.1 m
- Distance between groups: 0.8 m
- Ground temperature: 20° C
- Ground thermal resistivity: 1 K·m/W
- Soil resistivity: 100 Ωm
- Cross bonding of screens, with 10 major sections, and each minor section of 1 km length.

Table II lists properties of the submarine collection grid cables, as well as the export cables over the submarine and land sections. DC resistances are based on IEC 60228 and AC resistances are calculated for max operating temperature of 90° C according to IEC 60287.

Figure 29 shows a simplified line diagram of the TC-RWP export system. A shunt reactor is placed by the onshore transformer substation, and it compensates 100% of both cables' reactive power. From Table II, the total capacitance of the export cables is $30 \cdot 0.194 + 30 \cdot 0.220 = 12.46$ μF.

A 100% cable compensation is assumed, therefore, $Q_L = Q_c = 63.2$ MVA per phase, and $L = 1/(\omega^2 C)$. Assuming that the shunt reactor's X/L ratio is 500, the reactor's parameters (per phase) are

$$L = 0.8132\,\text{H}, \quad X_L = 255.5\,\Omega, \quad R = 0.510\ \Omega.$$

Data for offshore transformers:

a robust moisture barrier.

Table II: Properties of TC-RWP cables. A_c: conductor area, D_c: conductor diameter, t_i: insulation thickness, D_i: diameter over insulation, t_s: lead sheath thickness, D_o: outer diameter, R_{DC}: DC resistance, R_{AC}: AC resistance at 50 Hz, L: inductance, C: capacitance.

Cable	Type	A_c mm^2	D_c mm	t_i mm	D_i mm	t_s mm	D_o mm	R_{DC} Ω/km	R_{AC} Ω/km	L mH/km	C μF/km
Collection grid-66kV	1x3x95 Cu	95	11.2	9.0	31.6	1.3	113.0	0.1933	0.2466	0.440	0.170
Collection grid-66kV	1x3x630 Cu	630	29.8	21.4	86.2	2.0	157.0	0.0302	0.0361	0.330	0.320
Export sub. -220kV	1x3x1200 Cu	1200	41.2	21.4	86.2	2.0	212.0	0.0151	0.0211	0.361	0.194
Export land-220kV	3x1x1400 Cu	1400	46.4	23.0	97.4	-	115.6	0.0129	0.0164	0.510	0.220

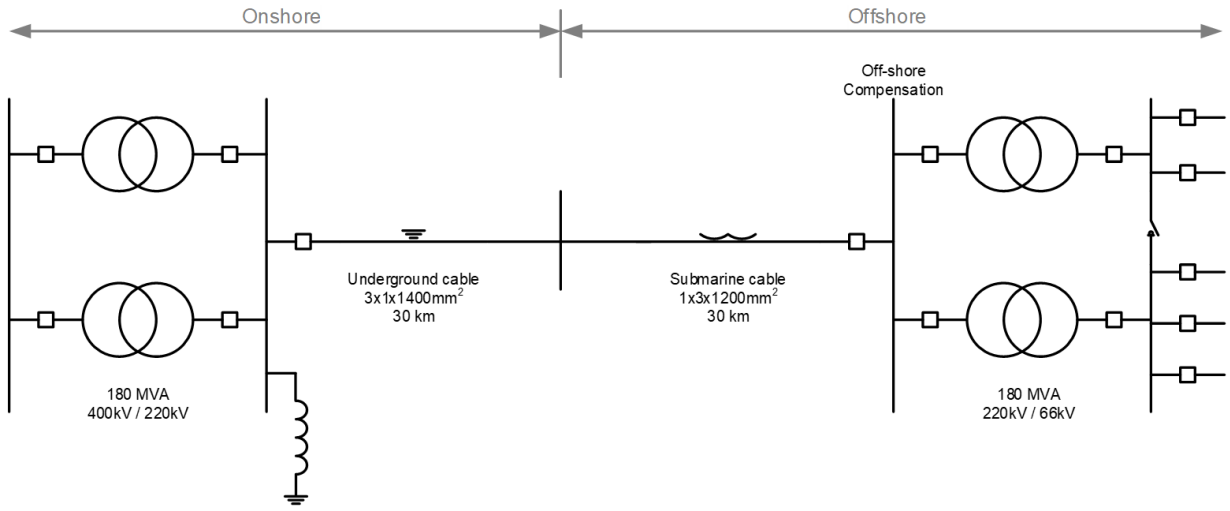


Figure 29: A simplified electrical diagram of the TC-RWP electrical export system.

Table III: Properties of the B-RWP/IEA-RWP submarine export cable. The properties of the collection grid cables, and the definition of terminology, can be found in Table II.

Cable	Type	A_c mm^2	D_c mm	t_i mm	D_i mm	t_s mm	D_o mm	R_{DC} Ω/km	R_{AC} Ω/km	L mH/km	C μF/km
Export sub. -220kV	2x1x3x1400 Cu	1400	44.0	21.0	92	3.4	234	0.0129	0.0173	0.344	0.220

- Power: 180 MVA
- Ratio: 220/66 kV
- Vector group: YNd
- Short-circuit impedance: 12%
- No-load current: 0.2%

Export transformer, on land:

- Power: 180 MVA
- Ratio: 400/220 kV
- Vector group: YNyn
- Short-circuit impedance: 12%
- No-load current: 0.2%

7.3 Electrical design of the B-RWP/IEA-RWP

The inter-array grid voltage is 66 kV, which is required by NEA (2016). The electrical substation contains two 66/220 kV transformers rated at 400 MVA each; the extra capacity is used to handle reactive power. The AC system frequency is 50 Hz.

There are two 220 kV export cables for transmission from the substation to shore. Each cable is rated at 400 MVA. The export cable length between the substation and onshore grid connection is approximately 68 km, with 67 km offshore and approximately 1 km onshore.

The wind turbine transformer rating is 11 MVA. The following cable cross-sections were chosen based on the (static) current rating shown in Table I: 95 mm^2 and 630 mm^2. The cables can transport power of up to three and seven turbines, respectively.

When choosing submarine and land cables, the same assumptions were made as for the TC-RWP (Section 7.2). The submarine cable connects directly the onshore and offshore substations: there is no land cable. Table III lists properties of the submarine export cable; the collection grid cables are identical to those of Table II.

DC resistances are based on IEC 60228 and AC resistances are estimated for max operating temperature of 90° C according to IEC 60287. Dimensions, inductance and capacitance of the 220 kV submarine cable are estimated based on the authors' experience. A diagram of the export system is shown in Fig. 30.

Shunt reactors are placed at 220 kV side of the onshore transformer, and each of them compensate 100% of both cables' reactive power. Total capacitance (per phase) of each of the export cables is $68 \cdot 0.22 = 14.96\,\mu\mathrm{F}$. At 100% cable compensation, the reactor's parameters (per phase) are

$$L = 0.6773\,\mathrm{H}, \quad X_L = 212.8\,\Omega, \quad R = 0.426\ \Omega.$$

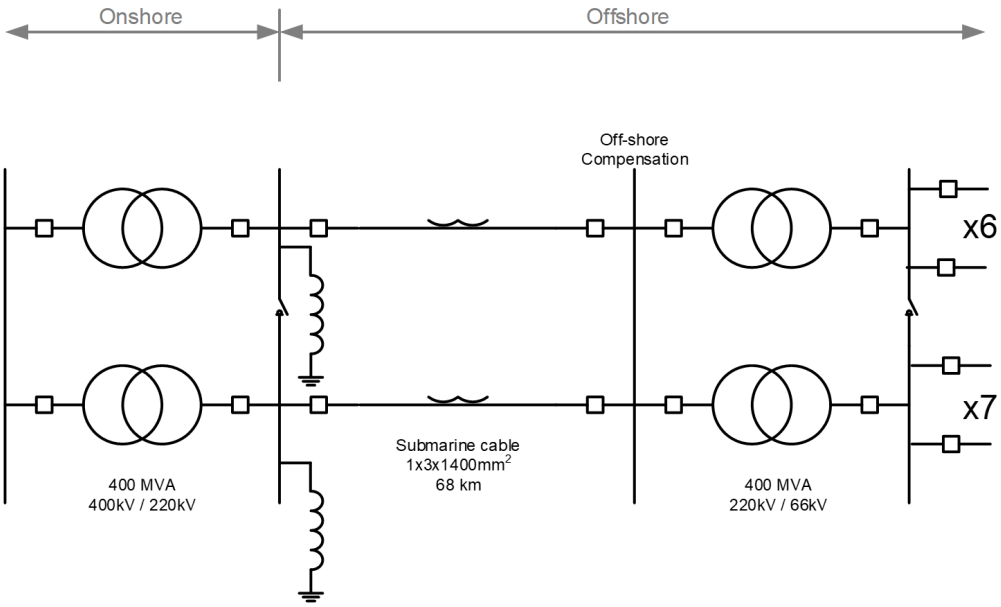


Figure 30: A simplified electrical diagram of the B-RWP/IEA-RWP electrical export system.

Resistance is modelled in series with each reactor's inductance. Placing shunt reactors onshore is a much cheaper solution than splitting the compensation and placing the reactors at both ends of the cable. However, due to the amount of reactive current in the cable, it might be beneficial to place an additional shunt reactor offshore. This was not investigated as part of the present design.

Data for offshore transformers:

- Power: 400 MVA
- Ratio: 220/66 kV
- Vector group: YNd
- Short-circuit impedance: 12%
- No-load current: 0.2%

Export transformer, on land:

- Power: 400 MVA
- Ratio: 400/220 kV
- Vector group: YNyn
- Short-circuit impedance: 12%
- No-load current: 0.2%

8 Closed-loop dynamics of the TotalControl Reference Wind Power Plant

8.1 Introduction

We close the report with a linear dynamic analysis of the TotalControl Reference Wind Power Plant (TC-RWP). The pieces are now in place to build a unified state-space model of the plant, like that

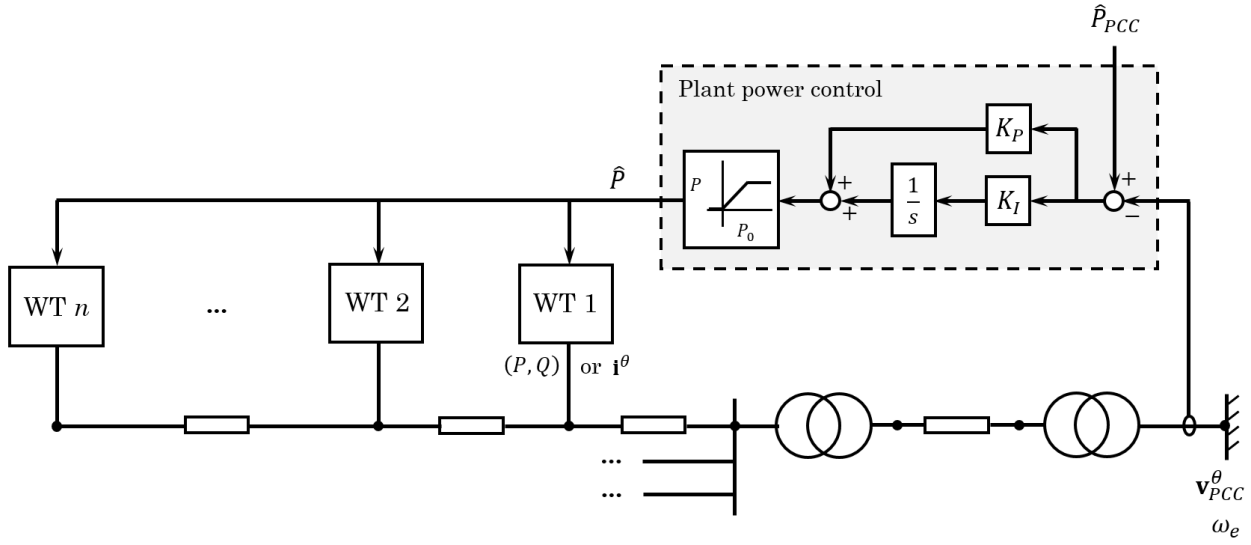


Figure 31: The TC-RWP system architecture with the simplest possible active power controller.

shown in Fig. 8. Figure 31 provides another view of the same system, here emphasizing the feedback loop from the grid through the plant controller. The wind turbines – each consisting of aeroelastic, actuator, electrical, and turbine control modules – are linked to the electric grid; the interface variables can be either d and q axis currents, or active and reactive power. The electric grid is linked back to the turbines through the plant controller. For this preliminary analysis the control algorithm is as simple as possible: PI control of the active power.

The plant geometry is also slightly idealized, to five identical strings of seven turbines, instead of two strings of seven and three of six turbines. The STAS code can handle arbitrary plant layouts; but nothing is gained in the present context by making the electrical layout more complicated, and a model consisting of five identical strings was more reliably programmed and verified.

8.2 Formulating the problem

As a practical shortcut for the present analysis, we forego a system-wide Newton-Raphson solution. Rather, in the spirit of Section 4.3, the mean operating power of each wind turbine is specified upfront. Then the steady-state wind turbine and electrical load flow analyses can be decoupled, and each solved independently by the Newton-Raphson method.

After obtaining operating points, about which the wind turbine and electrical grid equations are linearized, the equations are assembled into a unified set of state matrices,

$$\mathbf{N}\frac{d}{dt}\begin{bmatrix}\Delta\mathbf{x}_1\\ \vdots\\ \Delta\mathbf{x}_n\\ \Delta\mathbf{x}_E\\ \Delta\mathbf{x}_c\end{bmatrix} = \begin{bmatrix}\mathbf{A}_{11} & \dots & \mathbf{0} & \mathbf{A}_{1E} & \mathbf{A}_{1c}\\ \vdots & \ddots & \vdots & \vdots & \vdots\\ \mathbf{0} & \dots & \mathbf{A}_{nn} & \mathbf{A}_{nE} & \mathbf{A}_{nc}\\ \mathbf{A}_{E1} & \dots & \mathbf{A}_{En} & \mathbf{A}_{EE} & \mathbf{A}_{Ec}\\ \mathbf{A}_{c1} & \dots & \mathbf{A}_{cn} & \mathbf{A}_{cE} & \mathbf{A}_{cc}\end{bmatrix}\begin{bmatrix}\Delta\mathbf{x}_1\\ \vdots\\ \Delta\mathbf{x}_n\\ \Delta\mathbf{x}_E\\ \Delta\mathbf{x}_c\end{bmatrix} + \mathbf{B}\Delta\mathbf{u}, \tag{71}$$

which give the dynamic response for fluctuations about the steady state. With 35 wind turbines, the matrices in (71) are large – a hundred or more states per turbine, even for a compact model, and another couple hundred for the grid – and the methods need to scale to larger plants of 100 or more wind turbines. To make the matrices easier to handle, an eigenmode decomposition is performed for

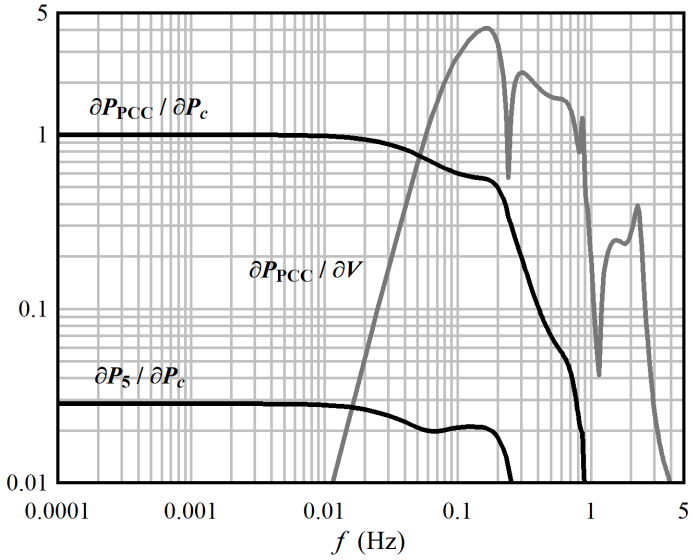


Figure 32: The responses of the wind power plant to an active power command and a fluctuating wind speed. The response of Turbine #5 is also shown.

each turbine,[26] and separately for the electrical system. This gives a transformation of variables

$$\mathbf{x} = \mathbf{\Gamma q}, \qquad \mathbf{\Gamma} = \begin{bmatrix} \mathbf{\Phi}_1 & & & & \\ & \ddots & & & \\ & & \mathbf{\Phi}_n & & \\ & & & \mathbf{\Phi}_E & \\ & & & & \mathbf{I} \end{bmatrix}. \tag{72}$$

Following the procedures of Section 3.6, the individual modes can be divided into dynamic and quasi-steady, based on their rapidity of response. This gives the reduced dynamic system, after inverting $\mathbf{N}$, which is straightforward, since this is block diagonal,

$$\frac{d\Delta\mathbf{q}}{dt} = \mathbf{\Gamma}^{-1}(\mathbf{N}^{-1}\mathbf{A})\mathbf{\Gamma}\Delta\mathbf{q} + \mathbf{\Gamma}^{-1}(\mathbf{N}^{-1}\mathbf{B}) = \mathbf{A}^{\Gamma}\Delta\mathbf{q} + \mathbf{B}^{\Gamma}\Delta\mathbf{u}. \tag{73}$$

Transfer functions can be computed by

$$\frac{\partial \mathbf{x}}{\partial \mathbf{u}} = \Gamma\left[(i\omega\mathbf{I} - \mathbf{A}^{\Gamma})^{-1}\mathbf{B}^{\Gamma}\right]. \tag{74}$$

8.3 An illustrative operating case

Let the wind speed throughout the plant be a uniform 10 m/s, with the turbines curtailed to 6 MW each. Under these conditions, what is the response of the plant to operator power commands, and to fluctuations in the plant wind speed? Figure 32 plots transfer functions of the output power at the PCC between the wind plant and the onshore grid, as a function of operator power commands, as well as a fluctuation in the plant wind speed. Here the controller gains were tuned to $K_P = 0.3/N_t$ and $K_I = 0.2/N_t$ s^{-1}, where $N_t = 35$ is the number of turbines in the plant.

It is seen from the unit magnitude of the $\partial P_{\mathrm{PCC}}/\partial P_c$ transfer function at low frequencies, and the low-pass response at higher frequencies, that the feedback control is functioning as intended.

[26] If their operating states are similar then this need be done only once, and the result can be applied to all the turbines. If some turbines are curtailed and others are not, or if there is a gradient in the wind across the plant, then the modes should be computed separately for each turbine.

The $\partial P_5/\partial P_c$ transfer function for Turbine #5 has roughly an amplitude of $1/N_t$ of the total plant command, which agrees with the simple uniformly-distributed power set-point control strategy in this example. The response of the plant to wind speed disturbances, $\partial P_{\mathrm{PCC}}/\partial V$, is driven to zero at low frequencies, and becomes larger at 0.1 Hz, where the turbine control and structural frequencies lie. The majority of the energy in the turbulence is at frequencies below 0.01 Hz, so we can expect that the plant power fluctuations in this example would be quite small.

This analysis demonstrates that the STAS program can indeed compute the linearized dynamics of a full-scale wind power plant. This is where we must close the report: an evaluation of the plant-scale dynamics, and how these are influenced by various flavors of control, await the next phase of research in the TotalControl project. It is recommended to validate the computations against measurements collected at an operating wind power plant.

9 Conclusions

SINTEF's STAS program was used to construct a model of the TotalControl Reference Wind Power Plant. STAS generates detailed aeroelastic, electrical and control models, which are integrated into a single, holistic state space. STAS provides both the nonlinear equations and a high-precision linearization about any operating point.

We have designed a wind turbine control algorithm that is suitable for the STAS framework, and provides tracking of operator power commands over the entire operating regime. The design of the controller is a bit quirky, as it is built around a single hard switch between two control modes. We expect that this design will aid probabilistic analysis of wind power plant dynamics: our hope is to run such analyses in real time, as part of a control algorithm.

It was shown that it is possible to solve for the steady-state operating point of a wind turbine, including aeroelastic, electrical, and control dynamics, using the Newton-Raphson method. Some guidance was given on how to improve the chances of convergence. Modifications to the basic Newton-Raphson method were necessary, in order to prevent the analysis from straying into conditions where the airfoils are stalled, and gradients lead away from the true solution.

STAS provides precise numerical gradiients with respect to any input parameter, using the complex step method, along with Octave/Matlab's built-in support for complex operations. A brief overview of the technique was given.

A high-performance wind plant control algorithm will consider both the production of the plant, and the loads on the wind turbines, in particular the influence of turbine control actions on wake velocity deficits and turbulence. A rapid way of evaluating the response of a wind turbine under given levels of turbulence is needed for real-time control, and this would also be very useful in design optimization and optimal control studies. The updating of turbulence spectra to changes in the operating state – wind speed or rotor speed – was greatly accelerated by developing a special scaling procedure that appropriately accounts for the timing of the blades relative to the turbulence field.

The electrical configuration of the TotalControl Reference Wind Power Plant was designed in detail, including the export system. The electrical design is sufficient to construct a model of the electrical dynamics and resonant modes, identify losses, and evaluate frequency and voltage stability.

Finally, the response of the TotalControl Reference Wind Power Plant to an operator power command was demonstrated, based on a simple PI feedback controller uniformly adjusting the turbine power set-points. Here the purpose was limited to showing that the closed-loop plant-scale dynamics can indeed be computed, including 35 full-fidelity wind turbines. The next phase of research in the TotalControl project will dive into the plant-scale dynamics, and how to shape these optimally through the design of the plant supervisory controller.

References

ABB (2010). *XLPE Submarine Cable Systems: Attachment to XLPE Land Cable Systems – User's Guide*. Rev. 5, ABB.

Anaya-Lara O, *et al.* (2009). *Wind Energy Generation: Modelling and Control*. Wiley.

Anaya-Lara O, *et al.* (2018). *Offshore Wind Energy Technology*. Wiley.

Andersen S, *et al.* (2018). *Reference Wind Power Plant*. TotalControl Deliverable D1.03.

Bak C, *et al.* (2013). *Description of the DTU 10 MW Reference Wind Turbine*. DTU Wind Energy Report-I-0092, Technical University of Denmark.

Bossanyi E (2000). Design of closed loop controllers for wind turbines. Wind Energy 3: 149-163.

Burton T, *et al.* (2001). *Wind Energy Handbook*. Wiley.

Cerviño LI, Bewley TR (2003). On the extension of the complex-step derivative technique to pseudo-spectral algorithms. Journal of Computational Physics 187: 544-549.

Connell JR (1982). The spectrum of wind speed fluctuations encountered by a rotating blade of a wind energy conversion system. Solar Energy 29: 363-375.

D'Arco S, Suul JA, Fosso OB (2015). A virtual synchronous machine implementation for distributed control of power converters in SmartGrids. Electric Power Systems Research 122: 180-197.

Davidson PA (2004). *Turbulence – An Introduction for Scientists and Engineers*. Oxford University Press.

Endegnanew AG, *et al.* (2013). *Design procedure for inter-array electric design*. Report D2.2, EERA DTOC.

Felippa CA, Haugen B (2005). A unified formulation of small-strain corotational finite elements: I. Theory. Computer Methods in Applied Mechanics and Engineering 194: 2285-2335.

Hansen MH (2004a). Aeroelastic stability analysis of wind turbines using an eigenval-ue approach. Wind Energy 7: 133-143.

Hansen MH (2004b). A Beddoes-Leishman type dynamic stall model in state-space and indicial formulations. Report Risø-R-1354, Risø National Laboratory.

Hansen MH, *et al.* (2005). *Control design for a pitch-regulated, variable speed wind turbine*. Report Risø-R-1500(EN), Risø National Laboratory.

Hansen MH, Henriksen LC (2013). *Basic DTU Wind Energy Controller*. Report DTU Wind Energy E-0028, Technical University of Denmark.

Hansen MOL. *Aerodynamics of Wind Turbines*. Second Edition, Earthscan.

Jonkman J, *et al.* (2009). *Definition of a 5-MW reference wind turbine for offshore system development*. Report NREL/TP-500-38060, National Renewable Energy Laboratory.

Kristensen L and Frandsen S (1982). Model for power spectra of the blade of a wind turbine measured from the moving frame of reference. Journal of Wind Engineering and Industrial Aerodynamics 10: 249-262.

Kundur P (1994). *Power System Stability and Control*. McGraw-Hill.

Leishman JG (2002). Challenges in modelling the unsteady aerodynamics of wind turbines. Wind Energy 5: 85-132.

Leithead WE, Connor B (2000). Control of variable speed wind turbines: design task. International Journal of Control 73: 1189-1212.

Lovelace EC, Jahns TM, Lang JH (2002). A saturating lumped-parameter model for an interior PM synchronous machine. IEEE Transactions on Industry Applications 38: 645-650.

Martins JRRA, *et al.* (2003). The complex-step derivative approximation. ACM Transactions on Mathematical Software 29: 245-262.

Merz KO (2015a). Environmental loads for frequency-domain aeroelastic analysis of offshore wind turbines. Memo AN 15.12.19, SINTEF Energy Research.

Merz KO (2016). Reduction of Aerodynamic States in the STAS Wind Turbine Module. Memo AN 16.12.70, SINTEF Energy Research.

Merz KO (2018a). *STAS Aeroelastic 1.0 – Theory Manual*. Report 2018:00834, SINTEF Energy Research.

Merz KO, Pedersen MD (2018b). Offshore wind turbine controls. Chapter 5 of Anaya-Lara *et al.*, *Offshore Wind Energy Technology*. Wiley.

Merz KO (2019a). STAS Actuator 1.0 – Theory Manual. Memo AN 19.12.06, SINTEF Energy Research.

Merz KO (2019b). STAS Electric 1.0 – Theory Manual. Memo AN 19.12.07, SINTEF Energy Research.

Mulders SP, van Wingerden JW (2018). Delft Research Controller: an open-source and community-driven wind turbine baseline controller. Journal of Physics: Conference Series 1037: Paper 032009.

NEA (2016). *Borssele Wind Farm Zone, Wind Farm Sites III and IV – Project and Site Description*. Netherlands Enterprise Agency.

Press WH *et al.* (2007). *Numerical Recipes – The Art of Scientific Computing*. Third Edition, Cambridge University Press.

Snel H, Schepers JG (1995). *Joint Investigation of Dynamic Inflow Effects and Implementation of an Engineering Method*. Report ECN-C--94-107, Energy Research Centre of the Netherlands.

Stevens BL, Lewis FL (2003). *Aircraft Control and Simulation*. Second Edition, Wiley.

Sønderby IB (2013). *Low-order aeroelastic models of wind turbines for controller design*. PhD Thesis, Department of Wind Energy, Technical University of Denmark.

Thomassen PE, *et al.* (2012). A novel tool for FEM analysis of offshore wind turbines with innovative visualization techniques. Paper ISOPE-I-12-080, presented at the Twenty-second International Offshore and Polar Engineering Conference, 17-22 June, 2012, Rhodes, Greece.

Wijenayake AH, Schmidt PB (1997). Modeling and analysis of permanent magnet synchronous motor by taking saturation and core loss into account. Proceedings of the Second International Conference on Power Electronics and Drive Systems, Singapore, May 26-29, 1997.

Zhao R, *et al.* (2012). Fatigue distribution optimization for offshore wind farms using intelligent agent control. Wind Energy 15: 927-944.

Østergaard KZ, *et al.* (2007). Estimation of effective wind speed. Journal of Physics: Conference Series 75: Paper 012082.

Technology for a better society

www.sintef.no