Universität Rostock Traditio et Innovatio

Fault Diagnosis and Fault-Tolerant Control of Wind Turbines

Nonlinear Takagi-Sugeno and Sliding Mode Techniques

Sören Georg

Dissertation
zur Erlangung des Grades
Doktor-Ingenieur (Dr.-Ing.)

der Fakultät für Maschinenbau und Schiffstechnik
der Universität Rostock

Rostock, 2015

urn:nbn:de:gbv:28-diss2015-0018-2

Gutachter:

1. Gutachter: Prof. Dr.-Ing. Harald Aschemann,
Fakultät für Maschinenbau und Schiffstechnik, Universität Rostock

2. Gutachter: Prof. Dr.-Ing. Christoph Woernle,
Fakultät für Maschinenbau und Schiffstechnik, Universität Rostock

3. Gutachter: Prof. Dr.-Ing. Horst Schulte,
Fachbereich I, Hochschule für Technik und Wirtschaft Berlin

Promotionskommission:

Prof. Dr.-Ing. Frank-Hendrik Wurm (Vorsitzender), Universität Rostock

Prof. Dr.-Ing. Harald Aschemann, Universität Rostock

Prof. Dr.-Ing. Christoph Woernle, Universität Rostock

Prof. Dr.-Ing. Horst Schulte, Hochschule für Technik und Wirtschaft Berlin

Prof. Dr. Uwe Ritschel, Universität Rostock

Datum der Einreichung: 13. Mai 2014

Datum der Verteidigung: 4. Dezember 2014

Danksagung

> Πρῶτον μὲν εὐχαριστῶ τῷ ϑεῷ μου
> τῷ προετοιμάσαντι ταύτην τὴν ὁδόν,
> ἵνα ἐν αὐτῇ περιπατήσω.

Ich möchte mich zunächst herzlich bei Herrn Prof. Horst Schulte bedanken für das Vertrauen, das er in mich gesetzt hat, für die gute inhaltliche Betreuung meiner Arbeit und für die gute Zusammenarbeit in der Arbeitsgruppe Regelungstechnik der HTW Berlin in den vergangenen drei Jahren.

Herrn Prof. Harald Aschemann danke ich dafür, dass er mich als externen Doktoranden der Universität Rostock betreut und das erste Gutachten für meine Arbeit übernommen hat. Weiterhin danke ich Prof. Christoph Woernle für die Übernahme des zweiten Gutachtens, Prof. Frank-Hendrik Wurm für die Übernahme des Prüfungsvorsitzes und Prof. Uwe Ritschel als Mitglied der Promotionskommission.

Ein weiterer Dank geht an Heidemarie Brümmer von der Drittmittelverwaltung der HTW Berlin für die unkomplizierte Zusammenarbeit sowie an Anke Öhler vom Projektträger FZ Jülich für eine unbürokratische Betreuung des Forschungsprojekts.

Florian Pöschke, den ich zusammen mit Prof. Schulte bei seiner Abschlussarbeit betreut habe, verdanke ich interessante inhaltliche Diskussionen, von denen ich wiederum selbst profitiert habe.

Schließlich danke ich meinen Zimmerkollegen Nico Goldschmidt und Eckhard Gauterin für die gute Arbeitsatmosphäre und die gute projektübergreifende Zusammenarbeit.

Diese Dissertation ist entstanden im Rahmen eines vom Bundesministerium für Bildung und Forschung geförderten Forschungsprojektes (Förderkennzeichen 17N1411).

Abstract

This work deals with fault diagnosis and fault-tolerant control strategies for wind turbines, which are implemented in simulations using observer-based methods.
The thesis is divided in two main parts. In the first part **Modelling and Control**, the necessary groundwork is laid out concerning the dynamic wind turbine models, the rotor speed control and the estimation of an effective wind speed. This first part forms a basis for the second part **Fault Diagnosis and Fault-Tolerant Control**, where the fault diagnosis and fault-tolerant control schemes are described and applied to several fault scenarios.

Modelling and Control For the controller and observer designs, a dynamic nonlinear reduced-order model of the wind turbine with four degrees of freedom is first set up. The nonlinearities stem from the aerodynamics and are modelled by means of aero maps for the rotor torque and thrust.
The Takagi-Sugeno (TS) model structure, i.e. weighted sums of linear submodels, is used throughout this work to adequately deal with the nonlinearities. Two different methods are applied to obtain the TS structure from the underlying nonlinear model, namely fuzzy blending of Taylor-linearised submodels and the sector nonlinearity approach, which yields an exact representation of the original model. The first approach is used for the controller design, whereas the second is used for all observer designs in this work.
Based on the TS model obtained using local linearisations, a state-space controller in TS structure (Parallel Distributed Compensation) is designed for the rotor speed control in the full load region, where the linear sub-controllers are designed by means of the LQ optimality criterion.
In order to implement the observers for fault reconstruction, the current wind speed is needed. This quantity can be measured using a nacelle anemometer, which is routinely done in wind turbines. However, this is merely a single-point measurement, which is not an adequate estimate for the whole wind field acting on the rotor. Therefore, a TS observer is designed in this work to reconstruct an effective wind speed, which is later used as an input signal for the observers for fault reconstruction.

Fault Diagnosis and Fault-Tolerant Control The fault diagnosis task in this work is realised by means of direct fault reconstruction. To this end, a Takagi-Sugeno sliding mode observer (TS SMO) is designed. The so-called equivalent output injection signal, which corresponds to the average behaviour of the discontinuous switching term, can be directly evaluated to obtain estimates of the faults.
One important contribution in this work is a modified, weighted switching term for the TS SMO, which enables simultaneous reconstruction of faults of significantly differing orders of magnitude, which was not feasible using a TS SMO with the standard switching term.
As example faults, both actuator input and parameter faults as well as sensor faults are simulated and reconstructed. As simulation models, both the reduced-order nominal model and the aero-elastic code FAST by NREL are used. As for the actuator parameter faults, the direct fault reconstruction method with TS sliding mode observers has been applied for the first time in this work.
The reconstructed faults are then utilised to set up fault-tolerant control schemes for both actuator and sensor faults by means of direct fault compensation. Promising results are obtained in the simulation studies, where the active fault compensation in many cases nearly reproduces the behaviour of the fault-free simulations. The stability of the controller in the full load region including faults and active fault compensation is preserved, which is validated by analysing the derivatives of appropriate Lyapunov functions.

Contents

List of Figures

List of Symbols

Chapter 2: Wind Turbine Modelling

$\mathbf{A}$	System matrix
$\mathbf{B}$	Input matrix
$\mathbf{C}$	Output matrix
C_Q	Aerodynamic rotor torque coefficient
C_T	Aerodynamic rotor thrust coefficient
$\mathbf{D}$	Damping matrix
d_{11}	Aerodynamic damping parameter for tower fore-aft motion
d_B	Flap-wise blade damping constant
d_S	Drivetrain damping constant
d_T	Fore-aft tower damping constant
E_K	Kinetic energy
E_P	Potential energy
$\mathbf{f}$	Vector of external forces
$\mathbf{g}$	Vector of nonlinear system parts
F_{st}	Reference thrust force for aerodynamical calculations
F_T	Aerodynamic rotor thrust force
J_g	Generator inertia
J_r	Rotor inertia
$\mathbf{K}$	Stiffness matrix
$\tilde{\mathbf{K}}$	Stiffness matrix for model with torsion angle
k_B	Effective blade flap-wise stiffness parameter

k_S	Drivetrain stiffness parameter
k_T	Effective tower fore-aft stiffness parameter
L	Lagrangian function
$\mathbf{M}$	Mass matrix
m_B	Effective mass of blade-tip motion
m_{Blade}	Blade mass
m_{Hub}	Hub mass
m_{Nacelle}	Nacelle mass
m_{Rotor}	Rotor mass
m_T	Effective mass of nacelle-tower motion
m_{Tower}	Tower mass
N	Number of blades
n_g	Gearbox ratio
P_D	Dissipation function
$\mathbf{q}$	Vector of generalised coordinates
R	Rotor radius
r_B	Effective rotor radius for centrifugal blade stiffening
T_a	Aerodynamic rotor torque
T_g	Applied generator torque
$T_{g,d}$	Demanded generator torque
$\mathbf{u}$	Input vector
v	Wind speed
$\mathbf{x}$	State vector
y_B	Flap-wise blade tip displacement
$\mathbf{y}$	Output vector
y_T	Fore-aft tower top displacement
α	Parameter for centrifugal blade stiffening
β	Pitch angle
β_d	Demanded pitch angle

θ_g	Generator rotational angle
θ_r	Rotor rotational angle
θ_s	Shaft torsion angle
λ	Tip speed ratio
ρ	Air density
τ	Delay time constant for pitch dynamics
τ_g	Delay time constant for torque dynamics
ω_g	Generator angular velocity (generator speed)
ω_r	Rotor angular velocity (rotor speed)

Chapter 3: Takagi-Sugeno Models

$\mathbf{A}_i$	System matrices for TS submodels
$\mathbf{B}_i$	Input matrices for TS submodels
$\mathbf{C}_i$	Output matrices for TS submodels
f	Nonlinear function in TS model
h_i	Membership functions for TS model
N_l	Number of dissimilar nonlinearities in TS model
N_r	Number of TS submodels
w	Weighting functions
$\mathbf{z}$	Vector of premise variables

Chapter 4: Control

$\mathbf{a}$	Vector of affine system part
$\mathbf{A}_{\mathrm{CL},i}$	Closed-loop system matrices
$\mathbf{E}_i$	Disturbance matrices
$\mathbf{k}$	Vector of control gains
$\mathbf{P}$	Symmetric solution matrix for Lyapunov equation
V	Lyapunov function
x_I	Integrator state

$\omega_{r,\mathrm{SP}}$	Rotorspeed setpoint

Chapter 5: Wind Observer

$\mathbf{e}$	Output error vector
$\mathbf{L}$	Observer feedback matrix
$\bar{v}$	Mean wind speed
$\hat{\mathbf{x}}$	Reconstructed system vector
$\hat{\mathbf{y}}$	Reconstructed output vector
$\hat{\mathbf{z}}$	Reconstructed vector of premise variables
τ_v	Delay time constant for wind model

Chapters 6/7: Fault Diagnosis and Fault-Tolerant Control

$\boldsymbol{\mathcal{A}}$	Transformed system matrix for TS SM observer design
$\mathbf{A}_f$	Augmented system matrix for sensor fault reconstruction
$\boldsymbol{\mathcal{A}}_{22}^s$	Design matrix for TS SM observer design
$\boldsymbol{\mathcal{B}}$	Transformed input matrix for TS SM observer design
$\mathbf{D}$	Matrix of external disturbances
$\boldsymbol{\mathcal{D}}$	Transformed disturbance matrix for TS SM observer design
$\mathbf{e}_1$	Error vector of non-measurable system states
$\mathbf{e}_y$	Error vector of measurable system states
$\tilde{\mathbf{e}}_y$	Error vector of measurable system states including sensor faults
$\mathbf{F}$	Fault matrix
$\boldsymbol{\mathcal{F}}$	Transformed fault matrix for TS SM observer design
$\mathbf{f}_a$	Actuator fault vector
$\hat{\mathbf{f}}_a$	Reconstructed actuator fault vector
$\mathbf{f}_s$	Sensor fault vector
$\hat{\mathbf{f}}_s$	Reconstructed sensor fault vector
n	Number of system states
$\mathbf{N}$	Selection matrix for sensor fault reconstruction

p	Number of outputs
$\mathbf{P}_2$	Solution matrix of Lyapunov equation for TS SMO design
$\mathbf{T}_i$	Transformation matrices for TS SM observer
$\mathbf{u}_{\text{corr}}$	Corrected input vector
$\tilde{\mathbf{u}}_{\text{corr}}$	Corrected input vector with actuator faults
$\mathbf{W}$	Weighting matrix for sliding mode switching term
$\mathbf{x}_1$	Vector of non-measurable system states
$\mathbf{x}_a$	Augmented system vector for sensor fault reconstruction
$\tilde{\mathbf{y}}$	Output vector in the presence of sensor faults
$\mathbf{y}_f$	Augmented output vector for sensor fault reconstruction
α	Scaling constant for fault scenarios
δ	Scalar constant for equivalent output injection signal
ζ	Damping constant for 2nd-order pitch dynamics
$\boldsymbol{\nu}$	Discontinuous term for TS SM observer
$\boldsymbol{\nu}_{\text{eq}}$	Equivalent output injection signal
$\boldsymbol{\xi}$	Vector of unmodelled dynamics / external disturbances
$\hat{\boldsymbol{\xi}}$	Reconstructed vector of unmodelled dynamics / external disturbances
Ξ	Upper norm bound of combined disturbance and actuator fault vector
Ξ_{f_a}	Upper norm bound for actuator fault vector
Ξ_{ξ}	Upper norm bound for disturbance vector
ρ	Scalar gain factor for sliding mode switching term
$\boldsymbol{\rho}$	Gain matrix for sliding mode switching term
Ψ	Upper norm bound for sensor fault vector
Ψ_d	Upper norm bound for derivative of sensor fault vector
ω_n	Frequency parameter for 2nd-order pitch dynamics

1

Introduction

1.1 Motivation and Goal

Steadily increasing sizes and a growing complexity of wind turbines give rise to increasing demands regarding system safety and availability. The safety demands are commonly met by introducing redundancy in the system architecture, like additional sensors. Some of these system reduncancies are vital for a safe operation of wind turbines. The classic example is the pitch system for adjusting the angles of a rotor blade. For each of the three blades, one totally independent pitch system is used, such that in the worst case of a failure in one or two pitch systems, the remaining one or two would still be able to bring the turbine to a standstill. By contrast, other redundancies rather add to the "perceived" safety. For instance, a typical minimal sensor setup for rotor and generator speed measurements is one generator and two rotor speed sensors. For pure safety reasons, a second generator speed sensor would not be necessary, since in the case of a failure of the sole generator speed sensor, the two rotor speed sensors would suffice to detect an overspeed.
While cherishing additional gains in safety, however important, redundancies always generate additional costs and possibly additional turbine downtimes caused by faults in the redundant system parts. The plus in safety may thus entail a loss of produced energy and revenue. One possible remedy would be to exploit functional redundancies rather than hardware redundancies. This route has been taken in [Echavarria et al., 2012]. Even when reducing hardware redundancies, large wind turbines are prone to faults from diverse sources. Nowadays, many of these faults, even if not critical, often lead to turbine shutdowns, again for safety reasons. Especially in offshore wind turbines, this may result in a substantially reduced availability, because rough weather conditions may prevent a quick replacement of faulty or damaged system parts.

The need for a continuous energy production necessitates the quest for reliable and effective fault diagnosis and fault-tolerant control schemes that enable to keep the turbine in operation in the presence of faults, perhaps at reduced speeds, while organising the technical repair. Apart from increasing availability and reducing turbine downtimes, intelligent fault diagnosis schemes might also obviate the need for more hardware redundancy, if virtual sensors could at least partly take the place of redundant hardware sensors. The fault diagnosis schemes currently employed in wind turbines are typically on the level of the supervisory control, where commonly used strategies include sensor comparison, model comparison and thresholding [Johnson and Fleming, 2011]. These strategies enable a safe turbine operation, which involves shutdowns in case of critical faults, but no intelligent fault-tolerant turbine control on the operational control level.

The goal of this work is thus first to investigate and develop fault diagnosis schemes that achieve a precise and fast reconstruction of faults. Second, these reconstructed signals shall be utilised to set up fault-tolerant control strategies to obtain a system behaviour in the presence of faults which is close to the fault-free behaviour. Moreover, the diagnosis schemes should be capable of reconstructing faults of diverse types and from different sources, in order to avoid an assortment of various methods and frameworks. It would thus be desirable to have one common methodological framework for fault diagnosis.

1.2 Fault Diagnosis and Fault-Tolerant Control

In this section, an overview and a concise literature survey on fault diagnosis and fault-tolerant control (FTC) is given. First, the general concepts are explained, followed by an overview of different approaches applied to wind turbines.

1.2.1 Fault Detection and Diagnosis

Before overviewing different methods and approaches to fault detection and diagnosis, some terminology has to be introduced. A good starting point for this is [Isermann and Ballé, 1997], which suggests definitions based on discussions held by the IFAC SAFEPROCESS committee.

Fault: “An unpermitted deviation of at least one characteristic property or parameter of the system from the acceptable / usual / standard condition.“

A fault is thus to be distinguished from a failure:

Failure: “A permanent interruption of a system’s ability to perform a required function under specified operating conditions.“

An error constitutes a special type of fault:

Error: “A deviation between a measured or computed value (of an output variable) and the true, specified or theoretically correct value.“

Both errors and faults, as well as failures, may occur in technical systems. A failure normally leads to a system shutdown, unless the failure happens in a non-critical system part. An example for a non-critical failure in a wind turbine would be the failure of the anemometer[1] heating. Only for low temperatures this failure might lead to a failure of the anemometer itself (iced-over) and even in this case the loss of the wind speed measurement does not necessarily demand an instant shutdown, because it is not used for closed-loop control and can be momentarily replaced by other measurements (like the average pitch angle). Granted, if no second, redundant anemometer is installed, a shutdown should nevertheless be conducted if the wind measurement is not available for a period of more than 20 or 30 min. Other failures demand instant system shutdown like the failure of one of the three pitch actuators of a wind turbine. Though the turbine might still be kept in operation mode, discrepancies between the three rotor blade angles may lead to large asymmetrical loads on the rotor and thus to serious mechanical damages.

As with failures, faults may be critical and non-critical. To stay with the wind measurement example, an offset fault of the wind speed measurement of 20 % has no direct influence on the control behaviour of the wind turbine and may be accepted for some time being. By contrast, an offset fault of 20 % in one of the pitch actuators, unless adequately corrected, is critical because it would lead to asymmetrical rotor loads.

For both critical and non-critical faults, a safe system operation and thorough supervision demands adequate detection and diagnosis of faults. This stipulates another clarification of terminology (again from [Isermann and Ballé, 1997]).

Fault detection: “Determination of the faults present in a system and the time of detection.“

Fault isolation: “Determination of the kind, location and time of detection of a fault. Follows fault detection.“

Fault identification: “Determination of the size and time-variant behaviour of a fault. Follows fault isolation.“

Fault diagnosis: “Determination of the kind, size, location and time of detection of a fault. Follows fault detection. Includes fault isolation and identification.“

Fault detection is thus the most basic category, which gives only a yes/no information whether and when a fault (any fault) has occured. For most technical systems, such a crude information will not suffice. Fault isolation is at least required to decide upon an action like leaving the system in operation (non-critical

---

[1] An anemometer is used to measure the wind speed

faults) or stopping the system (critical faults). More advanced fault handling would involve fault-tolerant strategies (see section 1.2.2). While the information about a fault occurrence (fault detection) and the type and location of the fault (fault isolation) are sufficient for some control applications to achieve the desired fault-tolerant strategy, some FTC schemes require more information about the fault, in particular about the fault magnitude [Alwi et al., 2011], i.e. fault identification. A direct reconstruction of a fault would be the most valuable information to be used in fault-tolerant control. A full fault diagnosis scheme would also include a decision process, whether a fault (and which kind of fault) has occured and which action is to be performed in response. Though important for a practical use on a wind turbine, this part of fault diagnosis is beyond the scope of this thesis, which focusses on fault reconstruction and the use of the reconstructed signals for fault-tolerant control.

Methods for Fault Detection and Isolation

There are numerous methods available for fault detection. The most broad classification is into signal and model-based methods. If only output, but no input measurements are available, signal-based (or signal-model-based) methods are the sole methods at hand. These comprise for example analysis of periodic signals (bandpass filters, Fourier analysis, maximum-entropy estimation) and vibrational analysis of machines [Isermann and Ballé, 1997; Isermann, 2006].
By contrast, model-based methods require input and output measurements and can be broadly grouped into observer-based methods, parity relations, and parameter estimation schemes. A standard reference for model-based fault detection and isolation (FDI) methods is [Chen and Patton, 1999], which was also used as a main reference for the following thumbnail sketch of FDI methods.

Parameter estimation schemes make use of system identification techniques to obtain estimates of physical system parameters (mass, friction, viscosity, inductance, etc.). An input-output mathematical model is used for online comparison of the system behaviour with the initial fault-free behaviour. A database for different faults can thus be created. Parameter estimation schemes are very suitable for the detection and isolation of parameter faults yet have some disadvantages like long computation times, slow reaction to incipient faults and difficult fault isolation. Moreover, the detection and isolation of multiple faults is difficult. A commonly used technique is the least-squares parameter estimation (see for example the survey paper [Isermann, 1984]).

Most model-based FDI methods rely on the calculation and subsequent evaluation of so-called residuals. "Residuals are quantities that represent inconsistency between the actual system variables and the mathematical model" and "respond to faults in characteristic manners" [Chen and Patton, 1999]. A residual generator is a linear processor of system inputs and outputs. In a fault-free case, a residual should ideally be equal to zero, whereas in a faulty case, a value different from zero should be obtained. Residuals can be generated using observer-based methods or parity relations.

The approach to generate residuals using parity relations is based on consistency checking of input and output data over a time window [Chen and Patton, 1999]. For example, the residual (parity function) may take the form of a moving average process of the most recent sensor output and actuator input values [Chow and Willsky, 1984]. The parity relation approach is equivalent to observer-based approaches in certain cases, yet provides less design flexibility [Chen and Patton, 1999].
Among the advantages of observer-based methods are suitability for the detection and isolation of actuator or sensor faults with a fast reaction to incipient faults, and a systematic design procedure. Provided a sufficient number of measurements is available, they can handle multiple faults.

In real systems, not only faults (actuator/sensor/parameter faults) may influence the system, but also external disturbances. On top of that, modelling errors will have a degrading effect on the FDI results. One important issue in FDI problems is thus to achieve disturbance decoupling, i.e. to make the residual robust against external disturbances and uncertainties. Disturbance decoupling can be achieved by means of eigenstructure assignment [Patton et al., 1987; Patton and Chen, 1992, 1997] or by using an Unknown Input Observer (UIO) [Watanabe and Himmelblau, 1982; Chen and Zhang, 1991; Guan and Saif, 1991; Saif and Guan, 1993; Chen et al., 1996].

The underlying system structure for a system with actuator and sensor faults as well as an unknown disturbance is [Chen and Patton, 1999]

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}\,\mathbf{x} + \mathbf{B}\,\mathbf{u} + \mathbf{E}\,\mathbf{d} + \mathbf{F}\,\mathbf{f}_a \\ \dot{\mathbf{y}} &= \mathbf{C}\,\mathbf{x} + \mathbf{f}_s\,, \end{aligned} \tag{1.1}$$

where $\mathbf{x}$, $\mathbf{y}$ and $\mathbf{u}$ denote the system, output and input vectors, respectively, and $\mathbf{d}$ the unknown disturbance. $\mathbf{f}_a$ and $\mathbf{f}_s$ denote the actuator and sensor faults. $\mathbf{E}$ is the unknown input distribution matrix, which must be known for the design of the UIO. In many cases, if actuator input faults are considered, the actuator fault matrix $\mathbf{F}$ is equal to the input matrix $\mathbf{B}$.

For UIOs, the state estimation error $\mathbf{e} = \mathbf{x} - \hat{\mathbf{x}}$ approaches zero asymptotically, regardless of the unknown input. The output error is thus decoupled from the unknown inputs. The residual is typically a weighted output estimation error and must be insensitive to the disturbance, but sensitive to the fault to be detected. Fault isolation can be achieved via "structured residuals", which are designed to be sensitive to some faults but insensitive to others.

Fault Identification / Reconstruction

Standardly, Unknown Input Observers are used to generate residuals but no direct fault estimates. By contrast, [Saif and Guan, 1993] introduced a special UIO for direct fault reconstruction, which yields very good results and relies on only one observer, in contrast to many other approaches which require a bank of observers for different faults. However, strong existence conditions have to be met for the proposed UIO.

Following a different route, sliding mode observers have also been studied and applied for direct fault reconstruction [Edwards and Spurgeon, 1994, 1998; Edwards et al., 2000; Tan and Edwards, 2003]. This particular concept is sometimes called the Edwards-Spurgeon observer for fault reconstruction [Alwi et al., 2011]. The sliding mode concept is a nonlinear technique, which can be utilised for both controllers and observers. It includes a nonlinear switching term, which establishes and maintains a motion on a so-called sliding surface, where a reduced-order dynamics compared to the normal system applies [Utkin, 1977, 1992].
For sliding mode observers, the equivalent output injection signal, which describes the average behaviour of the switching term, can be evaluated to yield direct fault estimates. In [Edwards, 2004; Edwards and Tan, 2006a], the UIO from [Saif and Guan, 1993] was compared to the sliding mode observer from [Tan and Edwards, 2003] as to their fault reconstruction capabilities. The authors found that the sliding mode observer (SMO) performs better in cases where parametric uncertainty is included in the simulation, which means that the robustness properties of the SMO are better. Also, the structural requirements are not as strong as for the UIO.
The applicability of sliding mode observers for real-time systems has been shown in [Edwards and Tan, 2006b], where the observer was implemented on a laboratory dc motor rig and used to achieve sensor FDI and fault-tolerant control.

The Edwards-Spurgeon observer is best suited for linear systems, such that the nonlinear switching term solely accounts for the disturbances and faults. For a highly nonlinear model like a wind turbine, a structure like the Edwards-Spurgeon observer is not ideal, as the sliding term would then have to capture both the nonlinear plant dynamics and the influence of the faults.
For this reason, a nonlinear extension of the Edwards-Spurgeon observer was introduced in [Gerland et al., 2010b,a], based on a nonlinear Takagi-Sugeno structure (see Chapters 3 and 6). In this way, the nonlinear plant dynamics can be fully dealt with using the TS structure, such that the switching term is solely responsibly for the faults.

Due to limitations of the switching term, neither the Edwards-Spurgeon observer nor the TS SMO from [Gerland et al., 2010b,a] are able to simultaneously reconstruct faults whose orders of magnitude differ significantly. For this reason, a modified switching term for the TS SMO was introduced in [Georg and Schulte, 2014c] and used throughout this work for fault reconstruction (see Section 6.3).

1.2.2 Fault-Tolerant Control

There are several approaches to fault-tolerant control. For an overview of different methods, see for example the review papers [Patton, 1997] and [Zhang and Jiang, 2003] or the book [Blanke et al., 2006]. Fault-tolerant control (FTC) can be broadly categorised into passive (PFTC) and active approaches (AFTC). Passive FTC falls mainly into the category of robust control, where a controller is designed such that it fulfills certain performance and stability criteria in the presence of disturbances and modelling uncertainties. Analogously, in passive FTC, a controller is designed such a restricted class of faults is tolerated by the controller. The disadvantage of PFTC approaches is that they have only limited fault-tolerance capabilities [Patton, 1997] and that the controller works suboptimally in the fault-free case.

For more significant faults, an active FTC strategy is needed, which involves prior fault diagnosis. In AFTC methods, one can distinguish between fault accomodation and control reconfiguration. Fault accomodation means the adaptation of the controller parameters to the faulty plant [Blanke et al., 2006]. Control reconfiguratio may involve the use of a different control structure altogether, like different inputs and outputs. Control reconfiguration is the only option if only fault detection and isolation, but no fault identification is possible. In such a case, the faulty system parts like sensors or actuators must be deactivated and replaced by others, which may also be virtual sensors or actuators [Blanke et al., 2006].

Even if a fault identification/reconstruction is possible, a control reconfiguration which leaves the original controller parameters unchanged may be advantageous compared to fault accomodation. A pure adaptation of controller parameters may enable a safe operation but not necessarily with a satisfying performance. By contrast, if the fault can be compensated for, the original controller, which is normally optimised for the fault-less system, can still be used for the faulty system.

In this work, a simple fault compensation approach is implemented by subtracting the reconstructed faults obtained from either the (faulty) inputs or outputs, i.e.

$$\mathbf{u}_{\text{corr}} = \mathbf{u} - \hat{\mathbf{f}}_a \,, \qquad \mathbf{y}_{\text{corr}} = (\mathbf{y} + \mathbf{f}_s) - \hat{\mathbf{f}}_s \,. \tag{1.2}$$

These corrected inputs or outputs than act as virtual actuators or sensors respectively, such that the originally designed controller can still be used. This is a pragmatic approach that requires no additional or integrated design of the fault-tolerant controller.

The fault compensation in (1.2) is a feedforward strategy without any feedback and relies upon the quality of the fault reconstruction. It is also possible to include a feedback term in the fault-tolerant controller, which compares the states or outputs of the actual with those of a fault-free reference system. This approach is called model-following [Patton, 1997]. In the case of wind turbines, however, it would not be clear what the reference system should be. It could only be a reduced-order state-space model (see Section 2). Such a system is very useful for controller and observer designs, yet is only an approximation of the real turbine dynamics and probably not appropriate to be used as a reference model in a feedback term for fault-tolerant control.

Despite its simplicity and reliance upon the fault reconstruction quality, it was thus decided in this work to use the fault compensation strategy according to (1.2), which yielded reasonable and in some cases very good results in simulations (see Chapter 7).

1.2.3 Fault Diagnosis and Fault-Tolerant Control for Wind Turbines

Fault Diagnosis

Besides basic fault detection strategies (see for example [Johnson and Fleming, 2011]), condition monitoring systems (CMS) are becoming increasingly common in wind turbines. CMS systems utilise signal-based methods like vibrational analysis to predict structural damages in components like bearings or the gearbox (see the fairly comprehensive survey paper [Hameed et al., 2009]). Though very important, these kind of structural damages, which are only detectable by means of signal-based methods, are not the focus of this work, but rather actuator and sensor faults.

In the recent years, advanced FDI methods for actuator and sensor faults have been successfully applied in wind turbine simulation studies. In [Wei and Verhaegen, 2011], blade root moment sensor faults and pitch actuator stuck faults are detected using a robust H_∞/H_- observer. The fault magnitudes are then estimated with a dynamical filter. For testing the algorithms, a linearised simulation model is used.

Many recent papers have tested their FDI methods with a wind turbine benchmark model first presented in [Odgaard et al., 2009] and later described in more detail in [Odgaard et al., 2013]. It is a reduced-order model implemented in MATLAB/Simulink® and includes a torsional-flexible drivetrain as well as delay models for pitch angle and converter dynamics. In contrast to the model presented in Chapter 2 of this work, no tower and blade degrees of freedom are included. The benchmark model includes a list of pre-defined actuator and sensor faults as well as detection requirements.

In this work, the FDI benchmark model from [Odgaard et al., 2009, 2013] was not chosen to test the FDI and FTC schemes for two reasons. First, as a reduced-order model with only drivetrain degrees of freedom and actuator dynamics, it was not considered realistic enough to be taken as a sole simulation model. Instead, the aero-elastic simulation package FAST [Jonkman and Buhl Jr., 2005] by NREL was used, which is a well-established tool for wind turbine load simulations in academia as well as industry. FAST features 24 degrees of freedom and incorporates the module AeroDyn [Laino and Hansen, 2002] for the aerodynamical calculations. The NREL 5 MW reference turbine [Jonkman et al., 2009] was chosen as a wind turbine reference model in all simulations in this work.
Second, as FAST is not useful as a control design model due to its comparably high order, an effort was made in this work to obtain an appropriate control design model adapted to the NREL 5 MW turbine, which comprises the most important degrees of freedom (see Chapter 2). In particular, aero maps for the rotor thrust and torque coefficient were derived and approximated by means of nonlinear analytical functions, which were later used for controller design in Chapter 4.
First simulation results with a comparison between FAST and the 4-DOF model presented in Chapter 2 were presented in [Georg et al., 2012]. In [Georg, Müller, and Schulte, 2014], FAST was used to test a TS observer for wind speed reconstruction. Later on, the FDI benchmark model [Odgaard et al., 2009, 2013] was enhanced in [Odgaard and Johnson, 2013], where FAST was chosen as a simulation model, too. This shows that the route taken in this work to choose FAST from the start was the right one to follow.

One other aspect that has to mentioned in connection with the reduced-order benchmark model in [Odgaard et al., 2009] is that a double redundancy is assumed for all sensors (rotor speed, generator speed, pitch angles), and in the sensor fault scenarios only one of the two sensors for one quantity is affected by a fault at a time. This means that if a fault occurs in one rotor speed measurement, there is still another correct rotor speed sensor available, equally for generator speed and pitch angle measurements. By contrast, in FAST, only one sensor is available for each measurement. The double sensor redundancy assumption in [Odgaard et al., 2009] thus constitutes a significant relaxation of the FDI demands. This has to be borne in mind when comparing results from works that use the benchmark model [Odgaard et al., 2009] for validation with those that use FAST.

In [Odgaard and Stoustrup, 2012b], the results of a wind turbine FDI competition are summarised, where several actuator and sensor faults had to be detected within the first benchmark model [Odgaard et al., 2009]. The presented solutions range from data-driven to observer-based methods. Most of these solutions rely on the evaluation of residuals, and achieve fault detection and isolation (see [Blesa et al., 2011; Chen et al., 2011; Dong and Verhaegen, 2011; Kiasi et al., 2011; Laouti et al., 2011; Ozdemir et al., 2011; Pisu

and Ayalew, 2011; Simani et al., 2011], [Simani, Castaldi, and Bonfè, 2011], [Stoican et al., 2011; Svärd and Nyberg, 2011; Zhang et al., 2011]). In none of these solutions, however, a direct fault reconstruction is implemented.
The competition was later extended to include FTC. From the solutions presented in [Casau et al., 2012; Kim et al., 2012; Simani and Castaldi, 2012b,a; Yang and Maciejowski, 2012; Rotondo et al., 2012; Sami and Patton, 2012d,a], fault estimation solutions (for some faults) are included in [Yang and Maciejowski, 2012], [Rotondo et al., 2012] and [Sami and Patton, 2012d,a]. In [Yang and Maciejowski, 2012], an extended Kalman Filter is used as one module for a model-predictive pre-compensator and estimates of a scaling as well as a constant pitch angle fault are obtained. Estimates for pitch angle faults (offset and parameter faults of the actuator dynamics) are also obtained in [Rotondo et al., 2012], where a parameter estimation scheme (Block Least-Squares Algorithm) is utilised. In terms of the basic methodologies, the works presented in [Sami and Patton, 2012d,a] probably come closest to the methods in this work, as the observer designs are based on a nonlinear TS model and the applied observers (Fuzzy Fast Adaptive Estimator, Proportional Multiple Integral Observer) yield direct reconstructions of fault signals.

For the FAST FDI benchmark model [Odgaard and Johnson, 2013], FDI and FTC solutions were presented in [Badihi and Zhang, 2013; Jain et al., 2013; Luo et al., 2013; Sheibat-Othman et al., 2013; **?**; Zeng et al., 2013]. From among these works, only [Sheibat-Othman et al., 2013] present fault estimation, though only for pitch dynamics parameter faults. The results, however, are not convincing. In the other 5 works, fault detection and isolation, yet no fault reconstruction is achieved.

The TS sliding mode observer introduced in [Gerland et al., 2010b,a] was succesfully applied in [Schulte et al., 2012] to a pitch system, i.e. a component subsystem of a wind turbine, to achieve sensor fault reconstruction of pitch angle sensor faults. The modified TS SMO with the weighted switching term introduced in [Georg and Schulte, 2014c] was applied in the same work to simultaneously reconstruct pitch angle and generator torque actuator faults. In [Georg and Schulte, 2014a], the same observer was applied for the reconstruction of actuator parameter faults. Though the obtained parameter reconstructions are not yet of sufficient quality to be used in FTC schemes, the may suffice for fast fault detection schemes and show the breadth of possible applications of the TS SMO (see Section 6.4.2).
The results presented in [Georg and Schulte, 2014c] were the first where direct fault reconstruction for wind turbines was applied in simulations using FAST. Using the TS SMO for fault reconstruction also has the advantage that several types of faults (additive, scaling, stuck faults) can be reconstructed with one and the same observer and directly applied for fault compensation. In this way, only two observers, one for actuator and one for sensor faults, are needed (see Chapters 6 and 7).

Fault-Tolerant Control

When it comes to fault-tolerant control for wind turbines, different approaches have also been applied and tested in simulation studies. In [Sloth et al., 2011], passive and active fault-tolerant controllers are compared with a robust controller in their effectiveness to accommodate altered actuator dynamics in the (hydraulic) pitch system model. The different controllers are designed for the full-load region based on linear parameter varying (LPV) models. The nonlinear simulation model largely corresponds to the one presented in Chapter 2 in this work.

Later publications on FTC for wind turbines mainly used the FDI benchmark model [Odgaard et al., 2009, 2013]. In [Sami and Patton, 2012b], a passive sensor FTC strategy is implemented based on a sliding mode controller for the partial load region that tolerates generator speed sensor faults and generator torque offset faults. In [Odgaard and Stoustrup, 2012a], an FTC scheme is set up based on a bank of Unknown Input observers. However, each single fault scenario requires a different observer design. In [Sami and Patton, 2012d,a], a fast adaptive fuzzy estimator respectively a proportional multiple integral observer are used to achieve a fault-tolerant behaviour in the partial load region. In [Rotondo et al., 2012], a parameter estimation scheme is used for obtaining fault estimates, which are then utilised in a virtual actuator/sensor fault-tolerant control strategy.

As for FTC solutions using the FAST FDI benchmark model [Odgaard and Johnson, 2013], [Badihi and Zhang, 2013] present a Fuzzy Modelling and Identification (FMI) technique, which is applied to achieve

FTC. However, the algorithm requires an initial training with a pre-defined dataset, which seems cumbersome. Also, for a real turbine, there would be no pre-defined dataset available, due to the stochastic wind behaviour. In [Jain et al., 2013], a projection-based approach for wind turbine FTC is presented but no FDI module and thus no explicit fault reconstruction. In [Luo et al., 2013], a Youla parameterisation of stabilising controllers is used with an embedded residual generator for fault detection. Again, no direct fault reconstruction is implemented.

In this work, an FTC scheme for actuator and sensor faults in wind turbines is presented based on direct fault reconstruction by means of a Takagi-Sugeno sliding mode observer [Gerland et al., 2010b,a; Georg and Schulte, 2014c]. This concept was applied in [Georg and Schulte, 2013] to achieve fault-tolerant control for wind turbines in the presence of actuator faults in the pitch angle and generator torque. In [Georg and Schulte, 2014b], the same concept is also used for sensor fault-tolerant control.

1.3 Outline of the Thesis

This thesis is structured in two main parts, **Modelling and Control** and **Fault Diagnosis and Fault-Tolerant Control**, where part I forms the basis for part II.

Part I: Modelling and Control The first part starts with Chapter 2, where a nominal reduced-order wind turbine model for the purpose of controller and observer design is studied, including its derivation from Lagrangian dynamics. In Section 2.3, the model parameters of the NREL 5 MW reference turbine [Jonkman et al., 2009] are adapted for the reduced-order model. Particularly important is to obtain the aero maps for the rotor thrust and torque, which is laid out in Section 2.3.3.

Throughout this thesis, the Takagi-Sugeno (TS) model structure [Takagi and Sugeno, 1985] is used as a framework to deal with system nonlinearities. In Chapter 3, the basic TS structure and the sector-nonlinearity approach to obtain an exact representation of a nonlinear model are presented. Furthermore, the nominal wind turbine model is derived in TS structure.

Chapter 4 is concerned with the design of the rotor speed controllers for the full load and transition region of a wind turbine. For the latter (linear) region, the generator torque is used as control input ("Torque Control") and a linear LQR design is performed (see Section 4.2). For the nonlinear full load region, the collective pitch angle is used as control input ("Pitch Control"). For this region, a PDC controller (Parallel Distributed Compensation) is used (Section 4.1.2) which is based on Taylor-linearised models at several stationary points. The PDC controller is in the same TS structure as the underlying design model. The linear subcontrollers at each local point are designed using the LQ criterion (Section 4.1.3 and 4.1.4). A global stability analysis for the full load region is conducted afterwards. In Section 4.3, a second-order notch filter is designed, which is needed if the generator speed is used as controlled variable instead of the rotor speed in simulation studies.

Chapter 5 deals with the design of a TS observer for the estimation of a rotor effective wind speed. The underlying observer model is obtained by means of the sector-nonlinearity approach, such that the observer can be used in the whole operating range. The wind speed is added as an additional state using a simple first-order delay model (Section 5.1.1) such that it can be directly reconstructed. The observer is designed by means of an optimal LMI criterion (Section 5.1.2). Simulation results are given in Section 5.2.

Part II: Fault Diagnosis and Fault-Tolerant Control In Chapter 6, observer-based methods for fault diagnosis are described. In particular, a Takagi-Sugeno Sliding mode observer (introduced in Section 6.1) is used in this work for direct reconstruction of faults (Section 6.2). Since a simultaneous reconstruction of faults of significantly differing orders of magnitude was not possible with the standard switching term of a TS Sliding Mode Observer, a modified, weighted switching term is introduced in this work (Section 6.3). It can be shown that an ideal sliding motion can be established in finite time when using the modified switching term (Section 6.3.2). In Sections 6.4 and 6.5, some examples for actuator and sensor faults of different fault types (additive, scaling, stuck faults) are simulated and reconstructed.

In Chapter 7, the fault-tolerant control strategies for actuator (Section 7.1) and sensor (7.2) faults are introduced and tested in simulations with several faults of different types. A global stability analysis in the full load region is conducted for both actuator (7.1.3) and sensor faults (7.2.2) to check whether the stability is ensured in the presence of faults and active fault compensation.

Finally, a conclusion, an overview of new contributions to the research field, and an outlook on possible future work are given in Chapter 8.

Part I

Modelling and Control

2

Wind Turbine Modelling

2.1 Simulation Model

Prior to testing any new control strategies on a real wind turbine, the efficacy of a scheme has to be tested in detailed aero-elastic simulation studies. Several simulation packages exist that are commonly used in academia and industry for wind turbine load simulation. In this work, the simulation package FAST [Jonkman and Buhl Jr., 2005] by NREL is used as a reference simulation evironment and the NREL 5 MW turbine [Jonkman et al., 2009] is taken as a reference model for all simulation studies.

2.2 Control Design Model

FAST is a detailed wind turbine model with 24 degrees of freedom. As such, it is appropriate for testing the developed control algorithms but not for control design. For the latter purpose, a reduced-order dynamic wind turbine model, which captures only dynamic effects directly influenced by the control, is derived in this section and will be used in several variants for model-based control and observer design. With slight modifications, it corresponds to the model presented in [Bianchi et al., 2007], yet originates from [Bindner, 1999].

The complete model consists of several submodels for the mechanical structure, the aerodynamics, as well as the dynamics of the pitch system and the generator/converter system (see figure 2.1). The generator/converter dynamics can be modelled as a first order delay system. However, as the delay time constant is very small, in some simulation examples in this work, an ideal converter is assumed, with the result that the reference generator torque signal is equal to the actual generator torque. In these cases, the generator torque is therefore treated as a system input.

2.2.1 Mechanical Model

A mechanical wind turbine model with four degrees of freedom is considered, where those degrees of freedom are included that are most strongly affected by the wind turbine control:

1. Fore-aft tower bending
2. Flap-wise blade bending
3. Rotor rotation
4. Generator rotation

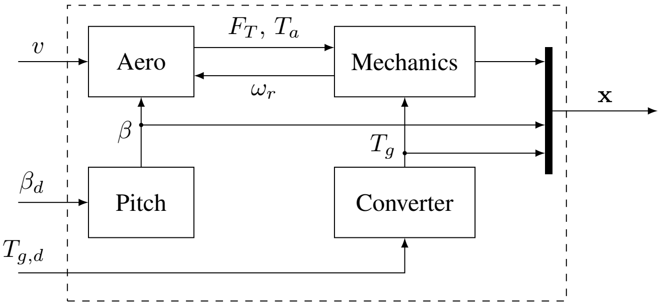


Figure 2.1: Schematics of the submodels of the complete wind turbine model with their respective inputs and outputs. v: wind speed; F_T, T_a: rotor thrust force and rotor torque; ω_r: rotor angular velocity; $\mathbf{x}$: state vector; T_g: generator torque; $T_{g,d}$: demanded generator torque; β: pitch angle; β_d: demanded pitch angle

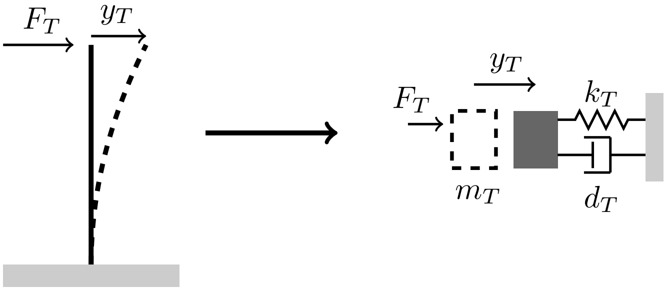


Figure 2.2: Tower bending (left) and mechanical model with translational spring and damper (right). Left subfigure reprinted from [Georg et al., 2012] © 2012 IEEE.

Tower and blade bending are not modelled by means of bending beam models, but only the translational displacement of the tower top and the blade tip, respectively, are considered, where the bending stiffness parameters are transformed to equivalent translational stiffness parameters (see figs. 2.2 and 2.3. For the tower, the equivalent translational stiffness parameter is derived by means of a direct stiffness method common in structural mechanics calculations (see section 2.3.2).

Upon a movement of the tower, the blades move, too. Therefore, the blade tip displacement is considered in the moving tower coordinate system and the tower motion must be taken into account for the derivation of the kinetic energy of the blade (see below in Subsection „Derivation of the Equations of Motion“). The moving tower coordinate system is also indicated by the arrows with label $\dot{y}_T$ in Figure 2.3.

The force F_T acts both on the tower and on N blades. Only one collective blade degree of freedom is considered. The N blade degrees of freedom would have to be considered individually if control strategies for load reduction involving individual blade pitch control were to be designed.

The assumption that the same external force F_T acts both on the tower degree of freedom and on the blade degree of freedom (with N blades) is a simplification. It is reasonable, however, because the rotor thrust force, which is caused by the aerodynamic lift forces acting on the blade elements, acts on the tower top and moreover causes a distributed load on each blade. This distributed load causes a bending of the blade, which could be modelled as a bending beam. A beam subjected not to a distributed load but to a concentrated load at the upper point (as in the model) must have a higher bending beam stiffness, in order that the same displacement results at the upper point. In the reduced-order wind turbine model, only the blade tip displacement will be considered, which necessitates the transition from the bending stiffness to a translational stiffness. To obtain an adequate translational stiffness constant, the bending stiffness of the bending beam must thus be larger than in the case with a distributed load.

The drivetrain consisting of rotor, shaft and generator is modelled as a two-mass inertia model including shaft torsion, where the two inertias are connected with a torsional spring with spring constant k_S and a

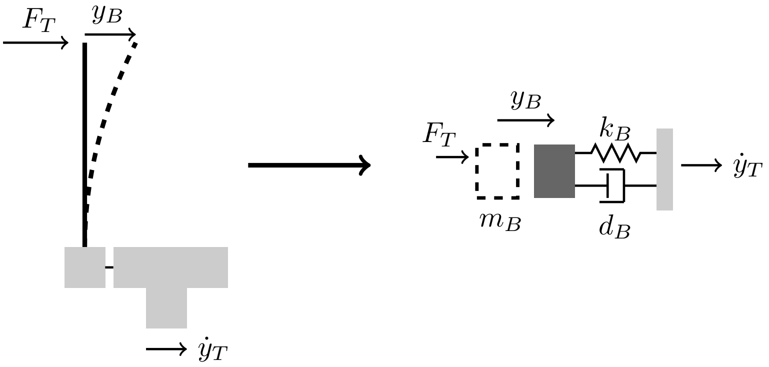


Figure 2.3: Blade bending in the moving tower coordinate system (left) and mechanical model with translational spring and damper (right). Left subfigure reprinted from [Georg et al., 2012] © 2012 IEEE.

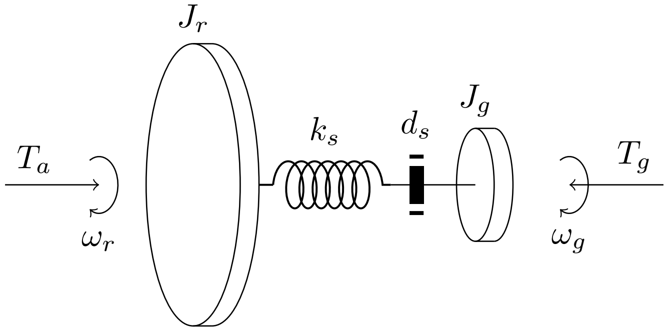


Figure 2.4: Model of the drivetrain. The angular velocities ω_r and ω_g are the time-derivatives of the rotation angles θ_r and θ_g; figure reprinted from [Georg et al., 2012] © 2012 IEEE.

torsional damper with damping constant d_S (see figure 2.4). The rotor is acted upon by the rotor torque T_a, which is caused by the lift forces on the individual blade elements. The applied generator torque T_g acts on the generator. An ideal gearbox with gearbox ratio 1 is considered, which simplifies the equations of motion for the drivetrain. For the purpose of comparing simulation results of the reduced-order model with simulation results of the NREL 5 MW reference turbine using FAST, where a gearbox ratio is included, a gearbox ratio can be implicitly considered in the model parameter for the generator inertia J_g by multiplying the given generator inertia of the reference turbine by the square of the gearbox ratio (see section 2.3).

Derivation of the Equations of Motion

The equations of motion are derived by means of Lagrangian dynamics, which first requires to define the generalised coordinates and generalised external forces. Afterwards, the energy terms of the system are deduced, from which the equations of motion are derived.

The vector of generalised coordinates is given by: $\mathbf{q} := (y_T \quad y_B \quad \theta_r \quad \theta_g)^T$

and the vector of external forces is defined by: $\mathbf{f} := (F_T \quad F_T \quad T_a \quad -T_g)^T$.

The generalised force F_T in the model is the rotor thrust force, which can be calculated from the wind speed at the blade and from the aero map of the thrust coefficient. The generalised force T_a is given by the aerodynamic rotor torque, which can be calculated from the wind speed and from the aero map of the torque coefficient (see section 2.2.4).

Taking account of the tower motion, the complete blade tip displacement is: $y_{B,\text{ges}} := y_T + y_B$

The kinetic energy is thus given by

$$\begin{aligned} E_K &= \frac{1}{2}\, m_T\, \dot{y}_T^2 \,+\, \frac{1}{2}\, N\, m_B\, \dot{y}_{B,ges}^2 \,+\, \frac{1}{2}\, J_r\, \dot{\theta}_r^2 \,+\, \frac{1}{2}\, J_g\, \dot{\theta}_g^2 \\ &= \frac{1}{2}\, m_T\, \dot{y}_T^2 \,+\, \frac{1}{2}\, N\, m_B\, \left(\dot{y}_T \,+\, \dot{y}_B\right)^2 \,+\, \frac{1}{2}\, J_r\, \dot{\theta}_r^2 \,+\, \frac{1}{2}\, J_g\, \dot{\theta}_g^2 \,. \end{aligned} \tag{2.1}$$

The potential energy can be obtained as

$$E_P \,=\, \frac{1}{2}\, k_T\, y_T^2 \,+\, \frac{1}{2}\, N\, k_B\, y_B^2 \,+\, \frac{1}{2}\, k_S\, \left(\theta_r \,-\, \theta_g\right)^2 \,. \tag{2.2}$$

The dampings in the system give rise to generalised friction forces, which can be written as derivatives of a quadratic form, the dissipation function [Landau and Lifshitz, 1976]. For the model, it is given by

$$P_D \,=\, \frac{1}{2}\, d_T\, \dot{y}_T^2 \,+\, \frac{1}{2}\, N\, d_B\, \dot{y}_B^2 \,+\, \frac{1}{2}\, d_S\, \left(\dot{\theta}_r \,-\, \dot{\theta}_g\right)^2 \,. \tag{2.3}$$

The Lagrangian equations of 2nd order including dissipation are given by [Landau and Lifshitz, 1976]

$$\frac{\mathrm{d}}{\mathrm{dt}}\left(\frac{\partial L}{\partial \dot{q}_i}\right) \,-\, \frac{\partial L}{\partial q_i} \,=\, f_i \,-\, \frac{\partial P_D}{\partial \dot{q}_i}\,, \tag{2.4}$$

where the Lagrangian function L denotes the difference between kinetic and potential energy. As the kinetic energy in equation (2.1) does not depend on the generalised coordinates and the potential energy in equation (2.2) does not depend on the generalised velocities, one obtains

$$\frac{\mathrm{d}}{\mathrm{dt}}\left(\frac{\partial E_K}{\partial \dot{q}_i}\right) \,+\, \frac{\partial E_P}{\partial q_i} \,+\, \frac{\partial P_D}{\partial \dot{q}_i} \,=\, f_i\,. \tag{2.5}$$

Explicitly calculating the derivatives in equation (2.5) for each of the four degrees of freedom yields four equations of motion, which describe the dynamics of the mechanical system:

$$\begin{aligned} \left(m_T \,+\, N m_B\right)\, \ddot{y}_T \,+\, N m_B\, \ddot{y}_B \,+\, d_T\, \dot{y}_T \,+\, k_T\, y_T &= F_T && (2.6)\\ N m_B\, \ddot{y}_T \,+\, N m_B\, \ddot{y}_B \,+\, N d_B\, \dot{y}_B \,+\, N k_B\, y_B &= F_T && (2.7)\\ J_r\, \ddot{\theta}_r \,+\, d_S\, \left(\omega_r \,-\, \omega_g\right) \,+\, k_S\, \left(\theta_r \,-\, \theta_g\right) &= T_a && (2.8)\\ J_g\, \ddot{\theta}_g \,-\, d_S\, \left(\omega_r \,-\, \omega_g\right) \,-\, k_S\, \left(\theta_r \,-\, \theta_g\right) &= -T_g && (2.9) \end{aligned}$$

This system of equations can be written in matrix form as

$$\mathbf{M}\,\ddot{\mathbf{q}} \,+\, \mathbf{D}\,\dot{\mathbf{q}} \,+\, \mathbf{K}\,\mathbf{q} \,=\, \mathbf{f}\,, \tag{2.10}$$

where the mass matrix $\mathbf{M}$, the damping matrix $\mathbf{D}$ and the stiffness matrix $\mathbf{K}$ are given by

$$\mathbf{M} \,=\, \begin{pmatrix} m_T + N m_B & N m_B & 0 & 0 \\ N m_B & N m_B & 0 & 0 \\ 0 & 0 & J_r & 0 \\ 0 & 0 & 0 & J_g \end{pmatrix}, \tag{2.11}$$

$$\mathbf{D} = \begin{pmatrix} d_T & 0 & 0 & 0 \\ 0 & N d_B & 0 & 0 \\ 0 & 0 & d_S & -d_S \\ 0 & 0 & -d_S & d_S \end{pmatrix}, \tag{2.12}$$

$$\mathbf{K} = \begin{pmatrix} k_T & 0 & 0 & 0 \\ 0 & N k_B & 0 & 0 \\ 0 & 0 & k_S & -k_S \\ 0 & 0 & -k_S & k_S \end{pmatrix}. \tag{2.13}$$

Model Including Gearbox Ratio

For the derivation of the equations of motion in section 2.2.1 a turbine without gearbox was assumed, such that rotor and generator speed are of the same order of magnitude. Comparison with simulations using FAST, where a gearbox ratio is included, can be achieved by means of a factor in the parameter J_g for the generator inertia (see remark on page 15 and section 2.3).

In some cases, though, it might be useful to include a gearbox ratio n_g directly into the model equations. In that case, the following expressions for the energy terms and the dissipation function are obtained, where only the drivetrain degrees of freedom are considered, since the tower and blade degrees of freedom remain unchanged:

$$E_K = \frac{1}{2} J_r \, \omega_r^2 + \frac{1}{2} J_g \, \omega_g^2 \tag{2.14}$$

$$E_P = \frac{1}{2} k_S \left(\theta_r - \frac{1}{n_g} \theta_g \right)^2 \tag{2.15}$$

$$P_D = \frac{1}{2} d_S \left(\omega_r - \frac{1}{n_g} \omega_g \right)^2 . \tag{2.16}$$

Evaluating the Lagrangian equations (2.5) yields the following equations of motion for the rotational degrees of freedom.

$$\dot{\omega}_r = \frac{1}{J_r} \left(-d_S \left(\omega_r - \frac{1}{n_g} \omega_g \right) - k_S \left(\theta_r - \frac{1}{n_g} \theta_g \right) + T_a \right) \tag{2.17}$$

$$\dot{\omega}_g = \frac{1}{J_g} \left(\frac{k_S}{n_g} \left(\theta_r - \frac{1}{n_g} \theta_g \right) + \frac{d_S}{n_g} \left(\omega_r - \frac{1}{n_g} \omega_g \right) - T_g \right) . \tag{2.18}$$

Including tower and blade dynamics, the system equations can be written in matrix form as $\mathbf{M} \, \ddot{\mathbf{q}} + \mathbf{D} \, \dot{\mathbf{q}} + \mathbf{K} \, \mathbf{q} = \mathbf{f}$, with

$$\mathbf{M} = \begin{pmatrix} m_T + N m_B & N m_B & 0 & 0 \\ N m_B & N m_B & 0 & 0 \\ 0 & 0 & J_r & 0 \\ 0 & 0 & 0 & J_g \end{pmatrix}, \quad \mathbf{D} = \begin{pmatrix} d_T & 0 & 0 & 0 \\ 0 & N d_B & 0 & 0 \\ 0 & 0 & d_S & -\frac{d_S}{n_g} \\ 0 & 0 & -\frac{d_S}{n_g} & \frac{d_S}{n_g^2} \end{pmatrix},$$

$$\mathbf{K} = \begin{pmatrix} k_T & 0 & 0 & 0 \\ 0 & N k_B & 0 & 0 \\ 0 & 0 & k_S & -\frac{k_S}{n_g} \\ 0 & 0 & -\frac{k_S}{n_g} & \frac{k_S}{n_g^2} \end{pmatrix}.$$

State-Space Model

The 2nd-order system of differential equations (2.10) can be transformed to a state-space model (1st-order system of differential equations) by introducing the state vector $\mathbf{x} := (\mathbf{q} \quad \dot{\mathbf{q}})^T$. To this end, equation (2.19) is solved for the second time derivative of the coordinate vector $\mathbf{q}$:

$$\begin{aligned} & \mathbf{M}\,\ddot{\mathbf{q}} + \mathbf{D}\,\dot{\mathbf{q}} + \mathbf{K}\,\mathbf{q} = \mathbf{f} \\ \Leftrightarrow \quad & \ddot{\mathbf{q}} = -\mathbf{M}^{-1}\mathbf{D}\,\dot{\mathbf{q}} - \mathbf{M}^{-1}\mathbf{K}\,\mathbf{q} + \mathbf{M}^{-1}\,\mathbf{f}\,. \end{aligned} \tag{2.19}$$

The state-space model can thus be obtained as

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}_{\text{mech}}\,\mathbf{x} + \mathbf{B}_{\text{mech}}\,\mathbf{u}_{\text{mech}} \\ \mathbf{y} &= \mathbf{C}_{\text{mech}}\,\mathbf{x}\,, \end{aligned} \tag{2.20}$$

with the state vector $\mathbf{x} = (y_T \quad y_B \quad \theta_r \quad \theta_g \quad \dot{y}_T \quad \dot{y}_B \quad \omega_r \quad \omega_g)^T$,
the input vector $\mathbf{u}_{\text{mech}} = (F_T \quad T_a \quad T_g)^T$, the system matrix
$\mathbf{A}_{\text{mech}} = \begin{pmatrix} \mathbf{0}_{4\times4} & \mathbf{I}_{4\times4} \\ -\mathbf{M}^{-1}\mathbf{K} & -\mathbf{M}^{-1}\mathbf{D} \end{pmatrix}$ and the input matrix $\mathbf{B}_{\text{mech}} = \begin{pmatrix} 0_{4\times3} \\ \mathbf{M}^{-1}\mathbf{Q} \end{pmatrix}$,
where the auxiliary matrix $\mathbf{Q} = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{pmatrix}$ represents the action of the external forces.

For the purpose of merely simulating the model, the output vector can be equated with the state vector, i.e. $\mathbf{C}_{\text{mech}} = \mathbf{I}_{8\times8}$. For a later observer design the measureable states have to be identified.

Model with Torsion Angle

If the rotor and generator rotation angles does not need to be calculated explicitly, it is advisable to consider only the torsion angle, i.e., the difference $\theta_s := \theta_r - \theta_g$ of the rotation angles[1], as a system state.

The structure of the state-space model remains equal to the one in (2.20):

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}_{\text{mech}}\,\mathbf{x} + \mathbf{B}_{\text{mech}}\,\mathbf{u}_{\text{mech}} \\ \mathbf{y} &= \mathbf{C}_{\text{mech}}\,\mathbf{x}\,. \end{aligned} \tag{2.21}$$

This time, however, with the following changes of state vector and system matrices:
$\mathbf{x} = \begin{pmatrix} y_T & y_B & \theta_s & \dot{y}_T & \dot{y}_B & \dot{\theta}_r & \dot{\theta}_g \end{pmatrix}^T$, $\mathbf{A}_{\text{mech}} = \begin{pmatrix} \mathbf{0}_{3\times3} & \mathbf{L}_{34} \\ -\mathbf{M}^{-1}\tilde{\mathbf{K}} & -\mathbf{M}^{-1}\mathbf{D} \end{pmatrix}$,
with $\mathbf{L}_{34} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -1 \end{pmatrix}$ and $\tilde{\mathbf{K}} = \begin{pmatrix} k_T & 0 & 0 \\ 0 & Nk_B & 0 \\ 0 & 0 & k_S \\ 0 & 0 & -k_S \end{pmatrix}$.
The input matrix is modified to $\mathbf{B}_{\text{mech}} = \begin{pmatrix} 0_{3\times3} \\ \mathbf{M}^{-1}\mathbf{Q} \end{pmatrix}$.

---

[1] In case a gearbox ratio n_g is included in the model equations, the torsion angle is defined as $\theta_s = \theta_r - \frac{1}{n_g}\theta_g$.

2.2.2 Pitch Model

In pitch-controlled wind turbines, the pitch angle of the blades is altered in the full load region to reduce the aerodynamic rotor torque and thereby to keep the turbine at the desired rotor speed. Moreover, the pitching of the blades to feather position (90°) functions as the main braking system to bring the turbine to standstill in critical situations. Two different types of pitch systems are common in wind turbines, hydraulic and electromechanical pitch systems. For hydraulic pitch systems, the dynamics can be modelled by means of a second-order delay model [Odgaard et al., 2009], which is able to display oscillatory behaviour. For electromechanical pitch systems, which are more widespread in use, a first-order delay model may be sufficient. In this work, a first-order delay model is employed:

$$\dot{\beta} = -\frac{1}{\tau}\beta + \frac{1}{\tau}\beta_d \,, \tag{2.22}$$

where β and β_d denote the physical and the demanded pitch angle, respectively, and τ denotes the delay time constant.

2.2.3 Model of the Generator and Converter Dynamics

An explicit model for the generator/converter dynamics can optionally be added to the complete system model (2.20) or (2.21). For mere simulation and control purposes, this is not necessary, since the generator/converter dynamics are fast. For the purpose of fault detection, however, an explicit generator/converter model needs to be included in order to detect actuator or sensor faults of the generator torque (see chapter 6). A simple first order delay model is sufficient and commonly used in the literature [Odgaard et al., 2009]:

$$\dot{T}_g = -\frac{1}{\tau_g}T_g + \frac{1}{\tau_g}T_{g,d} \,, \tag{2.23}$$

where $T_{g,d}$ denotes the demanded generator torque and τ_g the delay time constant.

2.2.4 Aerodynamic Model

The aerodynamic submodel consists of the expressions for the thrust force F_T acting on the rotor and the aerodynamic rotor torque T_a. These are determined by the reference force F_{st} and by the aerodynamic rotor thrust and torque coefficients C_T and C_Q that depend on the pitch angle β and the tip speed ratio $\lambda = \frac{\omega_r R}{v}$ [Gasch and Twele, 2012]:

$$F_T = F_{\text{st}}\, C_T\,(\lambda, \beta) \tag{2.24}$$

$$T_a = F_{\text{st}}\, R\, C_Q\,(\lambda, \beta)\ . \tag{2.25}$$

The reference force F_{st} is defined from the impact pressure $\frac{1}{2}\,\rho\, v^2$ and the rotor swept area $\pi\, R^2$ (with rotor radius R), where ρ denotes the air density:

$$F_{\text{st}} = \frac{1}{2}\,\rho\,\pi\, R^2\, v^2\,. \tag{2.26}$$

For the simulations, the static wind speed v is used but not the dynamic wind speed $v_e := v - \dot{y}_{B,ges} = v - (\dot{y}_T + \dot{y}_B)$ corrected by the tower and blade motion. The reason for using the static wind speed is that the aero maps used for the calculation of the rotor thrust and torque are also static, which renders a dynamic correction redundant. What is more, a dynamic correction in this case might lead to distorted results.

Thus it holds:

$$\begin{aligned} F_T &= \frac{\rho \pi R^2}{2} C_T(\lambda, \beta)\, v^2 && (2.27) \\ T_a &= \frac{\rho \pi R^3}{2} C_Q(\lambda, \beta)\, v^2 \,. && (2.28) \end{aligned}$$

The rotor thrust F_T and torque T_a are therefore nonlinearly dependent on the wind speed, the rotor speed, and the pitch angle.

2.2.5 Complete Model

Inserting expressions (2.27) and (2.28) for rotor thrust and torque into the mechanical model (2.21) and adding the models (2.22) and (2.23) for the pitch and generator/converter dynamics, a new nonlinear state-space model is obtained:

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}\,\mathbf{x} + \mathbf{B}\,\mathbf{u} + \mathbf{g}(\mathbf{x}, v)\,, \\ \mathbf{y} &= \mathbf{C}\,\mathbf{x}\,, \end{aligned} \qquad (2.29)$$

with a state vector extended by the pitch angle and the generator torque:
$\mathbf{x} = \begin{pmatrix} y_T & y_B & \theta_s & \dot{y}_T & \dot{y}_B & \omega_r & \omega_g & \beta & T_g \end{pmatrix}^T$.

Since the rotor thrust force and rotor torque, which were used as inputs for the input vector $\mathbf{u}_{\text{mech}}$ in the mechanical submodel (2.21), were explicitly inserted in (2.21), a new input vector is obtained for the complete model:

$\mathbf{u} = \begin{pmatrix} \beta_d & T_g \end{pmatrix}^T$, whose components are the demanded pitch angle and the generator torque. The wind speed is treated as a disturbance input.

The linear part of the state-space model (2.29) is determined by the new system and input matrices[2]

$$\mathbf{A} = \begin{pmatrix} \mathbf{0}_{3\times3} & \mathbf{L}_{34} & \mathbf{0}_{3\times1} & \mathbf{0}_{3\times1} \\ -\mathbf{M}^{-1}\tilde{\mathbf{K}} & -\mathbf{M}^{-1}\mathbf{D} & \mathbf{0}_{4\times1} & \begin{pmatrix} \mathbf{0}_{3\times1} \\ -\frac{1}{J_g} \end{pmatrix} \\ \mathbf{0}_{1\times3} & \mathbf{0}_{1\times4} & -\frac{1}{\tau} & 0 \\ \mathbf{0}_{1\times3} & \mathbf{0}_{1\times4} & 0 & -\frac{1}{\tau_g} \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} \mathbf{0}_{7\times1} & \mathbf{0}_{7\times1} \\ \frac{1}{\tau} & 0 \\ 0 & \frac{1}{\tau_g} \end{pmatrix}.$$

The system vector $\mathbf{g}$ in Equation (2.29) nonlinearly depends on the state vector $\mathbf{x}$ and the wind speed v:

$$\mathbf{g}(\mathbf{x}, v) = \begin{pmatrix} \mathbf{0}_{4\times1} \\ \frac{1}{N m_B} F_T(\mathbf{x}, v) \\ \frac{1}{J_r} T_a(\mathbf{x}, v) \\ \mathbf{0}_{3x1} \end{pmatrix}. \qquad (2.30)$$

Here, the rotor thrust F_T and torque T_a are given by (2.27) und (2.28), respectively. The mass and damping matrices are still defined as in (2.11), (2.12). The stiffness matrix is given as introduced in section 2.2.1.

---

[2] In the case that no explicit 1st-order dynamics of the generator torque were included in the model, the term $-1/J_g$ would appear in the input matrix $\mathbf{B}$ instead of the system matrix $\mathbf{A}$.

Centrifugal Blade Stiffening

In a real wind turbine, the centrifugal forces acting on the rotating rotor blades lead to a stiffening of the blades. As a consequence, the bending behaviour of a rotor blade depends on the rotor speed ω_r. Within the translational spring-mass system of the blade-tip displacement according to Figure 2.3, such a modification can be implemented by introducing a rotor speed dependent term in the translational blade stiffness parameter:

$$k_{B,\text{eff}} = k_B + k_B^{\text{centr}}(\omega_r) = k_B + \alpha\, m_B\, r_B\, \omega_r^2\,, \tag{2.31}$$

where r_B denotes the distance from the blade root to the blade centre of mass and α is a constant that needs to be adjusted to the simulated turbine. Including the centrifugal stiffening correction, the nonlinear system vector $\mathbf{g}(\mathbf{x}, v)$ of the state-space model (2.29) is modified to

$$\mathbf{g}(\mathbf{x}, v) = \begin{pmatrix} \mathbf{0}_{3\times 1} \\ \frac{N}{m_T}\, k_B^{\text{centr}}(\omega_r)\; y_B \\ \frac{1}{N m_B}\, F_T(\mathbf{x}, v) + \frac{m_T + N\, m_B}{m_T\, m_B}\, k_B^{\text{centr}}(\omega_r)\; y_B \\ \frac{1}{J_r}\, T_a(\mathbf{x}, v) \\ \mathbf{0}_{3x1} \end{pmatrix}. \tag{2.32}$$

The inclusion of a centrifugal term is inspired from the FAST simulation software, where the correction is done for every blade section. For the substitute translational blade bending model according to Figure 2.3, an approach according to (2.31) is appealing, for example if overspeed scenarios shall be taken into account. In first simulation tests with a freely rotating turbine at high tip-speed ratios, it could be shown that the results for the blade-tip displacement in the reduced-order model correspond better to the FAST simulation results when including the correction (2.31). However, for the usual operating regimes of a wind turbine, the corrections induced by the centrifugal blade stiffening are only minor. Furthermore, when including (2.31) into a TS model for controller or observer design, an additional nonlinearity is introduced, which doubles the number of TS submodels (see chapter 3). For these reasons, a centrifugal correction is not considered in the rest of this work.

2.3 Model Parameters

For all simulations in this work, the NREL 5 MW turbine is taken as a reference model, for which the system parameters and the corresponding input files for FAST simulation are available [Jonkman et al., 2009]. When adapting the NREL 5 MW turbine model for the reduced-order model (2.29) with four mechanical degrees of freedom, some parameters can be directly adopted from [Jonkman et al., 2009] and from example FAST input and log files, whereas others have to be recalculated or adapted, in particular the stiffness parameters for the tower and blade degrees of freedom.

All parameters for the reduced-order wind turbine model directly taken or adapted from the NREL 5 MW reference turbine are listed in appendix A.1. The next sections are concerned with those parameters that need to be specifically adapted for the reduced-order model.

2.3.1 Aerodynamic Damping Parameters

Due to weak structural damping in the tower, the damping of the nacelle-tower vibrations is mainly determined by the aerodynamic rotor damping and the damping due to the ground. For a reduced-order model like (2.29), the aerodynamic rotor damping in fore-aft direction can be taken as an estimate for the damping parameter d_T of the tower-top motion [Gasch and Twele, 2012]. The aerodynamic damping in fore-aft direction can be approximated as

$$d_{11}\left(\lambda, \beta\right) = 0.5\,\rho\,\pi\,R^2\,v\,d_{11}^{*}\left(\lambda\right)\ , \tag{2.33}$$

where the dimensionless parameter $d_{11}^{*}\left(\lambda, \beta\right)$ depends on the tip speed ratio and on the pitch angle and shows a similar behaviour for different turbine sizes [Kaiser and Gasch, 1996; Kaiser, 2000]. Estimating $d_{11}\left(\lambda, \beta\right)$ accordingly for different stationary points of the 5 MW reference turbine for the whole operating range of the turbine yields values between $3 \cdot 10^4\,\frac{\mathrm{Ns}}{\mathrm{m}}$ and $10 \cdot 10^4\,\frac{\mathrm{Ns}}{\mathrm{m}}$. Here, the tower damping parameter is set to a constant value of $d_T = 7 \cdot 10^4\,\frac{\mathrm{Ns}}{\mathrm{m}}$. The blade damping parameter is set to $d_B = 2 \cdot 10^4\,\frac{\mathrm{Ns}}{\mathrm{m}}$.

2.3.2 Structural Parameters

In FAST, the dynamics of fore-aft tower and flap-wise rotor blade bending are implemented by means of beam bending models, whereas in the reduced-order model (2.29), the tower and blade bending dynamics are reduced to simple spring-mass-damper systems for the tower top and blade tip deflections (see Figures 2.2 and 2.3). Therefore, the bending stiffness parameters given in the documentation for the NREL 5 MW turbine [Jonkman et al., 2009] must be converted to equivalent translational stiffness parameters.

In a first phenomenological approach, the translational stiffness parameters were obtained by manual adjustment according to results from simulations of load-free start-up procedures. The stiffness parameters were adjusted, such that the average behaviour of the time-series of the tower top and blade tip displacements y_T and y_B were in good agreement with the corresponding FAST simulation results for a constant wind speed of $v\ =\ 8\,\frac{m}{s}$ at a constant pitch angle of $\beta\ =\ 0°$.
This approach yielded the following parameters: $k_T = 1962000\,\frac{\mathrm{N}}{\mathrm{m}}$, $k_B = 40000\,\frac{\mathrm{N}}{\mathrm{m}}$.

The approach using simulations of load-free start-up procedures enables an estimate of the effective stiffness constants in a scenario of static deflection of the tower and blade, since an equilibrium of aerodynamic forces is reached at the idling tip speed ratio.

Apart from the stiffness parameters, effective mass parameters m_T and m_B for the tower-nacelle and blade motion in the translational spring-mass model have to be calculated, as only parts of the complete tower and blade masses contribute to the tower-top and blade-tip motions. For the tower-nacelle motion, the effective mass m_T in Equation (2.6) is estimated as
$m_T\ =\ m_{\text{Rotor}} + m_{\text{Nacelle}} + 0.25\,m_{\text{Tower}}$, which has proven a reasonable assumption [Gasch and Twele, 2012]. Similarly, the effective blade mass for the blade tip motion is estimated as $m_B\ =\ 0.25\,m_{\text{Blade}}$.

Direct Stiffness Method for Derivation of Effective Tower Stiffness

To obtain a more theoretically grounded result for the effective translational stiffness parameters, a direct stiffness method, common in structural mechanics calculations [**?**], was applied in [Georg, Müller, and Schulte, 2014].
The direct stiffness method allows to calculate eigenfrequencies and eigenmodes of structures consisting of several segments of defined length, mass and bending stiffness. For each segment, the characteristic forces and displacements can be calculated from the previous segment by means of a transfer matrix depending on the frequency of the structure [**?**]. Applying the total transfer matrix as the product of the individual transfer matrices, as well as the boundary conditions for the rigid and the free ends of the beam, yields a homogeneous system of equations for the displacements at the top of the total structure, which is fulfilled

for the eigenfrequencies of the structure. In order to calculate an equivalent bending stiffness for the complete structure, it is sufficient to find the first eigenfrequency ω_1. For the tower, it was calculated as $\omega_1 \approx 2.14\,\frac{\mathrm{rad}}{\mathrm{s}}$ and was validated against the NREL-Software BModes [Bir, 2012] ($\omega_{1,\mathrm{BModes}} \approx 2.08\,\frac{\mathrm{rad}}{\mathrm{s}}$). The equivalent bending stiffness B_{total} of the complete structure can then be calculated using beam bending theory for a virtual single beam with one free end:

$$\omega_1 = \kappa_1^2 \sqrt{\frac{B_{total}}{\mu_{total}}} \quad \Rightarrow \quad B_{total} = \frac{\omega_1^2\,\mu_{total}}{\kappa_1^4} \approx 4.44 \cdot 10^{11}\,\mathrm{Nm}^2\,, \tag{2.34}$$

where $\kappa_1 = 1.423 \cdot 10^{-2}$ is a factor that can be found in standard mechanics textbooks and μ_{total} is the total mass per length. Finally, the equivalent bending stiffness can be transferred into a translational spring stiffness constant using simple relations for the deflection w of the beam (with total length l) and the mass in the translational spring-mass system, where the applied force F corresponds to the rotor thrust force F_T:

$$w = \frac{F\,l^3}{3\,B_{total}} \quad \text{(beam)}, \quad F = k\,w \quad \text{(spring-mass system)} \qquad \Rightarrow \quad k = \frac{3\,B_{total}}{l^3}\,. \tag{2.35}$$

For the tower, the described approach yielded an equivalent translational stiffness constant of $k_T = 1981900\,\frac{\mathrm{N}}{\mathrm{m}}$, which is in good agreement with the constant obtained by means of the phenomenological approach.

However, for the equivalent blade stiffness constant, the direct stiffness calculation yielded no realistic result, such that the value obtained from the simple approach is taken for all simulation studies.

2.3.3 Aero Maps for Rotor Thrust and Torque

The equations (2.27) and (2.27) for the aerodynamics contain the rotor thrust and torque coefficients C_T and C_Q, which nonlinearly depend on the tip-speed ratio and on the pitch angle. The coefficients are determined by nonlinear aero maps, which must be known for the whole range of pitch angles and tip speed ratios up to the respective idling tip speed ratio. The description with aero maps is a static approximation of more detailed aerodynamic calculations implemented in wind turbine load simulation tools like FAST, which utilises the module AeroDyn [Laino and Hansen, 2002] for this purpose.

A common approach in wind turbine load simulation is the blade element momentum (BEM) method, where the aerodynamic lift and drag forces at each blade section are calculated and integrated afterwards to obtain the rotor thrust and torque (see for example [Hansen, Martin O. L., 2008; Burton et al., 2011; Gasch and Twele, 2012]).

Two methods were employed in this work to obtain the aero maps. First, a BEM code was implemented according to the algorithm described in [Gasch and Twele, 2012]. Using the geometrical and aerodynamic blade section parameters for the NREL 5 MW turbine, the aero maps can thus be calculated. This approach yields reasonable results.

However, as the internal calculations in AeroDyn are more precise[3], a second method was devised, whereby the aero maps were extracted from several FAST/AeroDyn simulation runs. Simulating load-free start-up procedures of the wind turbine at a given constant wind speed and zero applied generator torque ($T_g = 0$ Nm), the wind turbine accelerates up to the idling tip speed and will then have run through the whole domain of tip speed ratios from $\lambda = 0$ up to the idling tip speed ratio. This procedure requires relatively few simulation runs to determine the complete aero maps for C_Q and C_T. In appendix A.2, the procedure is described in more detail.

---

[3] A more complex aerodynamics calculation than BEM is implemented in AeroDyn

Analytical Approximation of the Aero Maps

For simulation purposes only, tabulated versions of the aero maps C_Q and C_T would suffice. For control design, however, the derivatives of the rotor torque (and thrust) are needed, which necessitates a description of the aero maps in form of analytical functions.

The C_Q map can be approximated using a modified and expanded form of the function for the power coefficient given in [Heier, 2006] (which is itself a modified version of the expressions given in [Wasynczuk et al., 1981] and [Amlang et al., 1992]):

$$\begin{aligned}\tilde{C}_Q(\lambda,\beta) = {} & c_1\left(1 + c_2\,(\beta + c_3)^{\frac{1}{2}}\right) \\ & + \frac{c_4}{\lambda}\,\left(c_5\,\lambda_i(\lambda,\beta) - c_6\beta - c_7\beta^{c_8} - c_9\right)\exp\left(-c_{10}\,\lambda_i(\lambda,\beta)\right) \quad (\lambda > 0)\,,\end{aligned} \tag{2.36}$$

with $\lambda_i(\lambda,\beta) = \dfrac{1}{\lambda + 0.08\,\beta} - \dfrac{0.035}{c_{11} + c_{12}\beta^3} \quad (\lambda > 0)\ .$

The function $\tilde{C}_Q$ may assume negative values. If the analytical form of C_Q shall be used in simulations, unphysical negative coefficients must be excluded. To this end, function (2.36) is multiplied by an additional term, which yields the final approximation for C_Q:

$$C_Q(\lambda,\beta) = \tilde{C}_Q(\lambda,\beta)\,\frac{1 + \operatorname{sgn}\tilde{C}_Q(\lambda,\beta)}{2}\,. \tag{2.37}$$

The coefficients were determined using the MATLAB® Curve Fitting Toolbox by fitting Equation (2.37) to the values of C_Q obtained from FAST/AeroDyn simulations of the 5 MW turbine. The following values were obtained:[4]

$$\begin{array}{llll} c_1 = 0.005 & c_2 = 1.53 & c_3 = 0.5 & c_4 = 0.18 \\ c_5 = 121 & c_6 = 27.9 & c_7 = 198 & c_8 = 2.36 \\ c_9 = 5.74 & c_{10} = 11.35 & c_{11} = 16.1 & c_{12} = 201\,. \end{array}$$

The map for C_T was approximated using the following polynomial-exponential function:

$$\tilde{C}_T(\lambda,\beta) = a_1 + a_2\,(\lambda - a_3\beta)\,e^{-a_4\beta} + a_5\,\lambda^2\,e^{-a_6\beta} + a_7\,\lambda^3\,e^{-a_8\beta}\,. \tag{2.38}$$

Again, to exclude unphysical negative coefficients, function (2.38) is multiplied by an additional term, which yields the final approximation for C_T:

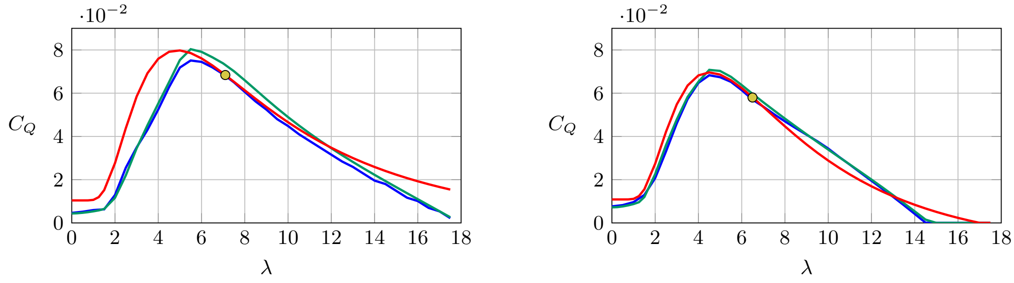
$$C_T(\lambda,\beta) = \tilde{C}_T(\lambda,\beta)\,\frac{1 + \operatorname{sgn}\tilde{C}_T(\lambda,\beta)}{2}\,. \tag{2.39}$$

The coefficients were determined as

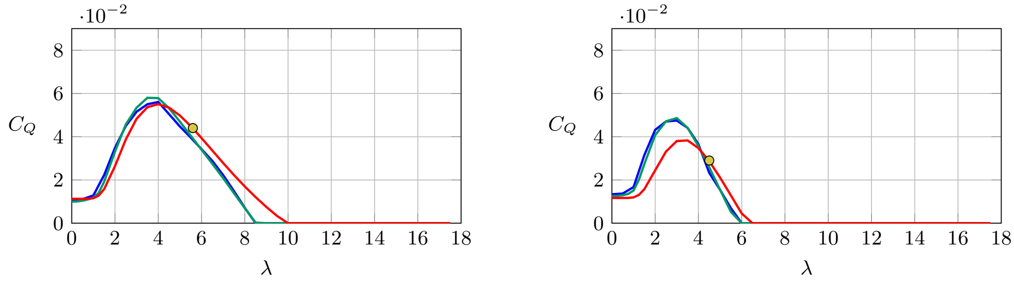
$$\begin{array}{llll} a_1 = 0.006 & a_2 = 0.095 & a_3 = -4.15 & a_4 = 2.75 \\ a_5 = 0.001 & a_6 = 7.8 & a_7 = -0.00016 & a_8 = -8.88\,. \end{array}$$

---

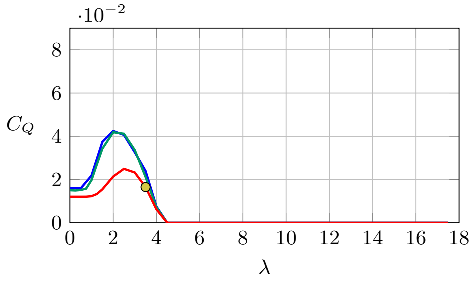
[4] These coefficients correspond to a calculation using the SI unit [rad] for the pitch angle.



(a) $\beta = 0°$ (b) $\beta = 5°$



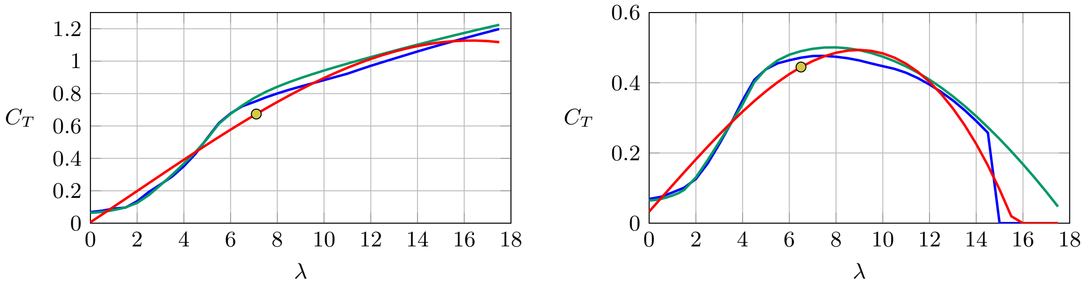
(c) $\beta = 10°$ (d) $\beta = 15°$



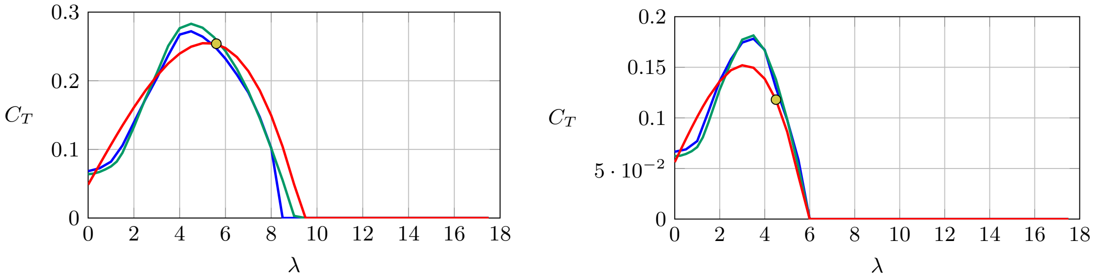
(e) $\beta = 20°$

Figure 2.5: Aerodynamic rotor torque coefficient C_Q for different fixed pitch angles. Blue lines: C_Q extracted from FAST/AeroDyn simulations; Green lines: C_Q obtained from BEM calculation; Red lines: C_Q obtained from analytical approximation (2.37). The yellow point indicates the respective stationary point in the full-load region for each fixed pitch angle.

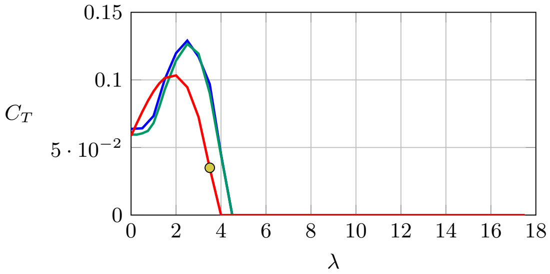
In Figures 2.5 and 2.6, the C_Q and C_T coefficients are depicted for five different fixed pitch angles between 0 and 20°. In each figure, the curves extracted from FAST/AeroDyn simulations are compared both to those obtained from BEM calculations and to those from the analytical functions (2.37) and (2.39). One can see that the overall agreement between the analytical functions and the curves obtained from FAST and from BEM calculations is quite good, considering that only one two-dimensional analytical function is used for either of the rotor torque and thrust coefficient. In particular, for the rotor torque coefficient C_Q, there should be a very good agreement between the derivatives of C_Q with respect to the tip speed ratio λ for each pitch angle at the respective stationary point in the full-load region. This is because for the controller design in the full-load region, a local controller is designed for each stationary point and the derivatives of C_Q with respect to λ and β enter the linearised matrix which influences the resulting controller gain coefficients (see chapter 4). One can see from Figure 2.5 that there is a very good agreement between the slopes of C_Q at the respective stationary points especially for small pitch angles. With increasing pitch angle, the discrepancy between the slopes of the curves obtained from the analytical function and the tabulated curves slightly increases, but is still acceptable. The stationary points depicted in Figures 2.5 and 2.6 correspond to the aero maps represented by the analytical functions.



(a) $\beta = 0°$ (b) $\beta = 5°$



(c) $\beta = 10°$ (d) $\beta = 15°$



(e) $\beta = 20°$

Figure 2.6: Aerodynamic rotor thrust coefficient C_T for different fixed pitch angles. Blue lines: C_T extracted from FAST/AeroDyn simulations; Green lines: C_T obtained from BEM calculation; Red lines: C_T obtained from analytical approximation (2.39). The yellow point indicates the respective stationary point in the full-load region for each fixed pitch angle.

The derivation of the aero maps for the 5 MW turbine described in this section and the approximated functions (2.36) and (2.38) were first published in [Georg et al., 2012]:

Sören Georg, Horst Schulte, Harald Aschemann. Control-Oriented Modelling of Wind Turbines Using a Takagi-Sugeno Model Structure. *IEEE International Conference on Fuzzy Systems*, pp. 1737 – 1743, Brisbane, Australia, 2012. © 2012 IEEE.

2.4 Remark on Measurement Noise

In the wind turbine FDI benchmark models [Odgaard et al., 2009, 2013; Odgaard and Johnson, 2013], white noise is added to all measurement signals. This is based on the assumption that adding white noise to the sensor signals leads to more realistic scenarios. This, however, is not the case, as a realistic simulation would demand a precise knowledge of each sensor and its measuring quality. From the author's working experience with a wind turbine manufacturer, all important measurements utilised in the present work (rotor and generator speed, pitch angle, generator torque), are virtually noise free or only superimposed with very weak noise. It was thus decided to omit measurement noise in this work.

What is present in wind turbines however, is one characteristic frequency imposed on the generator speed signal due to the periodic rotor excitation of the drive train. When using the generator speed as a controlled variable, a notch filter thus needs to be applied to smooth the generator speed signal. Such a notch filter is routinely applied in industrial wind turbine control [Bossanyi, 2000] and was also applied for the FAST simulations in Section 7.2 in this work, where the generator speed was used as controlled variable. For the rotor speed measurement, a simplifying assumption was made in this work that it is a continuous signal, which is justified for the purpose of validating the basic fault reconstruction facilities. In most current wind turbines, the rotor speed signal is discretised, due to the limited number of metal pieces on the main shaft that are scanned by a magnetic sensor, although better sensing technologies, which could yield nearly continuous signals, are already available, like optical scanning of densely spaced barcodes.

3

Takagi-Sugeno Model Structure

Due to the aerodynamics, the state-space model (2.29) of the wind turbine is nonlinear. When using this model as a basis for (nonlinear) controller and observer design, a way must be found to adequately deal with the nonlinearities.

Throughout this work, the nonlinear state-space models, as well as control laws and observer models, are formulated in terms of the Takagi-Sugeno (TS) model structure. TS models provide a useful and uniform framework for nonlinear controller and observer design. Originally introduced in the context of fuzzy systems [Takagi and Sugeno, 1985], TS models are weighted combinations of linear submodels and can either be derived from input-output data using system identification techniques [Takagi and Sugeno, 1985; Sugeno and Kang, 1988] or from mathematical models of nonlinear systems. Methods based on linear matrix inequalities (LMIs) allow for implicit stable controller and observer design for TS models [Wang et al., 1996; Tanaka and Wang, 1997].

The general TS structure of a state-space model is of the form

$$\begin{aligned} \dot{\mathbf{x}} &= \sum_{i=1}^{N_r} h_i(\mathbf{z})\ \left(\mathbf{A}_i\,\mathbf{x} + \mathbf{B}_i\,\mathbf{u}\right)\ , \\ \mathbf{y} &= \sum_{i=1}^{N_r} h_i(\mathbf{z})\,\mathbf{C}_i\,\mathbf{x}\,, \end{aligned} \tag{3.1}$$

with $N_r = 2^{N_l}$, where N_l denotes the number of dissimilar nonlinearities in the model, i.e. equal nonlinear terms that appear in several matrix positions are only counted as $N_l = 1$. Matrices $\mathbf{A}_i$, $\mathbf{B}_i$ and $\mathbf{C}_i$ have constant coefficients. The vector $\mathbf{z}$ of premise variables may comprise states x_k, inputs u_k, and external variables $\boldsymbol{\chi}$: $\mathbf{z} = \mathbf{z}\left(\mathbf{x}, \mathbf{u}, \boldsymbol{\chi}\right)$. The functions h_i are the membership functions and fulfill the two conditions

$$\sum_{i=1}^{N_r} h_i(\mathbf{z}) \quad = \quad 1 \tag{3.2}$$

$$0 \quad \leq \quad h_i(\mathbf{z}) \leq 1 \ (i \in \{1, \cdots, N_r\})\ . \tag{3.3}$$

where Equation (3.2) is the convex sum condition. In many cases, there is only one common output matrix $\mathbf{C}$, such that the usual output equation $\mathbf{y} = \mathbf{C}\,\mathbf{x}$ is obtained.

In order to obtain the TS structure of a nonlinear model, and provided that a mathematical nonlinear model is given, there are essentially two ways to derive the linear TS submodels. The first method is local Taylor linearisation of the nonlinear model around N_r stationary points and subsequent fuzzy-blending of the linear submodels to a weighted sum according to (3.1), where standard fuzzy membership functions can be used, like triangular, trapezoidal, sigmoid, or Gaussian functions [Schulte, 2006; Lendek et al., 2010]. The linearisation approach is used in chapter 4 to obtain an approximated TS model for the controller design in the full-load region.

The second method is the so-called sector nonlinearity approach [Tanaka and Sano, 1994a] (and references therein), [Ohtake et al., 2001; Tanaka and Wang, 2001], which yields an exact representation of the nonlinear model. This approach is used for all observer designs in this work. In the following, the procedure to derive a TS model using the sector nonlinearity approach is illustrated at the example of the reduced-order wind turbine model (2.29). This derivation was first published in [Georg et al., 2012] and forms the basis for all variants of observer state-space models in TS structure in this work.

First, the nonlinearity of the vector $\mathbf{g}(\mathbf{x}, v)$ from equation (2.30) in the state-space model (2.29) has to be shifted to the membership functions h_i, such that the subsystems $(\mathbf{A}_i\,\mathbf{x} + \mathbf{B}_i\,\mathbf{u})$ are linear state-space models.

To this end, $\mathbf{g}(\mathbf{x}, v)$ is first written as a product of a matrix and the state vector $\mathbf{x}$:

$$\mathbf{g}(\mathbf{x}, v) = \begin{pmatrix} \mathbf{0}_{4x1} \\ \frac{1}{Nm_B}\,F_T\,(\mathbf{x}, v) \\ \frac{1}{J_r}\,T_a\,(\mathbf{x}, v) \\ \mathbf{0}_{3x1} \end{pmatrix} = \underbrace{\begin{pmatrix} \multicolumn{3}{c}{\mathbf{0}_{4x9}} \\ \mathbf{0}_{1x5} & \frac{1}{Nm_B}\,F_T\,(\mathbf{x}, v)\,\frac{1}{x_6} & \mathbf{0}_{1x3} \\ \mathbf{0}_{1x5} & \frac{1}{J_r}\,T_a\,(\mathbf{x}, v)\,\frac{1}{x_6} & \mathbf{0}_{1x3} \\ \multicolumn{3}{c}{\mathbf{0}_{3x9}} \end{pmatrix}}_{=:\,\mathbf{A}_{\text{NL}}(\mathbf{z}) = \mathbf{A}_{\text{NL}}(\mathbf{x}, v)} \mathbf{x}\,, \tag{3.4}$$

provided that the rotor speed $\omega_r = x_6$ is greater than zero. This assumption is clearly valid in all situations when the turbine is running.[1]

Then, the state equation of the state-space model (2.29) can be written as follows:

$$\dot{\mathbf{x}} = (\mathbf{A}_{\text{lin}} + \mathbf{A}_{\text{NL}}\,(\mathbf{z}))\,\mathbf{x} + \mathbf{B}\,\mathbf{u}\,, \tag{3.5}$$

with

$$\mathbf{A}_{\text{lin}} = \begin{pmatrix} \multicolumn{3}{c}{\mathbf{0}_{3x3}} & \multicolumn{4}{c}{\mathbf{L}_{34}} & \mathbf{0}_{3x1} & \mathbf{0}_{3x1} \\ -\frac{k_T}{m_T} & \frac{Nk_B}{m_T} & 0 & -\frac{d_T}{m_T} & \frac{Nd_B}{m_T} & 0 & 0 & 0 & 0 \\ \frac{k_T}{m_T} & -\frac{k_B}{m_B} - \frac{Nk_B}{m_T} & 0 & \frac{d_T}{m_T} & -\frac{d_B}{m_B} - \frac{Nd_B}{m_T} & 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{k_S}{J_r} & 0 & 0 & -\frac{d_S}{J_r} & \frac{d_S}{J_r} & 0 & 0 \\ 0 & 0 & \frac{k_S}{J_g} & 0 & 0 & \frac{d_S}{J_g} & -\frac{d_S}{J_g} & 0 & -\frac{1}{J_g} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\frac{1}{\tau} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\frac{1}{\tau_g} \end{pmatrix}, \tag{3.6}$$

---

[1] For a practical implementation in a wind turbine control software, in order to prevent a division by zero, a tiny offset may be added to x_6 in the denominator to handle the case when the rotor speed is identical to zero, i.e., when the rotor is locked.

$$\mathbf{A}_{\mathrm{NL}}(\mathbf{z}) \;=\; \begin{pmatrix} & \mathbf{0}_{4x9} & \\ \mathbf{0}_{1x5} & f_1(\mathbf{z}) & \mathbf{0}_{1x3} \\ \mathbf{0}_{1x5} & f_2(\mathbf{z}) & \mathbf{0}_{1x3} \\ & \mathbf{0}_{3x9} & \end{pmatrix} . \tag{3.7}$$

The scalar nonlinear functions f_1 and f_2 are given by

$$f_1(\mathbf{z}) \;=\; \frac{1}{Nm_B}\, F_T\left(\omega_r, \beta, v\right) \frac{1}{x_6} \qquad (x_6 > 0) \;, \tag{3.8}$$

$$f_2(\mathbf{z}) \;=\; \frac{1}{J_r}\, T_a\left(\omega_r, \beta, v\right) \frac{1}{x_6} \qquad (x_6 > 0) \;, \tag{3.9}$$

which implies that the vector of premise variables in this case is given by $\mathbf{z} = (\omega_r, \beta, v)$. For the transition to the TS model structure (3.1), functions f_1 and f_2 are written in a different form. Let $\underline{f}_1$ and $\overline{f}_1$, respectively $\underline{f}_2$ and $\overline{f}_2$ denote the minimum and maximum values of the functions f_1 and f_2. Then the following identities hold:

$$f_1(\mathbf{z}) \;=\; w_{11}(\mathbf{z})\,\overline{f}_1 \;+ w_{12}(\mathbf{z})\,\underline{f}_1 \,, \tag{3.10}$$

$$f_2(\mathbf{z}) \;=\; w_{21}(\mathbf{z})\,\overline{f}_2 \;+ w_{22}(\mathbf{z})\,\underline{f}_2 \,, \tag{3.11}$$

where the weighting functions w_{jk} are given by

$$w_{j1}(\mathbf{z}) \;:=\; \frac{f_j(\mathbf{z}) - \underline{f}_j}{\overline{f}_j - \underline{f}_j} \quad \text{and} \quad w_{j2}(\mathbf{z}) \;:=\; \frac{\overline{f}_j - f_j(\mathbf{z})}{\overline{f}_j - \underline{f}_j} \qquad (j \in \{1, 2\}) \;. \tag{3.12}$$

Note that for simulation purposes only, $\underline{f}_j$ and $\overline{f}_j$ could be any real values, as long as $\underline{f}_j \neq \overline{f}_j$. However, when the TS model is used for controller or observer design, the choice of $\underline{f}$ and $\overline{f}$ matters, because these numbers describe the vertices of the linear submodels and therefore influence the control design.

From (3.10), (3.11) and (3.12) it follows that the weighting functions fulfill the convex sum condition

$$w_{j1} + w_{j2} \;=\; 1 \qquad (j \in \{1, 2\}) \;. \tag{3.13}$$

From equation (3.13), and by defining the sum of membership functions as the product of the convex sums of the weighting functions w_{jk}:

$$\sum_{i=1}^{N_r} h_i(\mathbf{z}) \;:=\; \prod_{j=1}^{N_l=2} \left(w_{j1} + w_{j2}\right) \;, \tag{3.14}$$

it directly follows that $\sum\limits_{i=1}^{N_r} h_i(\mathbf{z}) = 1$, i.e. the convex sum condition (3.2) for TS membership functions holds. Furthermore, the structure defined by (3.14) corresponds to the structure of the nonlinear functions in (3.10) and (3.11), such that f_1 and f_2, by multiplying them by an appropriate choice of a convex sum, can be written as combinations of their respective minimum and maximum values and the membership functions:

$$f_1 = \left(w_{11}\,\overline{f}_1 + w_{12}\,\underline{f}_1\right)\underbrace{(w_{21} + w_{22})}_{=1} = h_1\,\overline{f}_1 + h_2\,\overline{f}_1 + h_3\,\underline{f}_1 + h_4\,\underline{f}_1\,, \tag{3.15}$$

$$f_2 = \left(w_{21}\,\overline{f}_2 + w_{22}\,\underline{f}_2\right)\underbrace{(w_{11} + w_{12})}_{=1} = h_1\,\overline{f}_2 + h_2\,\underline{f}_2 + h_3\,\overline{f}_2 + h_4\,\underline{f}_2\,. \tag{3.16}$$

By using this formulation, the nonlinearities have been shifted to the membership functions, which in the considered case are given by

$$h_1 := w_{11}\,w_{21}\,, \quad h_2 := w_{11}\,w_{22}\,, \quad h_3 := w_{12}\,w_{21}\,, \quad h_4 := w_{12}\,w_{22}\,.$$

All other matrix entries a_{ij} of $\mathbf{A}_{\text{NL}}(\mathbf{z})$, which happen to be zero in this case, are multiplied by $\sum\limits_{i=1}^{N_r} h_i = 1$, such that $\mathbf{A}_{\text{NL}}(\mathbf{z})$ can be written as a weighted sum of four matrices with constant coefficients:

$$\mathbf{A}_{\text{NL}}(\mathbf{z}) = \sum_{i=1}^{N_r} h_i(\mathbf{z})\,\mathbf{A}_{\text{NL},i}\,, \tag{3.17}$$

with

$$\mathbf{A}_{\text{NL},1} = \begin{pmatrix} a_{11} & & & \cdots & & & a_{19} \\ & & & \vdots & & & \\ a_{51} & \cdots & a_{55} & \overline{f}_1 & \cdots & & a_{59} \\ a_{61} & \cdots & a_{65} & \overline{f}_2 & \cdots & & a_{69} \\ & & & \vdots & & & \\ a_{91} & & & \cdots & & & a_{99} \end{pmatrix}, \qquad \mathbf{A}_{\text{NL},2} = \begin{pmatrix} a_{11} & & & \cdots & & & a_{19} \\ & & & \vdots & & & \\ a_{51} & \cdots & a_{55} & \overline{f}_1 & \cdots & & a_{59} \\ a_{61} & \cdots & a_{65} & \underline{f}_2 & \cdots & & a_{69} \\ & & & \vdots & & & \\ a_{91} & & & \cdots & & & a_{99} \end{pmatrix},$$

$$\mathbf{A}_{\text{NL},3} = \begin{pmatrix} a_{11} & & & \cdots & & & a_{19} \\ & & & \vdots & & & \\ a_{51} & \cdots & a_{55} & \underline{f}_1 & \cdots & & a_{59} \\ a_{61} & \cdots & a_{65} & \overline{f}_2 & \cdots & & a_{69} \\ & & & \vdots & & & \\ a_{91} & & & \cdots & & & a_{99} \end{pmatrix}, \qquad \mathbf{A}_{\text{NL},4} = \begin{pmatrix} a_{11} & & & \cdots & & & a_{19} \\ & & & \vdots & & & \\ a_{51} & \cdots & a_{55} & \underline{f}_1 & \cdots & & a_{59} \\ a_{61} & \cdots & a_{65} & \underline{f}_2 & \cdots & & a_{69} \\ & & & \vdots & & & \\ a_{91} & & & \cdots & & & a_{99} \end{pmatrix}.$$

The state-space model (2.29) can thus be transformed into the TS form

$$\begin{aligned} \dot{\mathbf{x}} &= \sum_{i=1}^{N_r} h_i(\mathbf{z})\,\mathbf{A}_i\,\mathbf{x} + \mathbf{B}\,\mathbf{u} = \sum_{i=1}^{N_r} h_i(\mathbf{z})\,\left(\mathbf{A}_{\text{lin}} + \mathbf{A}_{\text{NL,i}}\right)\,\mathbf{x} + \mathbf{B}\,\mathbf{u}\,, \\ \mathbf{y} &= \mathbf{C}\,\mathbf{x}\,, \end{aligned} \tag{3.18}$$

with $\mathbf{z} = (\omega_r, \beta, v)$. In the above example, the nonlinearities were completely shifted into the matrix $\mathbf{A}$, such that the term $\mathbf{B}\,\mathbf{u}$ remains unchanged. Other nonlinear state-space models may call for a different TS structure, where the nonlinear terms may as well appear in the input matrix $\mathbf{B}$.

The minimum and maximum values of the functions f_1 and f_2 were estimated as

$$\begin{aligned} \underline{f}_1 &= 3.4 \cdot 10^{-4}\,\frac{\mathrm{m}}{\mathrm{s}}\,, & \overline{f}_1 &= 2.48 \cdot 10^{5}\,\frac{\mathrm{m}}{\mathrm{s}}\,, \\ \underline{f}_2 &= 7.4 \cdot 10^{-6}\,\mathrm{s}^{-1}\,, & \overline{f}_2 &= 335.6\,\mathrm{s}^{-1}\,. \end{aligned}$$

where the maximum values of C_Q and C_T were estimated according to the coefficients extracted from FAST/AeroDyn simulations (see chapter 2). The minimum values of both C_Q and C_T were set to 0.001 to avoid zero entries for either $\underline{f}_1$ or $\underline{f}_2$ in the matrices $\mathbf{A}_{\mathrm{NL},i}$, as this may affect the observability of the TS submodels. For the same reason, the minimum wind speed was set to $v_{\text{min}} = 1\,\frac{\mathrm{m}}{\mathrm{s}}$. The maximum wind speed was estimated as $v_{\text{min}} = 60\,\frac{\mathrm{m}}{\mathrm{s}}$, which corresponds to the 50-year return gust speed for wind turbine class II [GL Renewables Certification, 2010; Burton et al., 2011] and is well above the typical shut-down extreme wind speed of $35\,\frac{\mathrm{m}}{\mathrm{s}}$. The minimum rotor speed was set to $\omega_{r,\text{min}} = 0.01\,\frac{\mathrm{rad}}{\mathrm{s}}$, and the maximum rotor speed was estimated as $\omega_{r,\text{max}} = 16 \cdot \frac{\pi}{30}\,\frac{\mathrm{rad}}{\mathrm{s}}$, which is well above the rated speed of $\omega_{r,\text{SP}} = 1.267\,\frac{\mathrm{rad}}{\mathrm{s}}$.

4

Wind Turbine Control

The rotor speed control for a wind turbine is usually designed separately for the different load regions (partial load, transition region, full load). In the partial load region below the rated wind speed, the pitch angle is kept at the fine pitch angle (usually $0°$), where maximum energy extraction is possible, while the generator torque is adjusted such that the turbine is operating around the optimal tip speed ratio and thereby around the maximum power coefficient. The standard approach is a deterministic squared-law for the generator torque in dependence on the rotor speed: $T_g \sim \omega_r^2$.
In the transition region, the generator torque is raised to the maximum value using a fast controller (see section 4.2). In the full load region above the rated wind speed, only the pitch angle is adjusted to control the rotor speed around the desired rated rotor speed $\omega_{r,\text{SP}}$. Increasing the pitch angle reduces the aerodynamic lift force at the rotor blades and thereby leads to a reduced rotor torque.

The plant dynamics in the partial load region, transition region, and full load region are thus completely different. For these reasons, the two controllers using pitch angle respectively generator torque as control inputs are considered and designed separately. In the wind turbine literature, these two separate control loops are typically termed **pitch control** and **torque control**, respectively [Burton et al., 2011]. For the sake of simplicity and comparability, these terms are therefore adopted in this work, too. For an overview of wind turbine control see also the classic treatment in [Bossanyi, 2000], which still defines an industrial standard.

In Figure 4.1, the different control zones of a wind turbine are illustrated by plotting the generator torque versus the rotor speed. Typically, there is a cut-in rotor speed $\omega_{r,1}$, below which the turbine rotates freely in load-free operation. Between $\omega_{r,1}$ and $\omega_{r,2}$, the generator torque is increased linearly until the optimal torque control law $T_g \sim \omega_r^2$ is reached [Jonkman et al., 2009]. This optimal law is then followed up to a defined rotor speed setpoint $\omega_{r,\text{SP,trq}}$, which defines the onset of the transition region, where the pitch angle is still kept at the fine pitch angle. The generator torque controller in the transition region should be reasonably fast, such that the maximum torque is reached when the pitch control is activated at the second setpoint $\omega_{r,\text{SP}}$. This second setpoint coincides with the rated rotor speed, at which the rated power of the wind turbine is reached. The values for the two setpoints $\omega_{r,\text{SP,trq}}$ and $\omega_{r,\text{SP}}$ are chosen differently in order to decouple the two control loops for torque and pitch control. There also exist other strategies to realise this decoupling (see [Bossanyi, 2000; Burton et al., 2011]).

In this chapter, controller designs for both the full load and the transition region are conducted. For the linear transition region, a standard state-space controller with integral part is designed using the LQ optimal criterion (see section 4.2). For the full load region, a controller in TS structure, a so-called PDC (Parallel Distributed Compensation), is designed based on Taylor-linearised submodels which are blended with standard Fuzzy membership functions (Section 4.1.1). By using this approach, two aims are achieved. First, the local subcontroller designs contain the derivatives of the aero maps, which represent the dynamics at each respective stationary point, resulting in control gains well tuned to the aerodynamical behaviour. Second, the PDC approach integrates the local controllers into a TS structure using the same membership functions as the underlying design model, such that a stability analysis by means of LMI criteria is possible.

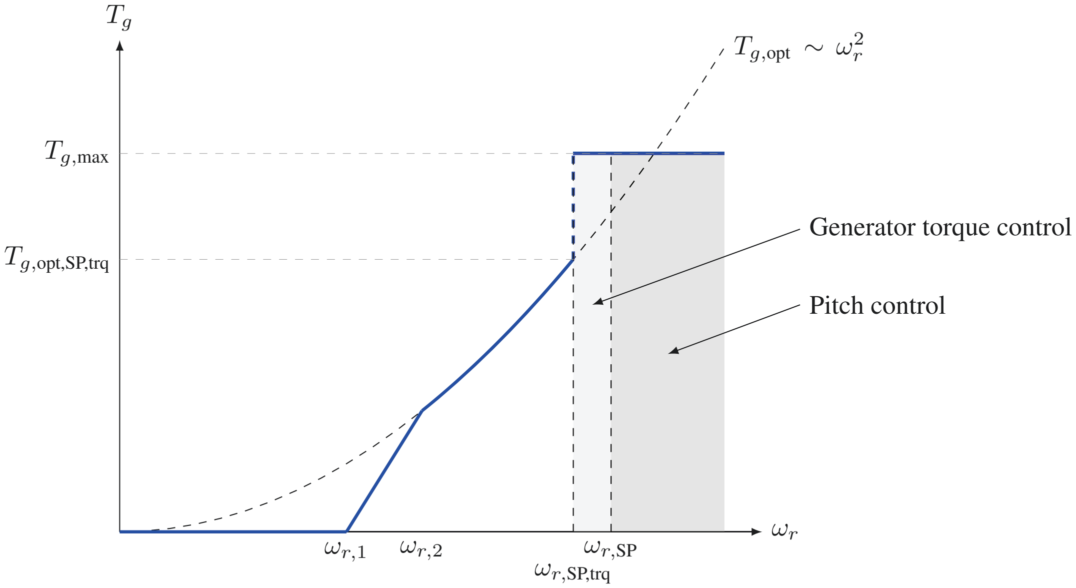


Figure 4.1: Regions of wind turbine control; Generator torque in dependence on the rotor angular speed

This does not mean that the chosen design method necessarily leads to a most optimal control in terms of energy yield or load reduction. An array of different methods is available for approaching wind turbine control, each with certain strengths and weaknesses. The works in [Boukhezzar and Siguerdidjane, 2005, 2010] present dedicated nonlinear state feedback control algorithms fed with an estimated wind speed using a Kalman Filter. In [Taher et al., 2013], a gain-scheduled controller is presented, where the local LQR weights are obtained from optimisation algorithms. In [Nourdine et al., 2010], an individual blade pitch control strategy is included in an LQG design to achieve fatigue load reduction (see also discussion in Section 4.1.3). In [Chadli and El Hajjaji, 2010], an integrated LMI controller design is conducted based on linearised models. However, only 4 local models are used, which can hardly guarentee a sufficient control performance. In [Bianchi et al., 2007], gain-scheduling controllers are implemented based on linear parameter varying (LPV) systems. An early treatment of LQR control for wind turbines is given in [Ekelund, 1997]. A standard gain-scheduling technique (with hysteresis) is used to switch between the linear controllers, such that no global stability analysis can be conducted for the complete full load region. As for the partial load region, [Corradini et al., 2013] present a sliding mode field-oriented control strategy utilising an aerodynamic torque observer. In [Galdi et al., 2008, 2009], fuzzy clustering methods are employed to obtain a TS model for maximum energy extraction in the partial load region.
Besides [Bossanyi, 2000; Burton et al., 2011], another overview of control strategies can be found in [Leithead and Connor, 2000]. For a recent, fairly comprehensive review of different methods and strategies see [Munteanu et al., 2008].

The control strategy used in this work achieves a reasonable control performance and constitutes a good compromise between simplicity and complexity. Moreover, the chosen TS controller structure allows to conduct stability analyses based on LMI condtions.

4.1 Full Load Region - Pitch Control

In the full load region, the generator torque is kept at a constant value of $T_g = T_{g,\max}$. For the pitch control design in the full load region, the wind turbine state-space model (2.29) is thus considered without an explicit torque state and using only the pitch angle as input. Then, (2.29) is modified to

$$\begin{aligned} \dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, u) &= \mathbf{A}\,\mathbf{x} + \mathbf{B}\,u + \mathbf{g}(\mathbf{x}, v) + \mathbf{a}\,, \\ \mathbf{y} &= \mathbf{C}\,\mathbf{x}\,, \end{aligned} \tag{4.1}$$

where the input $u = \beta_d$ is the desired pitch angle, the state vector is given by $\mathbf{x} = \begin{pmatrix} y_T & y_B & \theta_r & \theta_g & \dot{y}_T & \dot{y}_B & \omega_r & \omega_g \end{pmatrix}^T$ and $\mathbf{a}$ is a constant affine vector that can be easily obtained by considering the right column of the matrix $\mathbf{A}$ in (2.29):

$$\mathbf{a} = \begin{pmatrix} \mathbf{0}_{6x1} \\ -\frac{1}{J_g}\,T_{g,\max} \\ 0 \end{pmatrix}\,. \tag{4.2}$$

$\mathbf{A}$ and $\mathbf{B}$ are the modified system and input matrices [1]

$$\mathbf{A} = \begin{pmatrix} \mathbf{0}_{3\times3} & \mathbf{L}_{34} & \mathbf{0}_{3\times1} \\ -\mathbf{M}^{-1}\tilde{\mathbf{K}} & -\mathbf{M}^{-1}\mathbf{D} & \mathbf{0}_{4\times1} \\ \mathbf{0}_{1\times3} & \mathbf{0}_{1\times4} & -\frac{1}{\tau} \end{pmatrix}\,, \qquad \mathbf{B} = \begin{pmatrix} \mathbf{0}_{7x1} \\ \\ \frac{1}{\tau} \end{pmatrix}\,. \tag{4.3}$$

In Sections 4.1.3 and 4.1.4, two controller designs are conducted based on an 8-state respectively 2-state model. In both cases, all states are assumed to be measurable, i.e. the output matrix $\mathbf{C}$ is set equal to the identity matrix with the respective dimension.

4.1.1 Linearisation

In first tests it could be verified that an LMI-based control design using the TS model (3.18) obtained with the sector nonlinearity approach does not lead to a satisfying controller performance. The reason for this is probably that the aerodynamic information from the rotor torque and trust coefficients C_Q and C_T is not included in the TS submodels obtained with the sector nonlinearity approach. This information, however, is vital to design a pitch controller with a satisfying controller performance, because the aerodynamic sensitivity of the rotor blades, which is given by the derivatives of C_Q with respect to λ and β, varies with these two latter quantities. In particular, with increasing pitch angle, the aerodynamic sensitivity of a rotor blade increases.

For these reasons, the pitch control design in this work is based on a Taylor linearised model, where for each linear model a separate linear controller is designed. Afterwards, the local linear controllers are combined to a global nonlinear controller for the complete full load region by fuzzy-blending the linear controllers.

Stationary Points

Prior to designing the controllers for the local linear submodels, the stationary points of the nonlinear system must be determined. For all linear submodels, the stationary value of rotor and generator speed in

---

[1] $\mathbf{B}$ is a vector in this case. Nevertheless, a capital letter is used throughout, as well as for the disturbance matrix/vector $\mathbf{E}$ (see below), to be consistent and to avoid confusion with the notation for the error vector $\mathbf{e}$.

the full load region is equal to the rotor speed setpoint, which is given by $\omega_{r,SP} = 1.267$ [Jonkman et al., 2009]: $\omega_{r,i} = \omega_{g,i} = \omega_{r,SP}$.

The stationary pitch angle values were determined by simulating the nonlinear model at constant wind speeds (see table 4.1 and figure 4.2). In this case, the analytical functions (2.37) and (2.39) for C_Q and C_T were used, because the controller design is also carried out using the analytical approximations of the aero maps.

v [$\frac{\mathrm{m}}{\mathrm{s}}$]	β [deg]	v [$\frac{\mathrm{m}}{\mathrm{s}}$]	β [deg]	v [$\frac{\mathrm{m}}{\mathrm{s}}$]	β [deg]	v [$\frac{\mathrm{m}}{\mathrm{s}}$]	β [deg]	v [$\frac{\mathrm{m}}{\mathrm{s}}$]	β [deg]
11.26	0	11.7	2.6	12.5	5.9	14.5	10.7	18.0	15.5
11.3	0.26	11.8	3.1	12.7	6.5	15.0	11.5	20.0	17.5
11.4	0.9	11.9	3.6	13.0	7.4	15.5	12.3	22.0	19.3
11.5	1.5	12.0	4	13.5	8.6	16.0	13	24.0	20.9
11.6	2.1	12.2	4.8	14.0	9.7	17.0	14.3	26.0	22.5

Table 4.1: Stationary pitch angle values in the full load region in dependence on the wind speed, calculated using the nonlinear function (2.37) for C_Q.

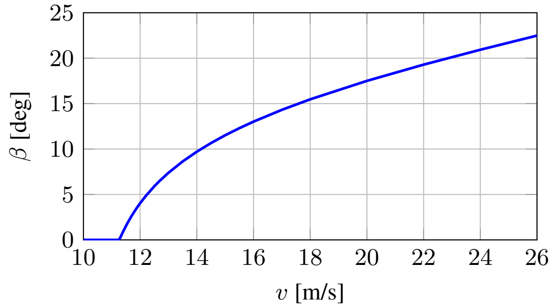


Figure 4.2: Stationary pitch angle values in the full load region in dependence on the wind speed, calculated using the nonlinear function (2.37) for C_Q.

Approximated TS Model for the Full Load Region

The nonlinear state-space model (4.1) is linearised around the stationary points listed in Table 4.1 and then approximated as a weighted combination of linear submodels:

$$\dot{\mathbf{x}} = \sum_{i=1}^{N_r} h_i(\beta)\left(\mathbf{A}_i \Delta\mathbf{x}_i + \mathbf{B}_i \Delta u_i + \mathbf{E}_i \Delta v_i\right) , \tag{4.4}$$

where N_r denotes the number of linear submodels. Here, a relatively large number number of $N_r = 25$ stationary points was used in order to reduce approximation errors. The difference terms are given by $\Delta\mathbf{x}_i = \mathbf{x} - \mathbf{x}_i$, $\Delta u_i = \beta_d - \beta_i$, $\Delta v_i = v - v_i$. Triangular functions, which fulfill the convex sum condition $\sum_i h_i = 1$, are used as membership functions in dependence on the pitch angle (see figure 4.3). It is common practice to use the pitch angle as scheduling variable instead of the wind speed for two reasons. First, the nacelle anemometer measurement of the wind speed is simply a single-point measurement which is no good estimate of the full wind field acting on the rotor. By contrast, the pitch angle, which is adjusted such that the rotor speed is kept at the desired setpoint, inherently reflects the rotor effective wind. Second, its measurement is less noisy and more reliable.

The matrices for the linear submodels are given by the Jacobi matrices of the complete nonlinear function $\mathbf{f}$ in Equation (4.1) with respect to the states $\mathbf{x}$, the input $u = \beta$ and the disturbance v, evaluated at the corresponding stationary points.

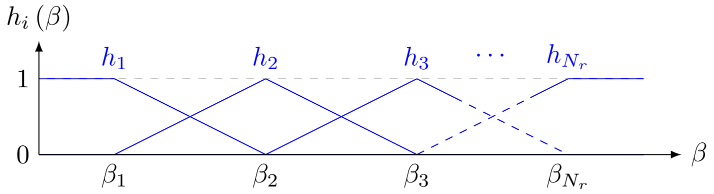


Figure 4.3: Triangular membership functions for approximated TS model. Figure reprinted from [Georg and Schulte, 2013] © 2013 IEEE.

The matrices $\mathbf{A}_i$ are given by

$$\mathbf{A}_i = \left.\frac{\partial \mathbf{f}}{\partial \mathbf{x}}\right|_i = \mathbf{A} + \left.\frac{\partial \mathbf{g}}{\partial \mathbf{x}}\right|_i , \tag{4.5}$$

where $\frac{\partial \mathbf{g}}{\partial \mathbf{x}}$ contains the derivatives of rotor thrust and torque:

$$\frac{\partial \mathbf{g}}{\partial \mathbf{x}} = \begin{pmatrix} & \mathbf{0}_{4\times 8} & & \\ \mathbf{0}_{1\times 5} & \frac{1}{N m_B}\frac{\partial F_T}{\partial \omega_r} & 0 & \frac{1}{N m_B}\frac{\partial F_T}{\partial \beta} \\ \mathbf{0}_{1\times 5} & \frac{1}{J_r}\frac{\partial T_a}{\partial \omega_r} & 0 & \frac{1}{J_r}\frac{\partial T_a}{\partial \beta} \\ & \mathbf{0}_{2\times 8} & & \end{pmatrix} . \tag{4.6}$$

Since $\frac{\partial \mathbf{f}}{\partial u} = \mathbf{B}$, there is only one common input matrix, which is equal to $\mathbf{B}$ given in (4.3). The disturbance matrices, which are vectors in this case, are given by

$$\mathbf{E}_i = \left.\frac{\partial \mathbf{f}}{\partial v}\right|_i = \left.\frac{\partial \mathbf{g}}{\partial v}\right|_i = \left.\begin{pmatrix} \mathbf{0}_{4\times 1} \\ \frac{1}{N m_B}\frac{\partial F_T}{\partial v} \\ \frac{1}{J_r}\frac{\partial T_a}{\partial v} \\ \mathbf{0}_{2\times 1} \end{pmatrix}\right|_i . \tag{4.7}$$

The disturbance matrices are not needed for the controller design but only for validating the approximated TS model (4.4) against the nonlinear model (4.1).

Derivatives of the Rotor Thrust and Torque

Using the aerodynamic equations (2.24), (2.25) and the definition (2.26) of the reference force $F_{\text{st}} = \frac{1}{2}\,\rho\,\pi\,R^2\,v^2$, the derivatives of the rotor thrust and torque with respect to the rotor speed, pitch angle and wind speed can be calculated as follows, where the derivatives of the aero maps are expressed in terms of derivatives with respect to λ and β, since C_Q and C_T are also defined in terms of these two variables:

$$\frac{\partial F_T}{\partial \omega_r} = F_{\text{st}}\,\frac{\partial C_T}{\partial \omega_r} = F_{\text{st}}\,\frac{\partial C_T}{\partial \lambda}\,\frac{\partial \lambda}{\partial \omega_r} = F_{\text{st}}\,\frac{\partial C_T}{\partial \lambda}\,\frac{R}{v} \tag{4.8}$$

$$\frac{\partial F_T}{\partial \beta} = F_{\text{st}} \frac{\partial C_T}{\partial \beta} \tag{4.9}$$

$$\begin{aligned} \frac{\partial F_T}{\partial v} &= \frac{\partial F_{\text{st}}}{\partial v} C_T + F_{\text{st}} \frac{\partial C_T}{\partial v} \\ &= 2 \frac{F_{\text{st}}}{v} C_T + F_{\text{st}} \frac{\partial C_T}{\partial \lambda} \frac{\partial \lambda}{\partial v} = 2 \frac{F_{\text{st}}}{v} C_T - F_{\text{st}} \frac{\partial C_T}{\partial \lambda} \left(R \frac{\omega_r}{v^2} \right) \end{aligned} \tag{4.10}$$

$$\frac{\partial T_a}{\partial \omega_r} = F_{\text{st}} R \frac{\partial C_Q}{\partial \omega_r} = F_{\text{st}} R \frac{\partial C_Q}{\partial \lambda} \frac{R}{v} \tag{4.11}$$

$$\frac{\partial T_a}{\partial \beta} = F_{\text{st}} R \frac{\partial C_Q}{\partial \beta} \tag{4.12}$$

$$\begin{aligned} \frac{\partial T_a}{\partial v} &= \frac{\partial F_{\text{st}}}{\partial v} R C_Q + F_{\text{st}} R \frac{\partial C_Q}{\partial v} \\ &= 2 \frac{F_{\text{st}}}{v} R C_Q + F_{\text{st}} R \frac{\partial C_Q}{\partial \lambda} \frac{\partial \lambda}{\partial v} = 2 \frac{F_{\text{st}}}{v} R C_Q - F_{\text{st}} R^2 \frac{\partial C_Q}{\partial \lambda} \frac{\omega_r}{v^2} \end{aligned} \tag{4.13}$$

Directional Derivatives of the Analytical Aero Maps

The above formulas (4.8) - (4.13) for the derivatives of rotor thrust and torque contain the directional derivatives of the C_T and C_Q aero maps with respect to the tip speed ratio and to the pitch angle. For the controller design, these derivatives should not be calculated from the tabulated aero maps extracted from FAST simulations, because in this case the derivatives would contain discontinuities that would lead to distorted results. Thus, the analytical forms of the aero maps presented in section 2.3.3, in particular the unlimited functions $\tilde{C}_Q$ (2.36) and $\tilde{C}_T$ (2.38) should be used to calculate the directional derivatives.

Derivatives of the rotor thrust coefficient C_T

$$\tilde{C}_T(\lambda, \beta) = a_1 + a_2 (\lambda - a_3 \beta) e^{-a_4 \beta} + a_5 \lambda^2 e^{-a_6 \beta} + a_7 \lambda^3 e^{-a_8 \beta}, \tag{4.14}$$

$$\frac{\partial \tilde{C}_T}{\partial \lambda} = a_2 e^{-a_4 \beta} + 2 a_5 \lambda e^{-a_6 \beta} + 3 a_7 \lambda^2 e^{-a_8 \beta}, \tag{4.15}$$

$$\frac{\partial \tilde{C}_T}{\partial \beta} = a_2 (a_3 a_4 \beta - a_4 \lambda - a_3) e^{-a_4 \beta} - a_5 a_6 \lambda^2 e^{-a_6 \beta} - a_7 a_8 \lambda^3 e^{-a_8 \beta}. \tag{4.16}$$

Derivatives of the rotor torque coefficient C_Q

$$\begin{aligned} \tilde{C}_Q(\lambda, \beta) = {} & c_1 \left(1 + c_2 (\beta + c_3)^{\frac{1}{2}} \right) \\ & + \frac{c_4}{\lambda} (c_5 \lambda_i(\lambda, \beta) - c_6 \beta - c_7 \beta^{c_8} - c_9) \exp(-c_{10} \lambda_i(\lambda, \beta)) \quad (\lambda > 0), \end{aligned} \tag{4.17}$$

with $\lambda_i(\lambda, \beta) = \dfrac{1}{\lambda + 0.08\beta} - \dfrac{0.035}{c_{11} + c_{12}\beta^3} \quad (\lambda > 0)$.

To obtain the derivatives of $\tilde{C}_Q$, Equation (4.17) is first written as:

$$\tilde{C}_Q(\lambda, \beta) = c_1 \left(1 + c_2 (\beta + c_3)^{\frac{1}{2}} \right) + f_1 f_2 f_3, \tag{4.18}$$

with

$$\begin{aligned}
f_1 &= \frac{c_4}{\lambda}\,, \\
f_2 &= c_5\,\lambda_i\,(\lambda,\beta) - c_6\,\beta - c_7\,\beta^{c_8} - c_9\,, \\
f_3 &= \exp\left(-c_{10}\,\lambda_i\,(\lambda,\beta)\right)\,.
\end{aligned}$$

The derivatives of these three functions with respect to λ and β are given by:

$$\begin{aligned}
\frac{\partial f_1}{\partial \lambda} &= -\frac{c_4}{\lambda^2} \\
\frac{\partial f_2}{\partial \lambda} &= c_5\,\frac{\partial \lambda_i\,(\lambda,\beta)}{\partial \lambda} \\
\frac{\partial f_3}{\partial \lambda} &= -c_{10}\,\exp\left(-c_{10}\,\lambda_i\,(\lambda,\beta)\right)\,\frac{\partial \lambda_i\,(\lambda,\beta)}{\partial \lambda} \\
\frac{\partial f_1}{\partial \beta} &= 0 \\
\frac{\partial f_2}{\partial \beta} &= c_5\,\frac{\partial \lambda_i\,(\lambda,\beta)}{\partial \beta} - c_6 - c_7\,c_8\,\beta^{c_8-1} \\
\frac{\partial f_3}{\partial \beta} &= -c_{10}\,\exp\left(-c_{10}\,\lambda_i\,(\lambda,\beta)\right)\,\frac{\partial \lambda_i\,(\lambda,\beta)}{\partial \beta}\,,
\end{aligned}$$

with

$$\begin{aligned}
\frac{\partial \lambda_i\,(\lambda,\beta)}{\partial \lambda} &= -\frac{1}{(\lambda + 0.08\,\beta)^2}\,, \\
\frac{\partial \lambda_i\,(\lambda,\beta)}{\partial \beta} &= -\frac{0.08}{(\lambda + 0.08\,\beta)^2} + \frac{3\,\cdot\,0.035\,c_{12}\,\beta^2}{(c_{11} + c_{12}\,\beta^3)^2}\,.
\end{aligned}$$

The directional derivatives of the analytical aero maps are thus given by:

$$\frac{\partial \tilde{C}_Q}{\partial \lambda} = \frac{\partial f_1}{\partial \lambda}\,f_2\,f_3 + f_1\,\frac{\partial f_2}{\partial \lambda}\,f_3 + f_1\,f_2\,\frac{\partial f_3}{\partial \lambda}\,, \tag{4.19}$$

$$\frac{\partial \tilde{C}_Q}{\partial \beta} = \frac{\partial f_1}{\partial \beta}\,f_2\,f_3 + f_1\,\frac{\partial f_2}{\partial \beta}\,f_3 + f_1\,f_2\,\frac{\partial f_3}{\partial \beta} + \frac{1}{2}\,c_1\,c_2\,\left(\beta + c_3\right)^{-\frac{1}{2}}\,. \tag{4.20}$$

Note that the functions f_1, f_2, f_3 here should not be confused with the nonlinear functions in the sector-nonlinearity derivation of the TS model in chapter 3.

4.1.2 Parallel Distributed Compensation Control

For each linear submodel of the approximated TS system (4.4), a linear state-space controller structure

$$u_i = -\mathbf{k}_i^T \, \Delta\mathbf{x}_i \tag{4.21}$$

with the vector of control gain factors $\mathbf{k}_i$ can be found such that the autonomous closed-loop system

$$\dot{\mathbf{x}} = \sum_{i=1}^{N_r} h_i(\beta) \, \mathbf{A}_{\mathrm{CL},i} \, \Delta\mathbf{x}_i \tag{4.22}$$

is stabilised if all closed-loop matrices $\mathbf{A}_{\mathrm{CL},i} = \mathbf{A}_i - \mathbf{B}\,\mathbf{k}_i^T$ are stable. The complete system includes the bounded external disturbance term $\sum_i^{N_r} h_i(\beta)\,\mathbf{E}_i\,\Delta v_i$:

$$\dot{\mathbf{x}} = \sum_{i=1}^{N_r} h_i(\beta) \, \mathbf{A}_{\mathrm{CL},i} \, \Delta\mathbf{x}_i + \sum_{i=1}^{N_r} h_i(\beta) \, \mathbf{E}_i \, \Delta v_i \, . \tag{4.23}$$

For systems with external inputs, the notion of input-to-state stability was introduced by Sontag [Sontag, 1989, 1990; Sontag and Wang, 1995]. For TS systems with bounded external inputs, a stable autonomous system ensures the input-to-state stability of the complete system including the disturbance [Lendek et al., 2010]. This will be exploited in the stability analysis on page 48 in Section 4.1.4.

The form of the closed-loop system in Equation (4.23) could also be derived by interpreting the term

$$u = -\sum_{i=1}^{N_r} h_i(\beta) \, \mathbf{k}_i^T \, \Delta\mathbf{x}_i \tag{4.24}$$

as an external, collective control input for a TS model $\dot{\mathbf{x}} = \sum_{i=1}^{N_r} h_i(\beta)\,(\mathbf{A}_i\,\Delta\mathbf{x}_i + \mathbf{E}_i\,\Delta v_i) + \mathbf{B}\,u$, which would be equivalent to the structure (4.4) controlled with the individual local control laws defined in (4.21).
A controller structure like the one in (4.24) is called Parallel Distributed Compensation (PDC) and was introduced for TS systems in [Tanaka and Sano, 1994a; Wang et al., 1995a,b]. In order to formally prove the stability of a TS system controlled with a PDC control structure, the membership functions of the PDC control law must be equal to those of the controlled system. For the PDC controller in Equation (4.24), the membership functions are defined as depicted in figure 4.3.

The fact that the approximated TS model (4.4) can be controlled by means of a PDC structure motivates to use the same PDC also for controlling the nonlinear model (4.1) and even in FAST simulations. In these cases, the global stability cannot be formally proven. However, given the large number of 25 linear submodels, a successful verification of the stability of the approximated wind turbine TS model controlled with a PDC would give strong support for the assumption that the designed PDC controller would also be able to stabilise the nonlinear model (4.1) and the FAST model.

I-Augmented System

The local controllers in (4.21) and thus also the submodels of the PDC control structure in (4.24) are classical state-feedback control laws which feature only proportional terms[2]. Using these structures, it is

---

[2] In the case of the rotor speed, this state ω_r was corrected by the reference control signal $\omega_{r,\mathrm{SP}}$.

not possible to achieve steady-state accuracy in the whole full-load region that is robust against changes in the wind speed. To realize this requirement, an integral control term needs to be added to the control structure, which is implemented here by introducing an additional state into the state-space model for controller design:

$$x_9 \,=\, x_I \,:=\, \int\limits_0^t \left(\omega_r(\tau) - \omega_{r,\text{SP}}\right)\, d\tau\,, \tag{4.25}$$

i.e. $\dot{x}_9 \,=\, x_6 - \omega_{r,\text{SP}}$. As a new state vector one obtains $\tilde{\mathbf{x}} \,=\, \left(\mathbf{x}^T\, x_I\right)^T$ and the matrices for the linear submodels in Equations (4.5) - (4.7) are augmented correspondingly:

$$\tilde{\mathbf{A}}_\mathbf{i} \,=\, \begin{pmatrix} \mathbf{A}_i & \mathbf{0}_{8\times 1} \\ \mathbf{CI} & 0 \end{pmatrix}\,, \quad \text{with} \quad \mathbf{CI} = \begin{pmatrix} \mathbf{0}_{1\times 5} & 1 & \mathbf{0}_{1\times 2} \end{pmatrix}\,, \tag{4.26}$$

$$\tilde{\mathbf{B}} \,=\, \begin{pmatrix} \mathbf{B} \\ 0 \end{pmatrix}\,, \quad \tilde{\mathbf{E}}_i \,=\, \begin{pmatrix} \mathbf{E}_i \\ 0 \end{pmatrix}\,. \tag{4.27}$$

The approximated TS system (4.4) is then modified to

$$\dot{\tilde{\mathbf{x}}} \,=\, \sum_{i=1}^{N_r} h_i\left(\beta\right)\left(\tilde{\mathbf{A}}_i\,\Delta\tilde{\mathbf{x}}_i \,+\, \tilde{\mathbf{B}}\,\Delta u_i \,+\, \tilde{\mathbf{E}}_i\,\Delta v_i\right)\,. \tag{4.28}$$

Equation (4.28) can also be written as an affine TS system with an external disturbance:

$$\dot{\tilde{\mathbf{x}}} \,=\, \sum_{i=1}^{N_r} h_i\left(\beta\right)\left(\tilde{\mathbf{A}}_i\,\tilde{\mathbf{x}} \,+\, \tilde{\mathbf{B}}\,u \,+\mathbf{a}_i\right) \,+\, \sum_{i=1}^{N_r} h_i\left(\beta\right)\tilde{\mathbf{E}}_i\,v\,, \tag{4.29}$$

with the affine vector

$$\mathbf{a}_i \,:=\, -\left(\tilde{\mathbf{A}}_i\,\tilde{\mathbf{x}}_i \,+\, \tilde{\mathbf{B}}\,u_i \,+\tilde{\mathbf{E}}_i\,v_i\right)\,. \tag{4.30}$$

System (4.28) shall be termed 8+I-state control design model.

The PDC controller for the I-augmented system is given by the structure in (4.24) plus an additional term due to the integrator state:

$$u \,=\, -\sum_{i=1}^{N_r} h_i\left(\beta\right)\left(\mathbf{k}_i^T\,\Delta\mathbf{x}_i \,+\, k_{I,i}\,x_I\right)\,, \tag{4.31}$$

where $k_{I,i}$ denote the control gains for the integral part. The modified control structure (4.31) was chosen in consideration of the role of the integral part in establishing steady-state accuracy. Consider a (virtual) scenario where the system states exactly match one of the stationary points j, i.e. $\Delta\mathbf{x}_j = \mathbf{0}$ and $x_I = x_{I,j}$, and the wind speed remains constant. In such a case, the pitch angle of the blades would be solely kept at its value by the integrator state, which corresponds to a memory state of the system. In this case, the PDC controller in Equation (4.31) would reduce to $u = u_j = -k_{I,j}\,x_{I,j}$.

The derivation of the affine TS system structure (4.29) in connection with the modified PDC controller (4.31) is given in more detail in appendix A.3.

4.1.3 Multivariable LQR Controller Design

In this section, the controller matrices $\tilde{\mathbf{k}}_i^T = \begin{pmatrix}\mathbf{k}_i^T & k_{I,i}\end{pmatrix}$ for the PDC controller (4.31) are designed based on the 8+I-state model separately for each linear submodel by means of the LQR design method. For an introduction into this method, the reader is referred to standard textbooks on control theory, for example [Franklin et al., 1994].

The LQR controller design minimises the functional

$$J = \int_0^\infty \left(\tilde{\mathbf{x}}^T \mathbf{Q} \tilde{\mathbf{x}} + u^T R u\right) d\tau , \tag{4.32}$$

with the weighting matrices $\mathbf{Q}$ and R. In order for these to be determined, the maximal absolute values of the contributing states first have to be estimated. For the wind turbine model, the following estimates are taken:

$y_{T,\max} = 1.5\,\text{m}$, $y_{B,\max} = 10\,\text{m}$, $\theta_{s,\max} = 0.002\,\text{rad}$, $\dot{y}_{T,\max} = 0.2\,\frac{\text{m}}{\text{s}}$, $\dot{y}_{B,\max} = 2\,\frac{\text{m}}{\text{s}}$,
$\omega_{g,\max} = \omega_{r,\max} = 1.57\,\frac{\text{rad}}{\text{s}}$, $\beta_{\max} = 1.57\,\text{rad}$

Subsequently, the individual states are weighted with relative weights, where the state $x_6 = \omega_r$ is weighted most since it is the primary quantity to be controlled:

$Q_1 = 0$, $Q_2 = 0$, $Q_3 = 2 \cdot 10^{-9}$, $Q_4 = 0.02$, $Q_5 = 0.1$, $Q_6 = 1.57$, $Q_7 = 0$, $Q_8 = 1.5706 \cdot 10^{-6}$, $Q_I = 0.05$.

In the final weighting matrix, the relative weights are devided by the squares of the estimated maximum absolute values to take account of the magnitudes of the individual states:

$$\mathbf{Q} = \text{diag}\begin{pmatrix} \frac{Q_1}{y_{T,\max}^2} & \frac{Q_2}{y_{B,\max}^2} & \frac{Q_3}{\theta_{s,\max}^2} & \frac{Q_4}{\dot{y}_{T,\max}^2} & \frac{Q_5}{\dot{y}_{B,\max}^2} & \frac{Q_6}{\omega_{r,\max}^2} & \frac{Q_7}{\omega_{g,\max}^2} & \frac{Q_8}{\beta_{\max}^2} & Q_I \end{pmatrix} . \tag{4.33}$$

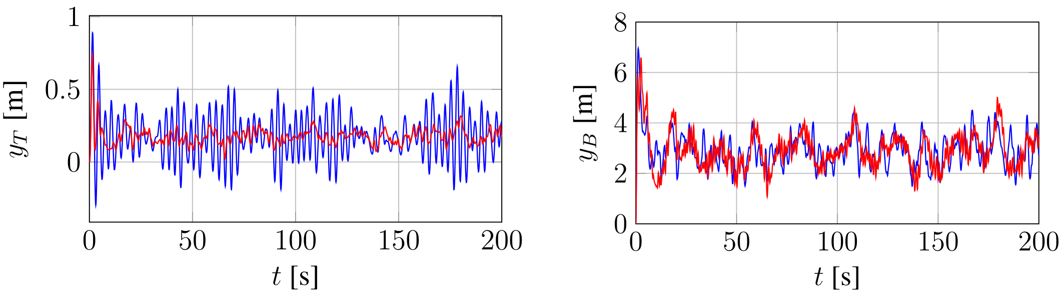
The weight for the control input is chosen as $R = 1$.

The thus designed controller shall be termed 8-I-state LQR controller.

A similar ansatz (I-augmented model with LQR controller design) was also employed in
[Dolan and Aschemann, 2013].

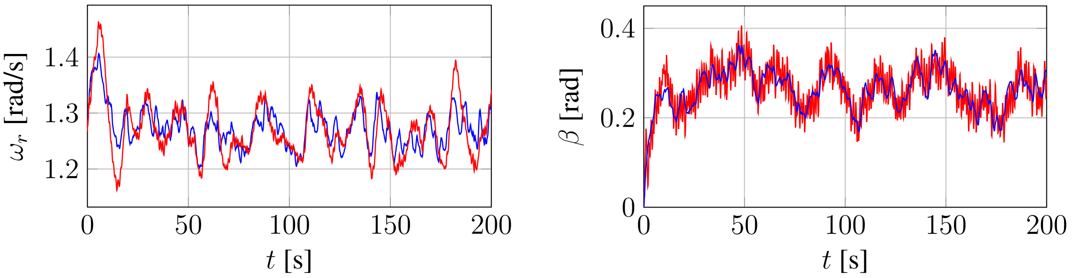
Load Reduction - Damped Tower Oscillations

For a precise rotor speed control, only the rotor speed state ω_r would have to be included into the control design model. Including also the other states may potentially lead to damped oscillations and thereby a load reduction of other degrees of freedom, for example the tower oscillations. Since the frequency of the (fore-aft) tower oscillation is comparably low (between 0.2 and 0.5 Hz for standard turbine sizes between 1 and 3 MW), this oscillation can in principle be damped using pitch control. One way to achieve this is by adding an additional term $\delta\beta = \frac{-D_p}{\partial F_T/\partial\beta}\,\dot{y}_T$ to the pitch controller, which is proportional to the fore-aft tower top velocity [Bossanyi, 2000; Burton et al., 2011]. Here, D_p is a design damping constant. An integrated LQR design, however, would be a more elegant and consistent way, because the weights for the different states could be tuned such that an optimal balance of load reduction in the individual degrees of freedom could be attained. Some works have dealt with wind turbine LQR controller design for load reduction, like [Nourdine et al., 2010]. In [Nourdine et al., 2010], however, the simulations to verify the load reductions are only carried out using a model with the same degrees of freedom as the control design model. For a realistic estimation, though, a standard wind turbine load simulation code like FAST should be used.



(a) Tower top displacement

(b) Blade tip displacement



(c) Rotor speed

(d) Pitch angle

Figure 4.4: Comparison of 2+I-state (blue lines) and 8+I-state (red lines) LQR controller designs in a simulation of the nominal model using turbulent wind input (mean wind speed: 18 m/s).

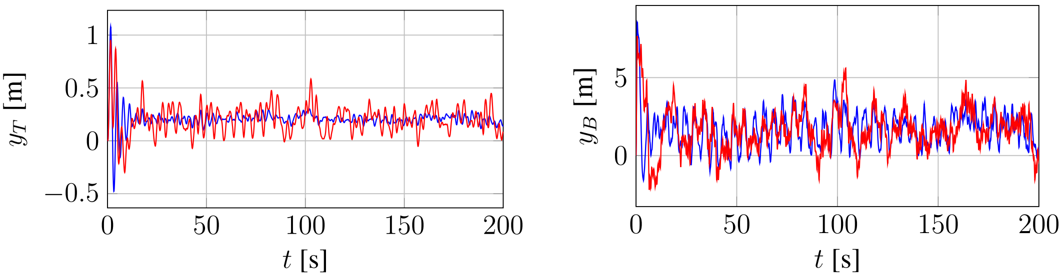
In this work, the controller obtained with the LQR design described above was applied in simulations using both the nominal 9-state wind turbine model (2.29) with four degrees of freedom and FAST. Also, the 8+I-state LQR design was compared to a 2+I-state LQR controller based on a further reduced model including only the rotor speed and the pitch angle plus the integrator state (see section 4.1.4 for the description of the reduced-order design model and the chosen LQR weights). In Figure 4.4, the tower top displacement y_T, the blade tip displacement y_B, the rotor speed ω_r and the pitch angle β are plotted using both the 8+I-state and the 2+I-state controller in the nominal 9-state simulation model (2.29) which shall from now on be termed "nominal" model for the sake of simplicity.

As can be seen, in the nominal model simulation, a substantial damping of the tower oscillation is achieved with the 8+I-state LQR controller compared to the 2+I-state controller, at the price of a slightly suboptimal behaviour of the controlled rotor speed and an increased pitch activity. In order to exclude too high pitch rates, the pitch rate was limited to 10 deg/s, which is a typical value for electromechanical pitch systems.

When using the same controllers in FAST, however, not only is there no tower damping, but rather an amplification of the tower oscillation (see Figure 4.5)! Also, the controlled rotor speed behaviour is unacceptable. It was also tested whether a better load reduction behaviour can be achieved when designing the controller based on a model without the blade tip degree of freedom, i.e. a 6+I model. This however, yielded no improvement and neither did an omission of the pitch angle state in the control design model.

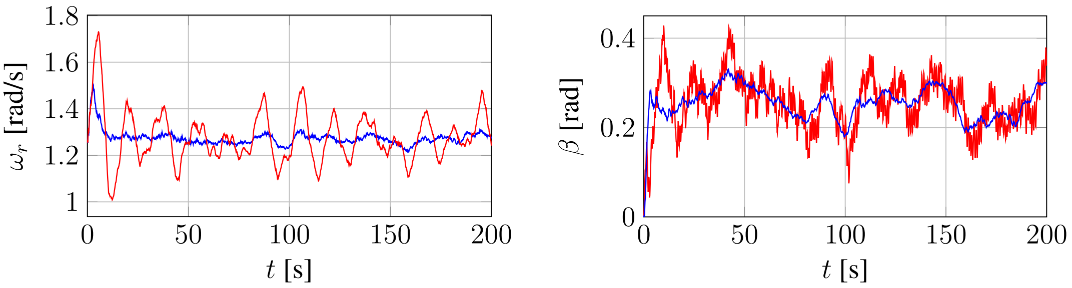
A better load reduction behaviour could probably be achieved by putting more effort into tuning the LQR gains or by further improving the design model. However, this also shows that there is not much robustness in a load reduction controller for wind turbines, which illustrates both the potential and the problem of multivariable state-space control design for wind turbines. There is, in principle, a high potential for load reduction, but a controller that leads to a load reduction in simulations of the design model does not guarantee that a load reduction is also achieved in more detailed aero-elastic simulations, let alone in a real wind turbine.

In order to realize additional tower damping on a real turbine, it may thus be necessary to add a lead-lag filter for the tower feedback-loop to adjust the phase of the additional pitch command [Burton et al.,



(a) Tower top displacement

(b) Blade tip displacement



(c) Rotor speed

(d) Pitch angle

Figure 4.5: Comparison of 2+I-state (blue lines) and 8+I-state (red lines) LQR controller designs in a FAST simulation using turbulent wind input (mean wind speed: 18 m/s).

2011]. From a control theory standpoint, this would go against the grain, but it shows the practical limits of applying pure multivariable control strategies without any further modifications.

4.1.4 Controller Design with a Reduced-Order 2-State Model

In view of the limitations described in the previous section to exploit the load reduction possibilities of the 8+I-state LQR controller in FAST simulations and since load reduction is not the primary goal of this work, a controller based on a further reduced model with only two states, rotor speed and pitch angle, is designed and used throughout the rest of this work. A global stability analysis is conducted at the end of this section. The reduced-order control design and the global stability analysis were first published in [Georg and Schulte, 2013], though the stability analysis has been amended in this work.

The state vector for the control design model is thus given by $\mathbf{x} = (\omega_r \quad \beta)^T$ and the system dynamics of rotor speed and pitch angle are described by

$$\dot{\mathbf{x}} = \begin{pmatrix} \dot{\omega}_r \\ \dot{\beta} \end{pmatrix} = \begin{pmatrix} \frac{1}{J_r + J_g}\left(T_a - T_g\right) \\ -\frac{1}{\tau}\beta + \frac{1}{\tau}\beta_d \end{pmatrix} . \tag{4.34}$$

Equation (4.34) represents a stiff drivetrain model plus the pitch actuator dynamics. The latter model is included to take account of the delay time of the pitch actuator dynamics in the control design.
For the controller design, the model is augmented with an integrator state in the same way as described in Section 4.1.2: $\dot{x}_3 = \dot{x}_I := x_1 - \omega_{r,\mathrm{SP}}$. The modified system vector is thus given by $\tilde{\mathbf{x}} = \left(\mathbf{x}^T \ x_I\right)^T$.

The linearisation is done as described in section 4.1.1, whereas here only the derivatives of the rotor torque, and thus the derivatives of the C_Q map are needed. The following matrices for the linear submodels are obtained:

$$\tilde{\mathbf{A}}_i = \left.\begin{pmatrix} \frac{1}{J_r+J_g}\frac{\partial T_a}{\partial \omega_r} & \frac{1}{J_r+J_g}\frac{\partial T_a}{\partial \beta} & 0 \\ 0 & -\frac{1}{\tau} & 0 \\ 1 & 0 & 0 \end{pmatrix}\right|_i , \quad \tilde{\mathbf{B}} = \begin{pmatrix} 0 \\ \frac{1}{\tau} \\ 0 \end{pmatrix} , \quad \tilde{\mathbf{E}}_i = \left.\begin{pmatrix} \frac{1}{J_r+J_g}\frac{\partial T_a}{\partial v} \\ 0 \\ 0 \end{pmatrix}\right|_i . \tag{4.35}$$

The approximated TS design system is given by

$$\dot{\tilde{\mathbf{x}}} = \sum_{i=1}^{N_r} h_i(\beta)\left(\tilde{\mathbf{A}}_i\,\Delta\mathbf{x}_i + \tilde{\mathbf{B}}\,\Delta u_i + \tilde{\mathbf{E}}_i\,\Delta v_i\right) . \tag{4.36}$$

As in section 4.1.2, the PDC control law is given by

$$u = -\sum_{i=1}^{N_r} h_i(\beta)\left(\mathbf{k}_i^T\,\Delta\mathbf{x}_i + k_{I,i}\,x_I\right) . \tag{4.37}$$

The same $N_r = 25$ stationary points as in the previous sections were used.

LQR Controller Design

The weights of the three states were selected as the corresponding states for the 8+I-state LQR design in section 4.1.3:

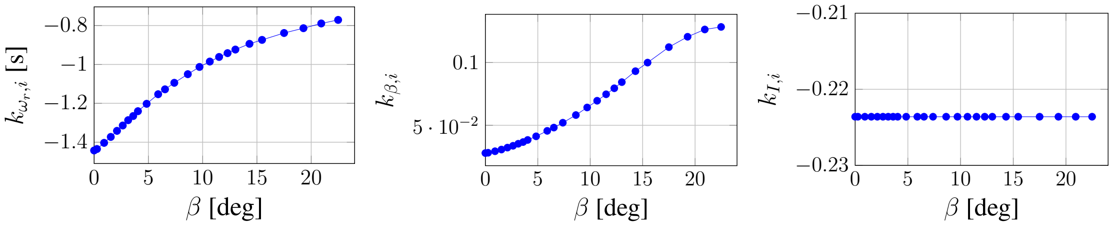
$Q_1 = 1.57$, $Q_2 = 1.57 \cdot 10^{-6}$, $Q_I = 0.05$. Analogously to the design in section 4.1.3, the normalisation with the respective estimated maximum values for rotor speed and pitch angle was considered for the final state weighting matrix:

$$\mathbf{Q} = \operatorname{diag}\begin{pmatrix} \frac{Q_1}{\omega_{r,\max}^2} & \frac{Q_2}{\beta_{\max}^2} & Q_I \end{pmatrix} , \tag{4.38}$$

and the weight for the control input was set to $R = 1$.

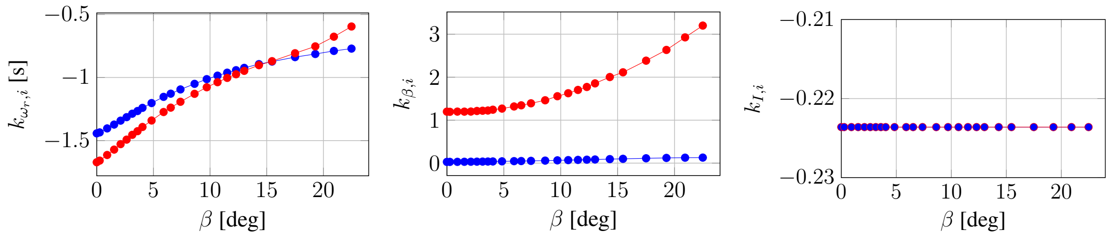
The thus-obtained controller gains in dependence on the pitch angle are depicted in Figure 4.6. It is well visible that the absolute values of the controller gains $k_{\omega_r,i}$ for the first state decrease with increasing pitch angle. This is due to the aerodynamic sensitivity of the rotor blades, which increases with increasing pitch angle, i.e., for larger pitch angles, smaller changes of the pitch angle are required to yield the same change in the driving aerodynamic rotor torque [Burton et al., 2011] (this can also be observed in Figure 4.2). This dependence is not visible in the controller gains for the integrator state, which is due to the fact that the LQR design method penalises the individual states and for the integrator state only one constant entry appears in the linearised system matrices $\tilde{\mathbf{A}}_i$ in (4.35). If pole assignment were used for the controller designs for the individual submodels, a similar dependence as for $k_{\omega_r,i}$ would be seen in the gains $k_{I,i}$.

In Figure 4.7, the controller gains for the 2+I-state design are compared to the corresponding gains for the 8+I-state controller design from Section 4.1.3. Whereas the dependence of the rotor speed gains $k_{\omega_r,i}$ on the pitch angle is similar in both controller designs, the pitch angle gains $k_{\beta,i}$ differ strongly, although the identical LQR weights were used for the respective states.



(a) Gains for the rotor speed (b) Gains for the pitch angle (c) Gains for the integrator state

Figure 4.6: Controller gains in dependence on the pitch angle for LQR controller designs based on the approximated 2+I-state TS model (4.36).



(a) Gains for the rotor speed (b) Gains for the pitch angle (c) Gains for the integrator state

Figure 4.7: Comparison of controller gains for 2+I-state (blue) and 8+I-state (red) LQR designs.

Global Stability Analysis - Full Load Region

Since the additional integrator state in the augmented system (4.36) is only relevant for the controller design, in this section it is investigated whether the global stability of the unaugmented approximated 2-state TS system

$$\dot{\mathbf{x}} = \sum_{i=1}^{N_r} h_i\,(\beta)\,(\mathbf{A}_i\,\Delta\mathbf{x}_i\; + \mathbf{B}\,\Delta u_i\; + \mathbf{E}_i\,\Delta v_i)\;, \tag{4.39}$$

with $\mathbf{x}$ as in (4.34), controlled with the PDC controller (4.37) is ensured for the whole full load region (see [Georg and Schulte, 2013]). As already introduced in section 4.1.2, the TS system (4.39) can also be written as an affine TS system with an external bounded disturbance $\sum_{i=1}^{N_r} h_i\,(\beta)\,\mathbf{E}_i\,v$:

$$\dot{\mathbf{x}} = \sum_{i=1}^{N_r} h_i\,(\beta)\,(\mathbf{A}_i\,\mathbf{x}\; + \mathbf{B}\,u\; + \mathbf{a}_i) + \sum_{i=1}^{N_r} h_i\,(\beta)\,\mathbf{E}_i\,v\,, \tag{4.40}$$

with the affine vector

$$\mathbf{a}_i \;:= -\left(\mathbf{A}_i\,\mathbf{x}_i\; + \mathbf{B}\,u_i\; + \mathbf{E}_i\,v_i\right)\;. \tag{4.41}$$

The PDC controller (4.37) can be written as

$$u = -\sum_{i}^{N_r} h_i(\beta)(\mathbf{k}_i^T\,\mathbf{x} + l_i)\; - \;k_I\,x_I\,, \tag{4.42}$$

with $l_i = -(k_{\omega_r,i}\,\omega_{r,\mathrm{SP}} + k_{\beta,i}\,\beta_i)$. Defining the affine term $\mathbf{g}_i := \mathbf{a}_i - \mathbf{B}\,l_i$, the corresponding closed-loop system can be written as

$$\dot{\mathbf{x}} = \sum_i^{N_r} h_i(\beta)\,(\mathbf{A}_{\mathrm{CL},i}\,\mathbf{x} + \mathbf{g}_i) \;-\; \mathbf{B}\,k_I\,x_I \;+ \sum_i^{N_r} h_i(\beta)\,\mathbf{E}_i\,v\,, \tag{4.43}$$

with the closed-loop matrices $\mathbf{A}_{\mathrm{CL},i} = \mathbf{A}_i - \mathbf{B}\,\mathbf{k}_i^T$.

As already mentioned in section 4.1.2, for systems with external disturbances, the input-to-state stability is the relevant property that needs to be verified. As shown in [Lendek et al., 2010], a TS system with bounded external inputs is input-to-state stable, if the Lyapunov stability of the corresponding autonomous TS system can be verified. Therefore, a Lyapunov stability analysis for the autonomous closed-loop TS system

$$\dot{\mathbf{x}} = \sum_i^{N_r} h_i(\beta)\,(\mathbf{A}_{\mathrm{CL},i}\,\mathbf{x} + \mathbf{g}_i) \;-\; \mathbf{B}\,k_I\,x_I, \tag{4.44}$$

will be conducted in the following.

Lyapunov Function The stability of an autonomous nonlinear system in general is ensured if a Lyapunov function can be found [Lyapunov, 1992; Khalil, 2002]. Two prerequisites have to be fulfilled for a function $V(\mathbf{x})$ to be a Lyapunov function: 1. $V(\mathbf{x}) > 0$ $(\mathbf{x} \in \mathbb{R}^n,\, \mathbf{x} \neq \mathbf{0})$. 2. The derivative of $V(\mathbf{x})$ must be negative: $\dot{V}(\mathbf{x}) < 0$. In particular, these two conditions are fulfilled for a quadratic Lyapunov function $V = \mathbf{x}^T\,\mathbf{P}\,\mathbf{x}$, with a symmetric, positive definite matrix $\mathbf{P}$.

Let $V(\mathbf{x}) = \mathbf{x}^T\mathbf{P}\mathbf{x}$ be a candidate Lyapunov function for the closed-loop system (4.44). Then it is obvious that $V > 0$ holds, since V is defined as a quadratic form. It must now be shown that the second property $\dot{V} < 0$ is fulfilled.

The derivative of the Lyapunov function is given by

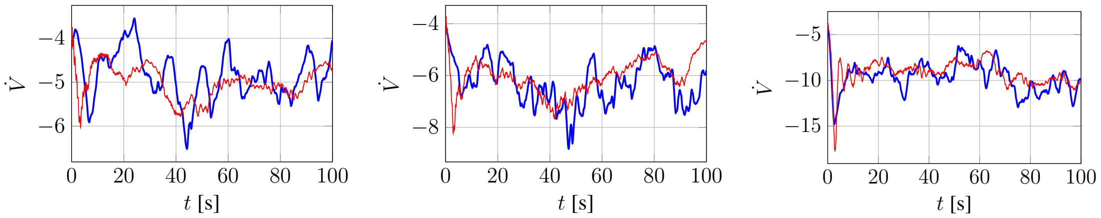
$$\begin{aligned} \dot{V} &= \dot{\mathbf{x}}^T\,\mathbf{P}\,\mathbf{x} + \mathbf{x}^T\,\mathbf{P}\,\dot{\mathbf{x}} \\ &= \sum_i^{N_r} h_i(\beta)\;\mathbf{x}^T\left(\mathbf{P}\,\mathbf{A}_{\mathrm{CL},i} \;+\; \mathbf{A}_{\mathrm{CL},i}^T\,\mathbf{P}\right)\mathbf{x} \;+\; 2\sum_i^{N_r} h_i(\beta)\;\mathbf{x}^T\,\mathbf{P}\,\mathbf{g}_i \;-2\,\mathbf{x}^T\,\mathbf{P}\,\mathbf{B}\,k_I\,x_I \end{aligned} \tag{4.45}$$

If a common symmetric, positive definite matrix $\mathbf{P}$ exists for all linear submodels, such that the Lyapunov condition

$$\mathbf{P}\,\left(\mathbf{A}_i - \mathbf{B}\mathbf{k}_i^T\right) \;+\; \left(\mathbf{A}_i - \mathbf{B}\mathbf{k}_i^T\right)^T\,\mathbf{P} \;<\; 0 \tag{4.46}$$

is fulfilled for all submodels, the first term in (4.45) is negative. The bilinear inequality (4.46) can be formulated as an LMI by adopting a strategy described in [Tanaka and Wang, 2001]. By multiplying the inequality (4.46) by $\mathbf{P}^{-1}$ from both the left and the right side and by defining $\mathbf{X} := \mathbf{P}^{-1}$, (4.46) can be written as

$$\mathbf{A}_i\,\mathbf{X} + \mathbf{X}\,\mathbf{A}_i^T \;-\; \mathbf{B}\,\mathbf{k}_i^T\,\mathbf{X} \;-\; \mathbf{X}\,\mathbf{k}_i\,\mathbf{B}^T \;<\; 0\,. \tag{4.47}$$



(a) 14 m/s (b) 18 m/s (c) 24 m/s

Figure 4.8: Time-derivatives of the Lyapunov function (4.45) obtained from turbulent simulations of the 2-state model (4.34) (blue lines) with three different mean wind speeds covering the whole full load region. Red lines: $\dot{V}$ obtained from FAST simulations for comparison.

By introducing the quantity $\mathbf{M}_i := \mathbf{k}_i^T \mathbf{X}$, (4.47) can be rewritten in LMI form as

$$\mathbf{A}_i \mathbf{X} + \mathbf{X} \mathbf{A}_i^T - \mathbf{B} \mathbf{M}_i - \mathbf{M}_i^T \mathbf{B}^T < 0\,. \tag{4.48}$$

For the present model, (4.48) could be verified by means of the LMI toolbox by Gahinet et al. [Gahinet et al., 1994][3]. For $\mathbf{P}$, the following valid solution was obtained:

$$\mathbf{P} = \begin{pmatrix} 5.8697 & -0.2465 \\ -0.2465 & 0.3080 \end{pmatrix}. \tag{4.49}$$

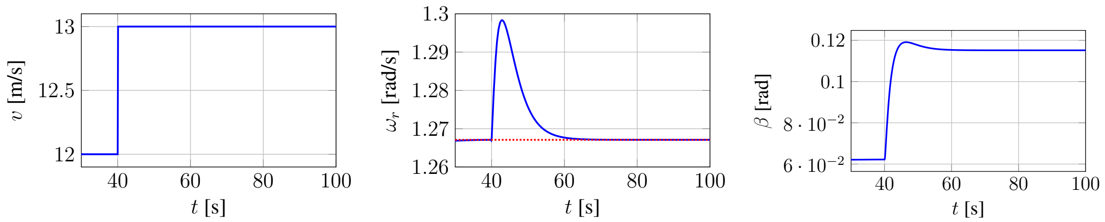
Including the other two unsymmetric terms of (4.45), an analytical estimate to prove the negativity of $\dot{V}$ in general for all possible values of $\tilde{\mathbf{x}}$ could not be obtained.
Instead, the time-series of $\dot{V}$ was calculated for several turbulent simulation runs of the reduced-order control design model (4.34) so as to obtain a reasonable estimate of $\dot{V}$ for the whole full-load region. In Figure 4.8, results of $\dot{V}$ are depicted for three turbulent simulations with mean wind speeds of 14, 18, and 24 m/s. It can bee seen that the derivatives of the Lyapunov function are well below zero in all three cases (for comparison, the derivatives $\dot{V}$ were also calculated from FAST simulation results, were the negativity is also ensured). This it not a formal proof of global stability, but it gives very good theoretical support that the reduced-order model (4.34) controlled with the PDC controller (4.37) is stable in the whole full load region. Since a large number of stationary points was used for the control design and since the rotor speed dynamics of the reduced-order model (4.34) differs from the nonlinear model (4.1) and (even from the FAST model) only in terms of the missing torsional degree of freedom, one can assume to a high level of confidence that the stability of the system (4.1) and even of the FAST model controlled with (4.37) is also ensured.

---

[3] which is now included in the MATLAB® Robust Control Toolbox

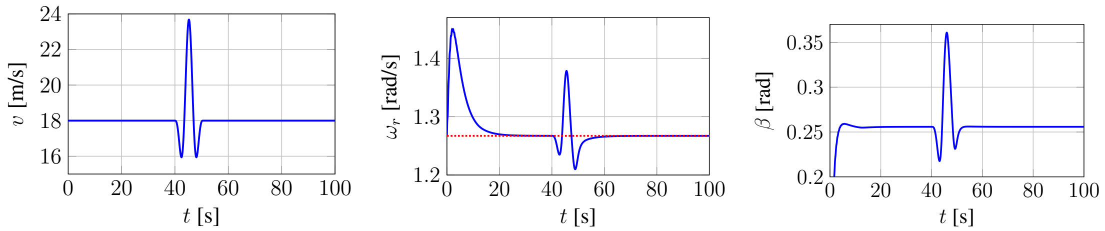
4.1.5 Simulation of the Closed-Loop Behaviour

For the controller validation, the nominal nonlinear model (2.29) was first simulated using a deterministic step of the wind speed (see Figure 4.9) and an IEC wind gust [4] (see Figure 4.10).



(a) Wind speed (b) Rotor speed (c) Pitch angle

Figure 4.9: Validation of the pitch controller (4.37) in a simulation of the nominal model (4.1) with a wind step from 12 to 13 m/s. The red dotted line in subfigure 4.9b indicates the demanded rotor speed.



(a) Wind speed (b) Rotor speed (c) Pitch angle

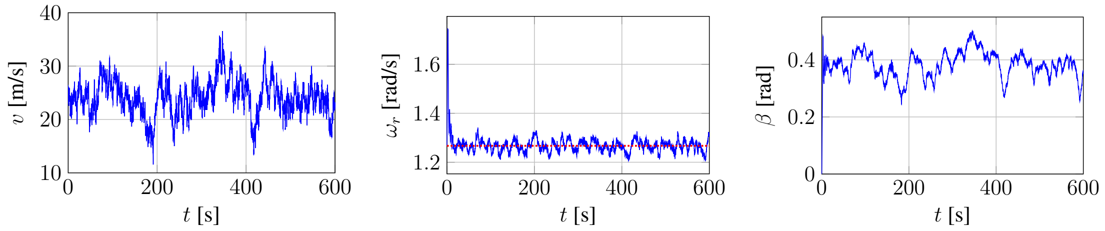
Figure 4.10: Validation of the pitch controller (4.37) in a simulation of the nominal model (4.1) with an IEC wind gust. The red dotted line in subfigure 4.10b indicates the demanded rotor speed.

Subsequently, the controller was integrated into the FAST simulation and simulated using turbulent wind input [5] (see Figure 4.11). A mean wind speed of 24 m/s was chosen, which is at the upper limit of the allowed operating region of a wind turbine. It is visible that except for the initial transient behaviour the rotor speed does not exceed or undershoot the reference speed of $\omega_r = 1.267\,\mathrm{rad/s}$ by more than 6 % during the whole simulation time of 600 s, which is a good value for this value of average wind speed. Whether the controller is suitable for using it with a real wind turbine would have to be validated by analysing the extreme and fatigue loads of the standard load simulation cases, which are defined for example in [GL Renewables Certification, 2010].

---

[4] The IEC gust data was generated using the Software IECWind by NREL [Laino, Laino].

[5] The turbulent wind data was generated using the Software TurbSim by NREL [Kelley, Kelley].



(a) Wind speed (b) Rotor speed (c) Pitch angle

Figure 4.11: Validation of the pitch controller (4.37) in a FAST simulation with turbulent wind (mean wind speed: 24 m/s, turbulence intensity: 14 % (NTM), Kaimal model). The red dotted line in subfigure 4.11b indicates the demanded rotor speed.

4.2 Partial Load Region - Torque Control

The generator torque control is treated differently for the different control regions of the wind turbine (see Figure 4.1). In the lower partial load region, the generator torque is controlled deterministically using a look-up table. When reaching the cut-in rotor speed $\omega_{r,1}$, the torque is ramped up linearly. Starting at a defined rotor speed $\omega_{r,2}$, the torque is adjusted by means of a square law ($T_g \sim \omega_r^2$) to follow the optimal torque, such that the rotor is constantly running around the optimal tip speed ratio. On reaching the reference speed $\omega_{r,\text{SP,trq}}$ for the torque controller (which is slightly below the reference speed $\omega_{r,\text{SP}}$ for the pitch controller), the generator torque is increased in the transition region (upper partial load region) by the controller up to the maximum value of $T_{\text{g,max}}$. The objective behind setting the two controller setpoints to different values is to achieve a decoupling of the controllers. In the whole partial load region, the pitch angle is kept at a constant value of of $\beta = 0°$.

4.2.1 Control Design

The torque controller is a linear controller and is designed around one stationary point ($\omega_{r,\text{SP,trq}} = 1.2\frac{\text{rad}}{\text{s}}$). The optimal tip speed ratio of the 5 MW turbine is $\lambda_{\text{opt}} \approx 7.55$. The wind speed at the stationary point can be obtained as $v_{\text{SP,trq}} = R\frac{\omega_{r,\text{SP,trq}}}{\lambda_{\text{opt}}} \approx 10\frac{\text{m}}{\text{s}}$.

A stiff drivetrain model including only the rotor speed and an integrator state (integral over the controller error) is utilised as a control design model, which means that the state vector is given by $\tilde{\mathbf{x}} = \begin{pmatrix}\omega_r & x_I\end{pmatrix}^T$ and the system dynamics of the (I-augmented) model is given by

$$\dot{\tilde{\mathbf{x}}} = \begin{pmatrix}\dot{\omega}_r \\ \\ \dot{x}_I\end{pmatrix} = \begin{pmatrix}\frac{1}{J_r+J_g}\left(T_a - T_g\right) \\ \\ \omega_r - \omega_{r,\text{SP,trq}}\end{pmatrix} . \tag{4.50}$$

Analogously to the control design model for the pitch controller in section 4.1.4, the linearised system is obtained as

$$\dot{\tilde{\mathbf{x}}} = \tilde{\mathbf{A}}\,\Delta\tilde{\mathbf{x}} + \tilde{\mathbf{B}}\,\Delta u + \tilde{\mathbf{E}}\,\Delta v \tag{4.51}$$

with the linearised system matrices

$$\tilde{\mathbf{A}} = \left.\begin{pmatrix}\frac{1}{J_r+J_g}\frac{\partial T_a}{\partial \omega_r} & 0 \\ \\ 1 & 0\end{pmatrix}\right|_{\omega_{r,\text{SP,trq}}} , \quad \tilde{\mathbf{B}} = \begin{pmatrix}-\frac{1}{J_r+J_g} \\ 0\end{pmatrix} , \quad \tilde{\mathbf{E}} = \left.\begin{pmatrix}\frac{1}{J_r+J_g}\frac{\partial T_a}{\partial v} \\ 0\end{pmatrix}\right|_{v_{\text{SP,trq}}} . \tag{4.52}$$

The control law is given by:

$$\Delta u \;=\; \Delta T_g \;=\; -k_{\omega_r,\text{trq}}\,\Delta\omega_r \;-\; k_{I,\text{trq}}\,x_I\,. \tag{4.53}$$

Altogether, the controlled generator torque is given as the sum

$$T_g \;=\; T_{g,\text{opt,SP,trq}} \;+\; \Delta T_g\,, \tag{4.54}$$

such that the controller (4.53) must be limited between the values 0 and $(T_{g,\text{max}} - T_{g,\text{opt,SP,trq}})$ (see Figure 4.1).

LQR Controller Design

As for the linearised pitch controller design in section 4.1.4, the torque control law is designed using the LQR method, where the weights for the rotor speed and the integrator state were chosen as $Q_1 = 1.57$ and $Q_I = 0.1$. For the final weighting matrix, a normalisation with the square of the largest estimated rotor speed value was included:

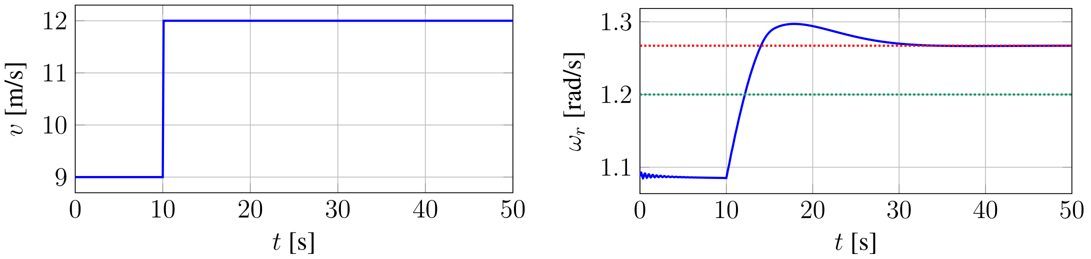
$$\mathbf{Q}_\text{trq} \;=\; \begin{pmatrix} \frac{Q_1}{\omega^2_{r,\text{max}}} & 0 \\ 0 & Q_I \end{pmatrix}\,. \tag{4.55}$$

Since the generator torque must be controlled very fast, the weight for the control input was chosen very small:
$R_\text{trq} = \frac{0.836}{T^2_{g,\text{max}}}$.
Using these weights, the LQR design yields the following controller gains:
$k_{\omega_r,\text{trq}} = -9.66 \cdot 10^6\ \frac{\text{Nm s}}{\text{rad}}$, $k_{I,\text{trq}} = -1.4457 \cdot 10^6\ \frac{\text{Nm}}{\text{rad}}$.

4.2.2 Simulation of the Closed-Loop Behaviour

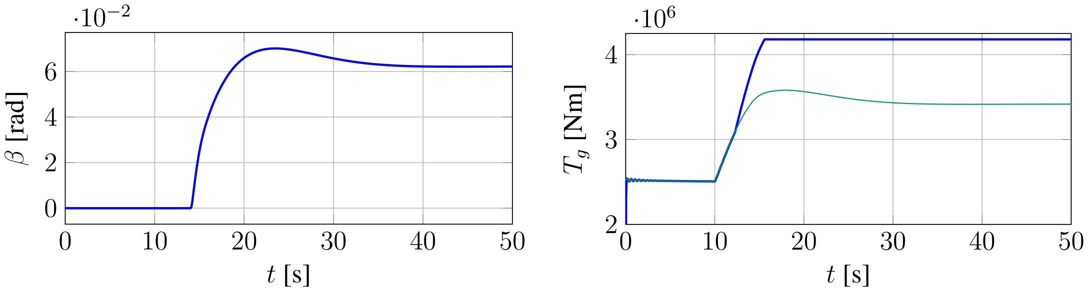
For the controller validation, the nonlinear model (2.29) was first simulated using a deterministic wind speed step from 9 to 12 m/s (see Figure 4.12). It can be seen that the generator torque is initially raised after the wind speed step according to the optimal torque curve. As soon as the reference rotor speed for the controller is exceeded, the torque is raised by the controller very fast, such that the torque leaves the optimal curve.
On exceeding the pitch controller setpoint, the pitch angle is raised by the pitch controller. It is visible that both controllers are simultaneously active for a short period. Before the maximum generator torque is reached, the pitch angle has already been adjusted. This coupling of the two controllers, however, is short and not problematic.

The torque controller (4.54) was also incorporated into the FAST simulation and simulated together with the pitch controller using turbulent wind input (see Figure 4.13). It can be seen that both controllers are fairly well decoupled.



(a) Wind speed

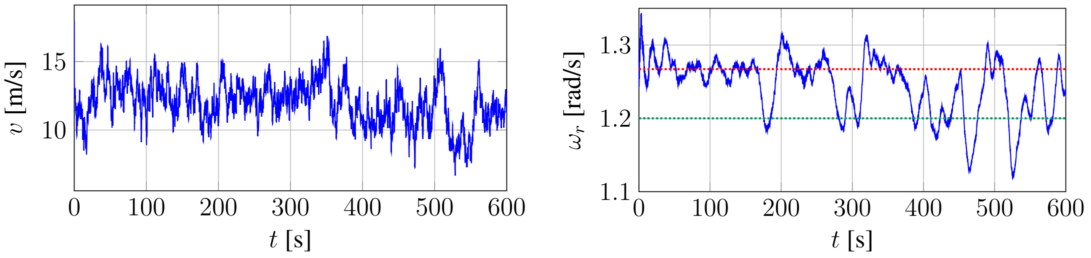
(b) Rotor speed



(c) Pitch angle

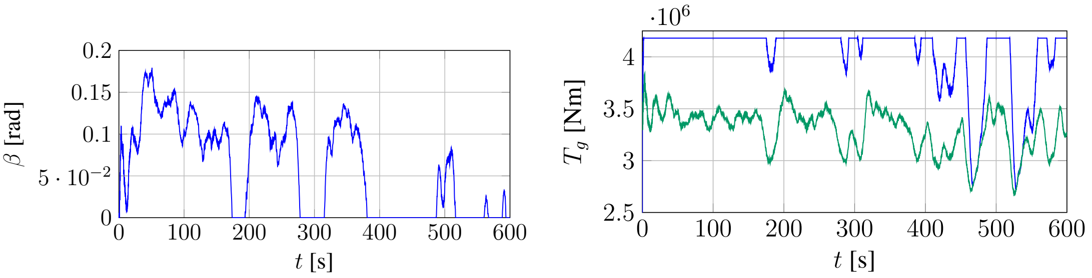
(d) Generator torque

Figure 4.12: Validation of the generator torque controller; simulation of the nominal model (2.29) with a wind step from 9 to 12 m/s; Subfigure 4.12b: green dotted line: torque controller setpoint; red dotted line: pitch controller setpoint; Subfigure 4.12d: green: optimal generator torque according to square law $T_g \sim \omega_r^2$; blue: actually adjusted generator torque.



(a) Wind speed

(b) Rotor speed



(c) Pitch angle

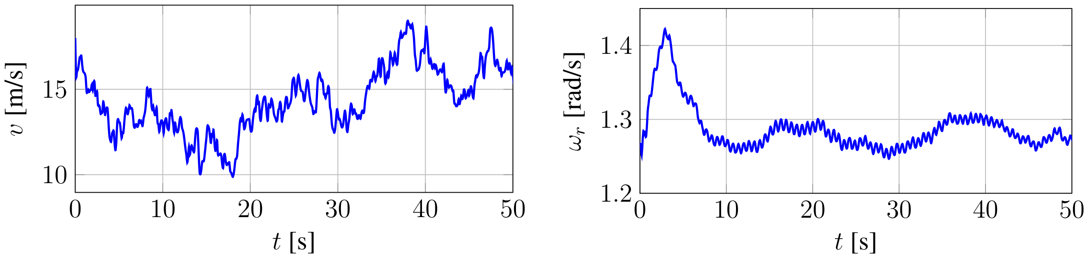
(d) Generator torque

Figure 4.13: Validation of the generator torque controller; FAST simulation with turbulent wind (mean wind speed: 12 m/s, turbulence intensity: 17 % (NTM), Kaimal model); Subfigure 4.13b: green dotted line: torque controller setpoint; red dotted line: pitch controller setpoint; Subfigure 4.13d: green: optimal generator torque; blue: actually adjusted generator torque.

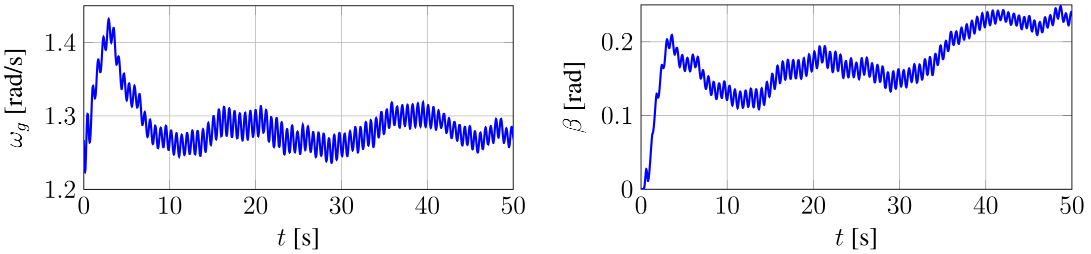
4.3 Filters

In the simulation examples in sections 4.1.5 and 4.2.2, the controlled variable is the rotor speed ω_r. In real wind turbine control systems, however, the generator speed measurement is standardly used as controlled variable, especially in turbines with a gearbox, where the resolution of the conventional rotor speed measurement systems would not be high enough for a precise closed-loop control.

Figure 4.14 shows what happens if the generator speed is used as controlled variable in FAST simulations along with the controllers (4.37) and (4.54).



(a) Wind speed (b) Rotor speed



(c) Generator speed (d) Pitch angle

Figure 4.14: FAST simulation with turbulent wind (mean wind speed: 14 m/s) using the generator speed as controlled variable

4.3.1 Second-Order Filter for Generator Speed

Taking a closer look at Figure 4.14, it can be seen that the generator speed is superimposed with an oscillation with a characteristic frequency of $1.65\,\mathrm{Hz} = 10.364\,\frac{\mathrm{rad}}{\mathrm{s}}$.

This excitation is due to a coupling between the pitch action and the drivetrain, as the stark oscillation is not visible in the partial load region where $\beta = 0°$. It is also not as pronounced when the rotor speed is used as the controlled variable in FAST.
In [Bir and Jonkman, 2007], an aeroelastic stability analysis was conducted for several characteristic modes of the NREL 5 MW turbine. It is interesting that the 2nd regressive flap mode (obtained from a Multi-Blade Coordinate Transformation) at wind speeds around 14 m/s has a damped frequency close to 1.65 Hz, similar to the characteristic frequency seen in Figure 4.14. However, it is stated in [Bir and Jonkman, 2007] that the linearisation scheme in FAST that was employed to analyse the different component modes cannot yet handle active controls, so the exact origin of the excitation is difficult to trace.

Regardless of the exact origin of the oscillation on the generator speed, a possible remedy to reduce its amplitude is a second-order filter for the generator speed, specifically tuned to achieve the highest damping

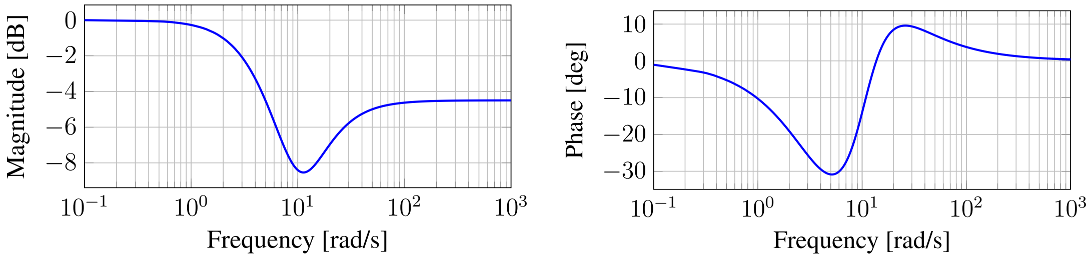


Figure 4.15: Bode plot of 2nd-order filter (4.56) with parameters (4.57)

for one characteristic frequency. The transfer function of a suitable filter is given by [Bossanyi, 2000]

$$G(s) = \frac{\frac{1}{\omega_1^2}\,s^2 + 2\,\frac{\zeta_1}{\omega_1} + 1}{\frac{1}{\omega_2^2}\,s^2 + 2\,\frac{\zeta_2}{\omega_2} + 1}\,, \tag{4.56}$$

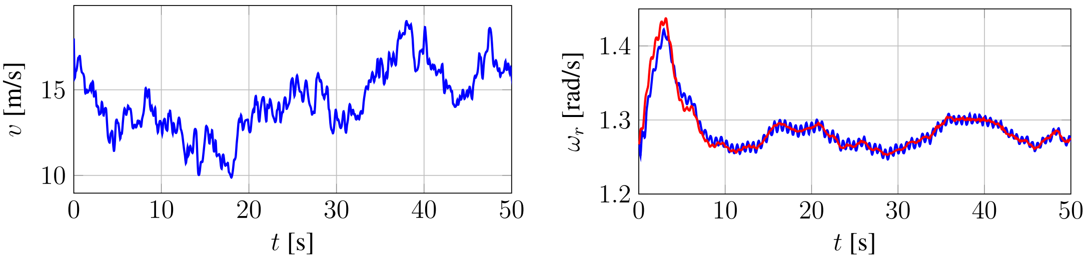
which is a notch filter in the case of $\omega_1 = \omega_2$. In order to achieve additional damping for higher frequencies, the numerator frequency ω_1 should be set equal to the characteristic frequency that is to be damped most and the denominator frequency ω_2 should be chosen such that $\omega_2 < \omega_1$. The damping parameters should be chosen such that $\zeta_1 < \zeta_2$. Otherwise, the characteristic frequency is amplified by the filter instead of being damped.

For the filtering of the generator speed, the following filter parameters were chosen:

$$\omega_1 = 10.364\,\frac{\text{rad}}{\text{s}}\,, \quad \omega_2 = 8\,\frac{\text{rad}}{\text{s}}\,, \quad \zeta_1 = 0.6\,, \quad \zeta_2 = 1.2\,. \tag{4.57}$$

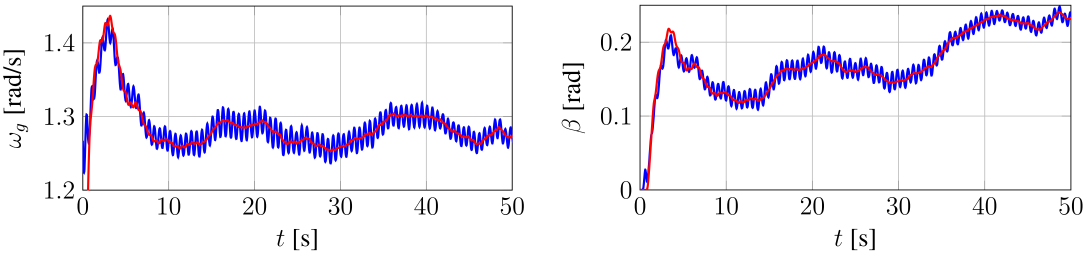
In Figure 4.15, a bode plot of the 2nd-order filter (4.56) with parameters (4.57) is depicted. From the plot of the magnitude in Subfigure (4.15a), it can be seen that the strongest damping is achieved for the characteristic frequency ω_1. Still, all frequencies greater than ω_1 are damped as well. For a real wind turbine control application, this effect can be quite useful, too, as resonances and vibrations on the measured generator speed signal can also occur for higher frequencies. Using a filter structure as in equation (4.56) with two distinct frequencies ω_1 and ω_2 can obviate the need for an additional low-pass filter.

In Figure 4.16, the results of the FAST simulation from Figure 4.14 above are depicted again, together with the results when including the 2nd-order filter for the generator speed (red lines). The damping effect for the different system states is clearly visible.



(a) Wind speed

(b) Rotor speed



(c) Generator speed

(d) Pitch angle

Figure 4.16: FAST simulation with turbulent wind (mean wind speed: 14 m/s) using the generator speed as controlled variable. Blue lines: unfiltered generator speed as controlled variable (compare Figure 4.14); red lines generator speed filtered with 2nd-order filter (4.56) used as controlled variable

5

Observer for Wind Speed Estimation

In this work, the nonlinear wind turbine model (2.29) is the basis for all controller and observer designs, including the TS sliding mode observers used for fault diagnosis (see Chapter 6). In all these cases, the aero maps C_Q and C_T for rotor torque and thrust have to be calculated, which depend on the rotor speed, the pitch angle, and the wind speed. Thus, either a measurement or an estimate of the wind speed is needed. In wind turbines, the wind speed is routinely measured by means of a nacelle anemometer. Being a single-point measurement, it is not a realistic estimate of the 3D wind field that is actually acting on the rotor. Furthermore, the anemometer is situated behind the rotor and thereby influenced by the air currents set in motion by the rotor. The anemometer measurement is thus not ideal for precise measurements of the current effective wind speed acting on the rotor. Exceptions are measurements of average wind speeds ($>=$ 1 min) or measurements signaling maximum allowed wind speeds for the supervisory control system.

In this chapter, a TS observer is designed to obtain an estimate for the effective wind speed. The rotor effective wind speed is the concept of a virtual single point wind speed that yields the same variations in wind torque as the corresponding 3D turbulent wind field [Van der Hooft, E. L. et al., 2003].

Other methods have been applied for wind speed estimation in the literature. See for example [Ma et al., 1995; Ma, 1997], where Kalman filtering, extended Kalman filtering and the Newton-Raphson method are used and compared. Other dedicated algorithms have been applied, too. In [Østergaard et al., 2007], a state-observer for the rotor speed is combined with a PI controller to estimate the aerodynamic rotor torque. The effective wind speed is then reconstructed from the estimated torque signal via inversion of the aerodynamic model. A recent overview and comparison of several techniques for wind speed estimation can be found in [Soltani et al., 2013]. Besides a Kalman and an Extended Kalman Filter, a power Balance Estimator, a DAC Estimator, an Unknown Input Observer (UIO) (see also [Odgaard et al., 2008]) and an Immersion and Invariance Estimator (I&I) are compared. While being able to yield good wind speed estimates, these methods are either rather specialised (DAC, I&I) or have other detriments.

The Kalman Filter (KF) is only applicable to linear state-space models. Thus, estimating the wind speed for a wind turbine using a Kalman filter only yields good results in the region of one operating point of a linearised wind turbine model. A possible remedy is provided by the Extended Kalman Filter (EKF). However, it is not possible to verify formal stability for the error dynamics, since the EKF is an adaptive method. The UIO is based on linear models, too, which means that several observer designs for different local points need to be conducted to achieve a good performance in the whole operating range. As in [Østergaard et al., 2007], the Power Balance Estimator estimates the rotor torque, such that the wind speed needs to be calculated from the implicit nonlinear torque relation, which requires to implement an extra numerical calculation.

For observers in TS structure, the formal stability of the error dynamics can, at least in principle, be shown using linear matrix inequalities (LMI). Furthermore, when the observer model is obtained using the sector-nonlinearity approach, the resulting observer can be used for the whole operating range of a wind turbine. For these reasons, and to achieve a certain level of uniformity with the other TS observer designs in this work, an observer in TS structure is designed here for the estimation of the effective wind speed. The observer was first presented in [Georg, Müller, and Schulte, 2014].

5.1 Observer in Takagi-Sugeno Model Structure

State observers can be used to reconstruct unknown system states from the measurable states. The standard Luenberger observer for linear systems is a state-space model including a feedback of the output error $\mathbf{e}_{\mathbf{y}} = \mathbf{y} - \hat{\mathbf{y}}$, where $\hat{\mathbf{y}}$ is the reconstructed output signal [Luenberger, 1964, 1971]:

$$\dot{\hat{\mathbf{x}}} = \mathbf{A}\,\hat{\mathbf{x}} + \mathbf{B}\,\mathbf{u} + \mathbf{L}\,(\mathbf{y} - \hat{\mathbf{y}})\ , \qquad \hat{\mathbf{y}} = \mathbf{C}\,\hat{\mathbf{x}}\,. \tag{5.1}$$

As the wind turbine model (2.29) is nonlinear, a linear observer like (5.1) cannot be applied in the whole operating range. Therefore, a nonlinear observer in Takagi-Sugeno model structure is used, which is essentially a weighted sum of linear Luenberger observers [Tanaka and Sano, 1994b; Tanaka et al., 1998; Lendek et al., 2010]:

$$\dot{\hat{\mathbf{x}}} = \sum_{i=1}^{N_r} h_i(\mathbf{z}, \hat{\mathbf{z}})\ (\mathbf{A}_i\,\hat{\mathbf{x}} + \mathbf{B}\,\mathbf{u} + \mathbf{L}_i\,(\mathbf{y} - \hat{\mathbf{y}}))\ , \qquad \hat{\mathbf{y}} = \mathbf{C}\,\hat{\mathbf{x}}\,, \tag{5.2}$$

where the premise variables of the membership functions may depend on measurable or unmeasurable states. In the first case, the notation $h_i(\mathbf{z})$ would be used and $h_i(\hat{\mathbf{z}})$ in the latter case.

5.1.1 Observer Model

The state-space model (2.29) is used as a basis for the observer, where either the full model or submodels can be used depending on the desired observer model order. A set of MATLAB® scripts was developed to ease the creation of TS models and the design of the corresponding TS observers for the possible submodels.

In this work, only the rotational and torsional degrees of freedom are incorporated into the observer model but no tower and blade dynamics. This model configuration for the observer yields reasonable results while requiring relatively few measurement signals.

Wind Model

In order to estimate the wind speed v with a state observer, v is included into the system state vector $\mathbf{x}$ and a dynamic wind model is added to the system equations. The first-order delay model from [Ekelund, 1994] is used, modified by the mean value $\bar{v}$ of the wind speed, but without a white noise term:

$$\dot{v} = -\frac{1}{\tau_v}\,(v - \bar{v})\ , \tag{5.3}$$

where the time constant is estimated as $\tau_v = 4\,\mathrm{s}$. When applying the observer on a real wind turbine, the mean wind speed $\bar{v}$ could be calculated over an appropriate time period (e.g. 10 min) from the anemometer wind measurement, which would be sufficient for this purpose.

Since only the rotational and torsional degrees of freedom plus the estimated wind speed are considered for the observer model, the corresponding estimated state vector is

$\hat{\mathbf{x}} = \begin{pmatrix} \hat{\theta}_s & \hat{\omega}_r & \hat{\omega}_g & \hat{v} \end{pmatrix}^T$. Since the first order pitch dynamics adds no information as to the reconstruction of the unknown states, it is not considered in the observer model. This implies that the demanded pitch angle β_d is not included in the input vector, because there is no linear dependence on β_d but only a nonlinear dependence in $C_Q\left(\hat{\lambda}, \beta_d\right)$. The mean wind speed $\bar{v}$ can be included in the input vector: $\mathbf{u} = (T_g \quad \bar{v})^T$. The following states are assumed as measurable: θ_s, ω_r, ω_g. Measuring the rotor and generator speed signals is routinely done in wind turbines. For a real application of the observer, the rotor speed signal would have to measured with high resolution and both speed signals might need to be filtered (see section 4.3). It is only an assumption at this stage that the torsion angle is measurable. However, it should be possible to measure the rotation angles before and after the coupling between gearbox and generator (using for example absolute encoders) and thereby the torsion angle $\theta_s = \theta_r - \theta_g$, where the gearbox ratio can be taken into account simply as a factor.

Complete Observer Model

From the system (2.29) of nonlinear state-space equations for the wind turbine model, it is straightforward to obtain the nonlinear system matrix, the input matrix and the output matrix for the observer model:

$$\mathbf{A}\left(\hat{\mathbf{x}}, \beta_d\right) = \begin{pmatrix} 0 & 1 & -1 & 0 \\ -\frac{k_S}{J_r} & -\frac{d_S}{J_r} & \frac{d_S}{J_r} & f\left(\hat{\mathbf{x}}, \beta_d\right) \\ \frac{k_S}{J_g} & \frac{d_S}{J_g} & -\frac{d_S}{J_g} & 0 \\ 0 & 0 & 0 & -\frac{1}{\tau_v} \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ -\frac{1}{J_g} & 0 \\ 0 & \frac{1}{\tau_v} \end{pmatrix}, \quad \mathbf{C} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}, \tag{5.4}$$

with the nonlinear function $f\left(\hat{\mathbf{x}}, \beta_d\right) = \frac{1}{2\,J_r}\,\rho\,\pi\,R^3\,\hat{v}\,C_Q\left(\hat{\lambda}, \beta_d\right)$, which assumes the following estimated minimum and maximum values: $\underline{f} \,=\, 1.241 \cdot 10^{-5}\,\frac{1}{\mathrm{m\,s}}$, $\quad \overline{f} \,=\, 0.056\,\frac{1}{\mathrm{m\,s}}$.

Employing the same sector-nonlinearity procedure as in chapter 3, the observer model can be obtained in TS structure:

$$\dot{\hat{\mathbf{x}}} = \sum_{i=1}^{N_r=2} h_i(\hat{\mathbf{z}})\,\left(\mathbf{A}_i\,\hat{\mathbf{x}} + \mathbf{B}\,\mathbf{u} + \mathbf{L}_i\,(\mathbf{y} - \hat{\mathbf{y}})\right), \qquad \hat{\mathbf{y}} = \mathbf{C}\,\hat{\mathbf{x}}, \tag{5.5}$$

where the premise variable $\hat{\mathbf{z}}$ now depends on the reconstructed states: $\hat{\mathbf{z}} = (\hat{\omega}_r \;\; \hat{v} \;\; \beta_d)^T$.

5.1.2 Observer Gains and Stability

A common means to derive gain matrices for observers in TS structure is by applying the direct method of Lyapunov in form of linear matrix inequalities (LMI) [Tanaka and Wang, 1997; Lendek et al., 2010].

In general, the global asymptotic stability of a nonlinear system $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x})$ is guaranteed if there exists a Lyapunov function $V(\mathbf{x})$ satisfying the conditions $V(\mathbf{x}) > 0$ and $\dot{V}(\mathbf{x}) < 0$ for all trajectories $\mathbf{x} \neq \mathbf{0}$. In particular, the system is stable if it is quadratically stable, i.e., if a quadratic Lyapunov function $V = \mathbf{x}^T\mathbf{P}\mathbf{x}$, with a symmetric, positive definite matrix $\mathbf{P}$, exists.
In that case, for an autonomous TS system without an external input ($\dot{\mathbf{x}} = \sum_{i=1}^{N_r} h_i(\mathbf{z})\, \mathbf{A}_i$), the condition $\dot{V}(\mathbf{x}) < 0$ is equivalent to $\dot{V} = \dot{\mathbf{x}}^T\,\mathbf{P}\,\mathbf{x} + \mathbf{x}^T\,\mathbf{P}\,\dot{\mathbf{x}} = \mathbf{x}^T \left(\sum_{i=1}^{N_r} h_i(\mathbf{z}) \left(\mathbf{A}_i^T\,\mathbf{P} + \mathbf{P}\,\mathbf{A}_i\right)\right) \mathbf{x} < 0$. Since this condition must hold for all $\mathbf{x}$, the autonomous TS system is stable if there exists a common symmetric, positive definite matrix $\mathbf{P}$, such that ([Tanaka and Sugeno, 1992; Wang et al., 1996])

$$\mathbf{A}_i^T\,\mathbf{P} + \mathbf{P}\,\mathbf{A}_i < 0 \qquad (i \in \{1, \ldots, N_r\}) \,. \tag{5.6}$$

For the TS observer (5.5), where the membership functions depend on unmeasurable states ($h_i = h_i(\hat{\mathbf{z}})$), a modified form of the stability condition (5.6) with an additional LMI can be used to guarantee the stability of the error dynamics of the observer system [Bergsten et al., 2001]:

$$\mathbf{P}\,(\mathbf{A}_i - \mathbf{L}_i\mathbf{C}) + (\mathbf{A}_i - \mathbf{L}_i\mathbf{C})^T\,\mathbf{P} \leq -\mathbf{Q}\,, \qquad \begin{pmatrix} \mathbf{Q} - \mu^2 I & \mathbf{P} \\ \mathbf{P} & \mathbf{I} \end{pmatrix} > 0\,, \tag{5.7}$$

where $\mathbf{Q}$ is a symmetric, positive definite matrix and $\mu > 0$ is a known constant satisfying $\Delta(\mathbf{z}, \hat{\mathbf{z}}) \leq \mu\|\mathbf{e}\|$, with $\mathbf{e} = \|\mathbf{x} - \hat{\mathbf{x}}\|$ and $\Delta(\mathbf{z}, \hat{\mathbf{z}}) = \left\|\sum_{i=1}^{N_r} (h_i(\mathbf{z}) - h_i(\hat{\mathbf{z}}))\,(\mathbf{A}_i\,\mathbf{x} + \mathbf{B}\,\mathbf{u})\right\|$. The first inequality of (5.7) is not an LMI but can be recast into an LMI form by introducing $\mathbf{N}_i := \mathbf{P}\mathbf{L}_i$ [Tanaka and Wang, 2001]. As condition (5.7) concerns quadratic stability, it is only a sufficient stability condition, i.e., if it is not fulfilled, no formal statement can be made about the stability or instability of the considered system [Lendek et al., 2010].

Optimal LMI Observer Design

Condition (5.7) was first used to calculate the observer gains. However, this observer hardly had any modifying effect on the wind speed compared to the mere wind model (5.3). A possible remedy is to modify the gain matrices with a weighting matrix, such that the gains influencing the wind speed $\hat{v}$ are increased. A more systematic way is to make use of optimal fuzzy control concepts, where weighting matrices for the system states/outputs and inputs as well as a quadratic cost function can be included into the LMIs [Tanaka et al., 1998].
For the observer design in this work, theorem 5 from [Tanaka et al., 1998], which is applicable for controller design, was modified to be used for the dual TS-systems $\left(\mathbf{A}_i^T, \mathbf{C}^T\right)$. The observer gain matrices $\mathbf{L}_i$ are then obtained from the resulting gain matrices $\mathbf{K}_i$ as $\mathbf{L}_i = \mathbf{K}_i^T$.

Theorem 5 from [Tanaka et al., 1998], which can also be found in [Tanaka and Wang, 2001], is:

$$\underset{\mathbf{X},\mathbf{M}_1,\dots,\mathbf{M}_r,\mathbf{Y}_0}{\text{minimise}} \quad \lambda \quad \text{subject to} \tag{5.8}$$

$$\tag{5.9}$$

$$\begin{aligned}
\mathbf{X} &> 0\,, && (5.10)\\
\mathbf{Y}_0 &\geq 0\,, && (5.11)\\
\begin{pmatrix} \lambda & \mathbf{x}^T(0) \\ \mathbf{x}(0) & \mathbf{X} \end{pmatrix} &> 0\,, && (5.12)\\
\mathbf{U}_{ii} + (s-1)\,\mathbf{Y}_3 &< 0\,, && (5.13)\\
\mathbf{V}_{ij} - 2\,\mathbf{Y}_4 &< 0\,, && (5.14)
\end{aligned}$$

where $s > 1$ and

$$\mathbf{U}_{ii} = \begin{pmatrix} \left(\mathbf{X}\mathbf{A}_i^T + \mathbf{A}_i\mathbf{X} - \mathbf{B}_i\mathbf{M}_i - \mathbf{M}_i^T\mathbf{B}_i^T\right) & \mathbf{X}\mathbf{C}_i^T & -\mathbf{M}_i^T \\ \mathbf{C}_i\mathbf{X} & -\mathbf{W}^{-1} & 0 \\ -\mathbf{M}_i & \mathbf{0} & -\mathbf{R}^{-1} \end{pmatrix}, \tag{5.15}$$

$$\mathbf{V}_{ij} = \begin{pmatrix} \begin{matrix}\Big(\mathbf{X}\mathbf{A}_i^T + \mathbf{A}_i\mathbf{X} - \mathbf{B}_i\mathbf{M}_j - \mathbf{M}_j^T\mathbf{B}_i^T \\ +\mathbf{X}\mathbf{A}_j^T + \mathbf{A}_j\mathbf{X} - \mathbf{B}_j\mathbf{M}_i - \mathbf{M}_i^T\mathbf{B}_j^T\Big)\end{matrix} & \mathbf{X}\mathbf{C}_i^T & -\mathbf{M}_j^T & \mathbf{X}\mathbf{C}_j^T & -\mathbf{M}_i^T \\ \mathbf{C}_i\mathbf{X} & -\mathbf{W}^{-1} & 0 & 0 & 0 \\ -\mathbf{M}_j & \mathbf{0} & -\mathbf{R}^{-1} & 0 & 0 \\ \mathbf{C}_j\mathbf{X} & 0 & 0 & -\mathbf{W}^{-1} & 0 \\ -\mathbf{M}_i & 0 & 0 & 0 & -\mathbf{R}^{-1} \end{pmatrix} \tag{5.16}$$

$$\begin{aligned}
\mathbf{Y}_3 &= \text{block-diag}\left(\mathbf{Y}_0 \quad \mathbf{0} \quad \mathbf{0}\right)\,, && (5.17)\\
\mathbf{Y}_4 &= \text{block-diag}\left(\mathbf{Y}_0 \quad \mathbf{0} \quad \mathbf{0} \quad \mathbf{0} \quad \mathbf{0}\right)\,. && (5.18)
\end{aligned}$$

Using LMI conditions (5.10)-(5.14), a quadratic cost function $J = \int\limits_0^\infty \left(\mathbf{y}^T\mathbf{W}\mathbf{y} + \mathbf{u}^T\mathbf{R}\mathbf{u}\right) dt$ is implicitly included. If all system states $\mathbf{x}$, and not just the outputs $\mathbf{y}$ shall be included in the design, the matrix $\mathbf{C}$ can be replaced with the unit matrix $\mathbf{I}$ in Equations (5.15) and (5.16).

Within this scheme, the controller matrices can be obtained as

$$\mathbf{K}_i = \mathbf{M}_i\mathbf{X}^{-1}\,. \tag{5.19}$$

For the observer design, the dual system $\left(\mathbf{A}_i^T\,,\mathbf{C}_i^T\right)$ is used instead. In this case, the outputs assume the role of the inputs, such that the cost function is changed to $J = \int\limits_0^\infty \left(\mathbf{x}^T\mathbf{W}\mathbf{x} + \mathbf{y}^T\mathbf{R}\mathbf{y}\right) dt$ and Equations (5.15) and (5.16) are changed to

$$\mathbf{U}_{ii} = \begin{pmatrix} \left(\mathbf{X}\mathbf{A}_i + \mathbf{A}_i^T\mathbf{X} - \mathbf{C}_i^T\mathbf{M}_i - \mathbf{M}_i^T\mathbf{C}_i\right) & \mathbf{X} & -\mathbf{M}_i^T \\ \mathbf{X} & -\mathbf{W}^{-1} & 0 \\ -\mathbf{M}_i & \mathbf{0} & -\mathbf{R}^{-1} \end{pmatrix}, \tag{5.20}$$

$$\mathbf{V}_{ij} = \begin{pmatrix} \begin{matrix}\Big(\mathbf{X}\mathbf{A}_i + \mathbf{A}_i^T\mathbf{X} - \mathbf{C}_i^T\mathbf{M}_j - \mathbf{M}_j^T\mathbf{C}_i \\ +\mathbf{X}\mathbf{A}_j + \mathbf{A}_j^T\mathbf{X} - \mathbf{C}_j^T\mathbf{M}_i - \mathbf{M}_i^T\mathbf{C}_j\Big)\end{matrix} & \mathbf{X} & -\mathbf{M}_j^T & \mathbf{X} & -\mathbf{M}_i^T \\ \mathbf{X} & -\mathbf{W}^{-1} & 0 & 0 & 0 \\ -\mathbf{M}_j & \mathbf{0} & -\mathbf{R}^{-1} & 0 & 0 \\ \mathbf{X} & 0 & 0 & -\mathbf{W}^{-1} & 0 \\ -\mathbf{M}_i & 0 & 0 & 0 & -\mathbf{R}^{-1} \end{pmatrix} \tag{5.21}$$

In the LMI (5.12), the state vector has to be replaced by the error vector $\mathbf{e} = \mathbf{x} - \hat{\mathbf{x}}$, such that

$$\begin{pmatrix} \lambda & \mathbf{e}^T(0) \\ \mathbf{e}(0) & \mathbf{X} \end{pmatrix} > 0\,. \tag{5.22}$$

Analogous to Equation (5.19), but considering the dual system $\left(\mathbf{A}_i^T, \mathbf{C}_i^T\right)$, the observer gain matrices can be obtained as

$$\mathbf{L}_i = \left(\mathbf{M}_i\mathbf{X}^{-1}\right)^T\,. \tag{5.23}$$

The following weighting matrices ($\mathbf{W}$ for the system states and $\mathbf{R}$ for the system outputs) were used for the optimal LMI observer design:
$\mathbf{W} = \operatorname{diag}\left(\frac{W_1}{\theta^2_{s,\max}},\ \frac{W_2}{\omega_{r^2,\max}},\ \frac{W_3}{\omega^2_{g,\max}},\ \frac{W_4}{v^2_{\max}}\right)$, $\mathbf{R} = \operatorname{diag}\left(\frac{R_1}{\theta^2_{s,\max}},\ \frac{R_2}{\omega^2_{r,\max}},\ \frac{R_3}{\omega^2_{g,\max}}\right)$,
with $W_1 = 0.25$, $W_2 = 15.708$, $W_3 = 1.5708$, $W_4 = 60 \cdot 10^7$, $R_1 = 0.05$, $R_2 = 0.1571$, $R_3 = 1.5708$ and the estimated maximum values
$\theta_{s,\max} = 0.01\,\mathrm{rad}$, $\omega_{r,\max} = \omega_{g,\max} = 15 \cdot \frac{\pi}{30}\,\frac{\mathrm{rad}}{\mathrm{s}}$, $v_{\max} = 60\,\frac{\mathrm{m}}{\mathrm{s}}$ to normalise the chosen weights.

For the optimal LMI design procedure, the initial observer error is needed, which was set to $\mathbf{e}_0 = (0\,0\,0\,0)^T$. This is of course an idealisation. However, as can be seen from the simulation results in Figures 5.1 and 5.2, the observer is stable also for $\|\mathbf{e}_0\| > 0$, even when the initial error components are significantly larger than 0. [1]
The following observer gain matrices were obtained:

$$L_1 = \begin{pmatrix} 0.147 & -176.5 & 143.6 \\ -0.022 & 133 & -28.6 \\ 0.183 & -286.1 & 303.2 \\ 0.08 & 6698.1 & 741.2 \end{pmatrix}, \quad L_2 = \begin{pmatrix} 0.147 & -176.5 & 143.6 \\ -0.022 & 133 & -28.6 \\ 0.183 & -286.1 & 303.2 \\ 0.08 & 6698.1 & 741.2 \end{pmatrix} \tag{5.24}$$

$\mathbf{L}_1$ and $\mathbf{L}_2$, displayed here with rounded values, are not equal but differ by less than 0.1 %.

The formal stability of the error dynamics could not be verified with condition (5.7), which is a conservative condition due to the assumption of un unstructured uncertainty [Lendek et al., 2010]. However, the observer shows a stable behaviour in FAST simulations, even for large initial observer errors (see Section 5.2).

---

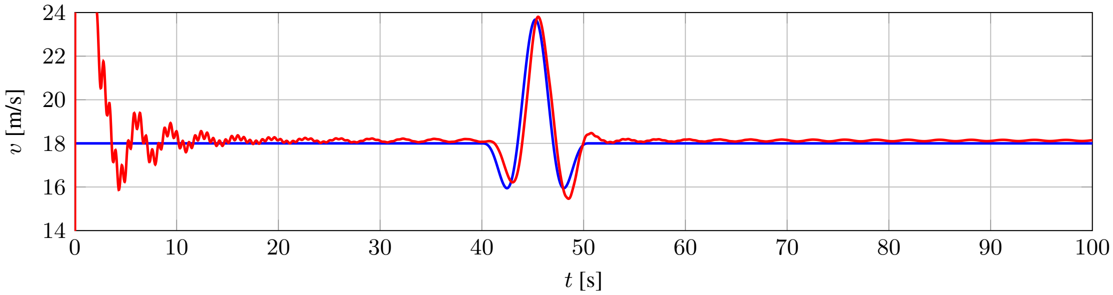
[1] An initial observer error vector of $\mathbf{e}_0 = (0.01\ 0.01\ 0.01\ 0)^T$ in the design procedure also yields a stable observer with a similar behaviour. However, including even larger errors in the design process leads to infeasible LMIs and observers without successful wind speed reconstruction. The design parameters were thus chosen based on pragmatic considerations.

5.2 Simulation Results

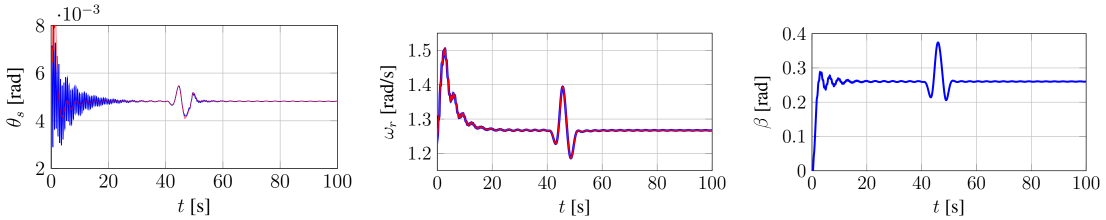
For the observer validation, FAST simulations were carried out using both IEC wind gusts and turbulent wind input.

Simulation with IEC Wind Gust

As a first simulation test, an IEC wind gust with a mean wind speed of 18 m/s was simulated. Results are shown in Figure 5.1. After the transient, the observer shows a stable behaviour and the wind gust is reconstructed with a phase offset of $\approx 0.4 - 0.5$ s. Almost perfect reconstruction is achieved for the rotor speed ω_r.



(a) Wind speed



(b) Torsion angle (c) Rotor speed (d) Pitch angle

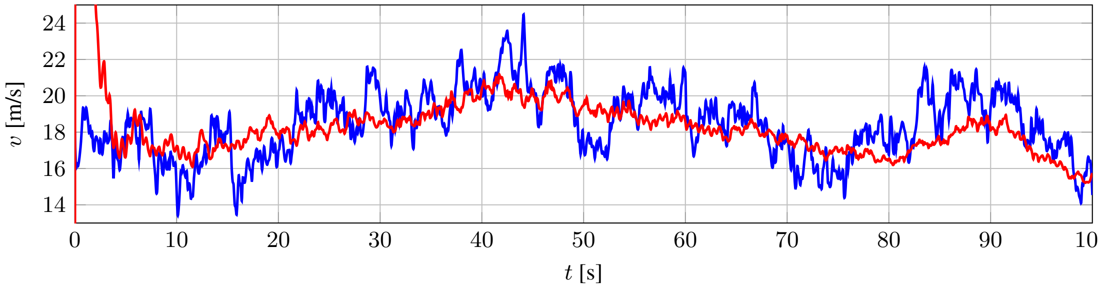
Figure 5.1: FAST Simulation with IEC wind gust. Blue: states from FAST output; red: estimated states; Initial values: $\theta_{s,0} = 0\,\text{rad}$, $\hat{\theta}_{s,0} = 0.1\,\text{rad}$, $\omega_{r,0} = \omega_{g,0} = 1.267\,\frac{\text{rad}}{\text{s}}$, $\hat{\omega}_{r,0} = \hat{\omega}_{g,0} = 0\,\frac{\text{rad}}{\text{s}}$, $v_0 = 18\,\frac{\text{m}}{\text{s}}$, $\hat{v}_0 = 1\,\frac{\text{m}}{\text{s}}$. The torsion angle is not directly available from the FAST outputs and was obtained by integrating the speed error signal from the FAST outputs of rotor and generator speed (corrected by the gear ratio). The pitch angle is only shown for reference.

Simulation with Turbulent Wind

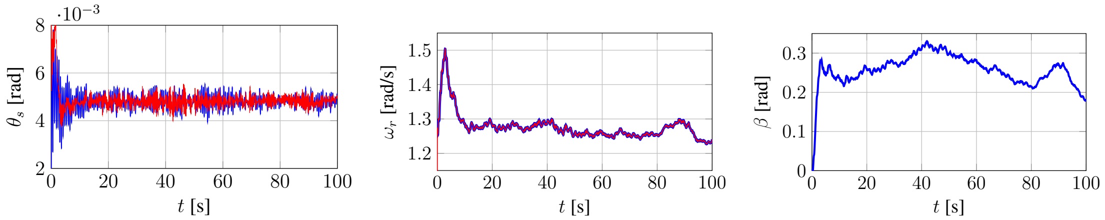
A second simulation run in FAST was conducted using a 3D turbulent wind field with a mean wind speed of 18 m/s. Results are shown in Figure 5.2.

When interpreting Figure 5.2a, it is important to remember that the observer estimates the rotor effective wind speed, i.e. a virtual single point wind speed that causes the same variations in wind torque as the corresponding 3D turbulent wind field [Van der Hooft, E. L. et al., 2003]. Although the calculations in FAST are based on the 3D wind field, the wind speed output from FAST (blue curve in Figure 5.2a) shows the nominal downwind component of the hub-height wind speed, so the two wind speed curves in Figure 5.2a are not directly comparable. The FAST wind speed output is shown to give an idea of the observer performance.

In case the observer shall be tested on a real turbine, the estimated effective wind speed would not be directly comparable to single point measurements neither on the nacelle nor on meteorological towers. It



(a) Wind speed



(b) Torsion angle (c) Rotor speed (d) Pitch angle

Figure 5.2: FAST simulation results with turbulent wind input (mean wind speed: 18 m/s, turbulence intensity: 15 % (NTM), Kaimal model). Blue: states from FAST simulation; red: estimated states; initial values as in Figure 5.1. The blue wind speed signal from FAST is the nominal downwind component of the hub-height wind speed, not the rotor effective wind speed.

would be interesting to compare the estimated wind speed to LIDAR measurements of the wind field measured in front of the rotor. LIDAR measurement techniques will become important for the next generation of wind turbines (see for example [**?**Mikkelsen et al., 2013]).

5.2.1 Further Reduced Observer Model

For the observer model (5.4) in section 5.1.1, the torsional degree of freedom was included. As a consequence, the torsion angle must be measurable for the observer design. If a reliable measurement of the torsion angle cannot be achieved, it is still possible to design an observer which is able to reconstruct the effective wind speed. In this case, only the rotor speed ω_r and the wind speed v are included as states in the observer model, which is then given by the following system and input matrices:

$$\mathbf{A}\left(\hat{\mathbf{x}}\right) = \begin{pmatrix} 0 & f\left(\hat{\mathbf{x}},\, \beta_d\right) \\ 0 & -\frac{1}{\tau_v} \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} -\frac{1}{J_r+J_g} & 0 \\ 0 & \frac{1}{\tau_v} \end{pmatrix}, \tag{5.25}$$

with the nonlinear function $f\left(\hat{\mathbf{x}},\, \beta_d\right) = \frac{1}{2\,(J_r+J_g)}\, \rho\, \pi\, R^3\, \hat{v}\, C_Q\left(\hat{\lambda},\, \beta_d\right)$, which assumes the following estimated minimum and maximum values: $\underline{f} = 1.1 \cdot 10^{-5}\, \frac{1}{\mathrm{m\,s}}$, $\overline{f} = 0.0495\, \frac{1}{\mathrm{m\,s}}$.

Using the same weights as in section 5.1.2, the following observer gains were obtained:

$$L_1 = \begin{pmatrix} 93.1 \\ 6.73 \cdot 10^3 \end{pmatrix}, \qquad L_2 = \begin{pmatrix} 93.1 \\ 6.73 \cdot 10^3 \end{pmatrix}, \tag{5.26}$$

In Figures 5.3 and 5.4, the reconstructed wind speeds using both the observer model (5.4) including the torsional degree of freedom and the reduced model (5.25) are depicted. It can be seen that there are only minor differences in the quality of the reconstructed signals. Here, the mean wind speed was set to 12 m/s instead of 18 m/s. When comparing the results to those in Figures 5.1 and 5.2, one can see that the quality of wind speed estimation is slightly reduced compared to the simulations with a mean wind speed of 18 m/s. In the case of the wind gust, the offset between the true and reconstructed wind speed is slighty increased to ≈ 0.7 s. Still, given that the same observer was used and no special observer design is needed for lower wind speeds, the estimation quality is still reasonable.

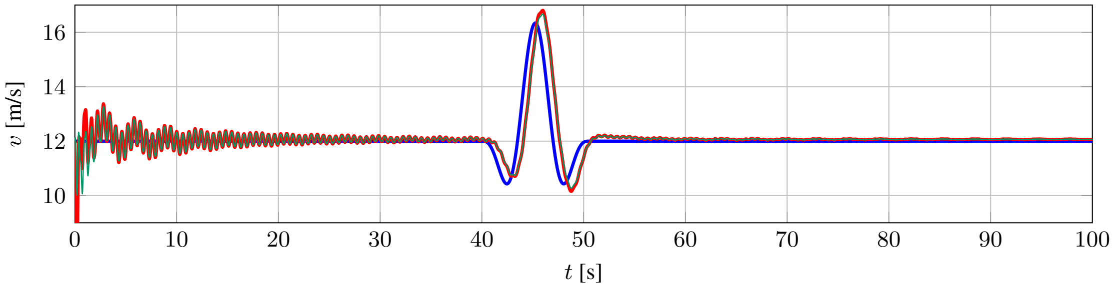


Figure 5.3: FAST simulation with IEC wind gust (mean wind speed: 12 m/s, turbulence intensity: 17 % (NTM), Kaimal model). blue lines: FAST wind speed signal. red lines: reconstructed wind speed using observer model (5.4) with torsional degree of freedom. green lines: reconstructed wind speed using the reduced observer model (5.25) without torsional degree of freedom.

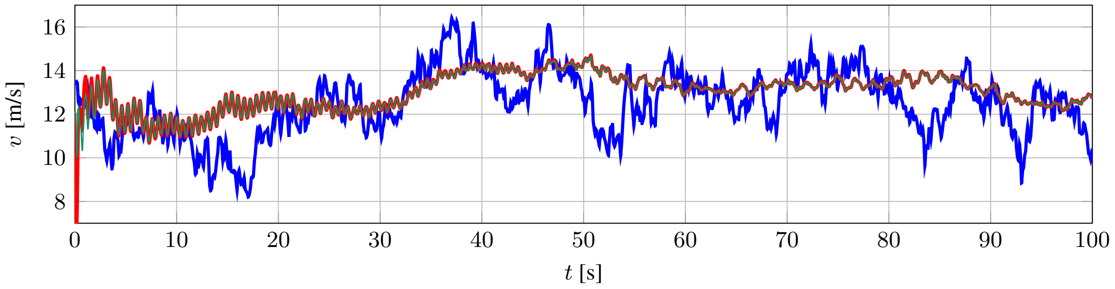


Figure 5.4: FAST simulation with turbulent wind input (mean wind speed: 12 m/s, turbulence intensity: 17 % (NTM), Kaimal model). blue lines: FAST wind speed signal. red lines: reconstructed wind speed using observer model (5.4) with torsional degree of freedom. green lines: reconstructed wind speed using the reduced observer model (5.25) without torsional degree of freedom.

Part II

Fault Diagnosis and Fault-Tolerant Control

6

Fault Reconstruction

In this chapter, actuator and sensor faults in dynamic wind turbine simulations are reconstructed by means of a Takagi-Sugeno sliding mode observer, which is a nonlinear extension of the Edwards-Spurgeon observer for fault reconstruction [Edwards and Spurgeon, 1994, 1998; Edwards et al., 2000].

6.1 Takagi-Sugeno Sliding Mode Observer

Edwards and Spurgeon have introduced a sliding mode observer concept where actuator and sensor faults can be directly reconstructed by evaluating the so-called equivalent output injection signal [Edwards and Spurgeon, 1998; Edwards et al., 2000]. This sliding mode observer is best suited for linear systems, such that the nonlinear switching term solely accounts for the disturbances and faults. For a highly nonlinear model like a wind turbine, a structure like the Edwards-Spurgeon observer is not ideal, as the sliding term would then have to capture both the nonlinear plant dynamics and the influence of the faults.

For this reason, a nonlinear extension of the Edwards-Spurgeon observer was introduced in [Gerland et al., 2010b,a], based on a nonlinear TS structure. In this way, the nonlinear plant dynamics can be fully dealt with using the TS structure, such that the switching term is solely responsibly for the faults.

In the literature, other sliding mode observers for nonlinear TS systems have been presented, for example in [Bergsten et al., 2001, 2002] and [Akhenak et al., 2007, 2008]. However, in [Bergsten et al., 2001, 2002], the observers are not used for FDI. In [Akhenak et al., 2008], fault detection and isolation is implemented by analysing the residual signals of the TS sliding mode observer. The advantage of the Takagi-Sugeno sliding mode observer (TS SMO) introduced in [Gerland et al., 2010b,a] is that the direct fault reconstruction methods available for the Edwards-Spurgeon observer can be employed in a similar way.

In this work, as in [Gerland et al., 2010b], the TS SMO design is based on a nonlinear TS system including disturbances, actuator faults, and sensor faults:

$$
\begin{aligned}
\dot{\mathbf{x}} &= \sum_{i=1}^{N_r} h_i(\mathbf{z}) \left(\mathbf{A}_i \mathbf{x} + \mathbf{B}_i \mathbf{u} + \mathbf{D}_i \boldsymbol{\xi} + \mathbf{F}_i \mathbf{f}_a \right) , \\
\tilde{\mathbf{y}} &= \mathbf{y} + \mathbf{f}_s = \mathbf{C}\mathbf{x} + \mathbf{f}_s ,
\end{aligned}
\tag{6.1}
$$

where $\boldsymbol{\xi}$ denotes the vector of disturbances or unmodelled dynamics, $\mathbf{f}_a$ denotes the vector of actuator faults and $\mathbf{f}_s$ the vector of sensor faults.

6.1.1 TS Sliding Mode Observer in Transformed Form

The TS SMO is designed in a transformed system, where the measurable system states $\mathbf{y}$ and the non-measurable system states $\mathbf{x}_1$ are separated. This separation is achieved by introducing a transformation $\mathbf{T}_c = \left[\mathbf{N}_c \ \mathbf{C}^T\right]^T$, where $\mathbf{N}_c$ denotes the null-space of $\mathbf{C}$. With a series of transformations

$$\mathbf{T}_i \; = \; \mathbf{T}_{L,i}\, \mathbf{T}_{a,i}\, \mathbf{T}_{b,i}\, \mathbf{T}_c \, ,^1 \tag{6.2}$$

the TS system is brought into a structure where the uncertainties and faults only act on the measurable system states [Gerland et al., 2010b]:

$$\begin{aligned} \dot{\mathbf{x}}_1 \; &= \; \sum_{i=1}^{N_r} h_i\left(\mathbf{z}\right) \left(\boldsymbol{\mathcal{A}}_{11,i}\, \mathbf{x}_1 \, + \, \boldsymbol{\mathcal{A}}_{12,i}\, \mathbf{y} \, + \boldsymbol{\mathcal{B}}_{1,i}\, \mathbf{u}\right) \, , && (6.3)\\ \dot{\mathbf{y}} \; &= \; \sum_{i=1}^{N_r} h_i\left(\mathbf{z}\right) \left(\boldsymbol{\mathcal{A}}_{21,i}\, \mathbf{x}_1 \, + \, \boldsymbol{\mathcal{A}}_{22,i}\, \mathbf{y} \, + \boldsymbol{\mathcal{B}}_{2,i}\, \mathbf{u} \, + \, \boldsymbol{\mathcal{D}}_{2,i}\, \boldsymbol{\xi} \, + \boldsymbol{\mathcal{F}}_{2,i}\, \mathbf{f}_a\right) + \dot{\mathbf{f}}_s \, . && (6.4) \end{aligned}$$

The structure of the transformed system matrices is as follows: $\boldsymbol{\mathcal{A}}_i \; = \; \mathbf{T}_i \mathbf{A}_i \mathbf{T}_i^{-1} \; = \; \begin{bmatrix} \boldsymbol{\mathcal{A}}_{11,i} & \boldsymbol{\mathcal{A}}_{12,i} \\ \boldsymbol{\mathcal{A}}_{21,i} & \boldsymbol{\mathcal{A}}_{22,i} \end{bmatrix}$, $\boldsymbol{\mathcal{B}}_i = \mathbf{T}_i\, \mathbf{B}_i = \begin{bmatrix} \boldsymbol{\mathcal{B}}_{1,i}^T \ \boldsymbol{\mathcal{B}}_{2,i}^T \end{bmatrix}^T$, $\boldsymbol{\mathcal{D}}_i = \mathbf{T}_i\, \mathbf{D}_i = \begin{bmatrix} \mathbf{0}^T \ \boldsymbol{\mathcal{D}}_{2,i}^T \end{bmatrix}^T$, $\boldsymbol{\mathcal{F}}_i = \mathbf{T}_i\, \mathbf{F}_i = \begin{bmatrix} \mathbf{0}^T \ \boldsymbol{\mathcal{F}}_{2,i}^T \end{bmatrix}^T$.

For a stable observer to exist, the following three existence conditions have to be fulfilled [Edwards and Spurgeon, 1998; Gerland et al., 2010b,a]:

Condition 1. The uncertainties and actuator faults are unknown but bounded: $\| \boldsymbol{\xi}^T\left(t\right) \ \mathbf{f}_a^T\left(t\right) \|^T \le \Xi$. Furthermore, individual bounds exist: $\| \boldsymbol{\xi}\left(t\right) \|^T \; \le \; \Xi_\xi$ and $\mathbf{f}_a\left(t\right) \; \le \; \Xi_{f_a}$. The sensor faults and their derivatives are assumed to be bounded, too: $\|\mathbf{f}_s\| \le \Psi$, $\|\dot{\mathbf{f}}_s\| \le \Psi_d$. Moreover, the system states and inputs are assumed to be bounded.

Condition 2. Let q be defined as the number of columns of $\left[\mathbf{D}_i \ \mathbf{F}_i\right]$. Then, the condition $q_i = \text{rank}\left(\mathbf{C}\left[\mathbf{D}_i \ \mathbf{F}_i\right]\right) \; = \; \text{rank}\left[\mathbf{D}_i \ \mathbf{F}_i\right]$ must be fulfilled. Furthermore, it must hold that $p > q_i$, where p is the number of measurable system states.

Condition 3. All invariant zeros of $\left(\mathbf{A}_i, \ \left[\mathbf{D}_i \ \mathbf{F}_i\right], \mathbf{C}\right)$ must lie in $\mathbb{C}_-$.

The TS sliding mode observer in transformed form is given by

$$\begin{aligned} \dot{\hat{\mathbf{x}}}_1 \; &= \; \sum_{i=1}^{N_r} h_i\left(\mathbf{z}\right) \left(\boldsymbol{\mathcal{A}}_{11,i}\, \hat{\mathbf{x}}_1 \, + \, \boldsymbol{\mathcal{A}}_{12,i}\, \hat{\mathbf{y}} \, + \boldsymbol{\mathcal{B}}_{1,i}\, \mathbf{u} \, - \boldsymbol{\mathcal{A}}_{12,i}\, \tilde{\mathbf{e}}_y\right) \, , && (6.5)\\ \dot{\hat{\mathbf{y}}} \; &= \; \sum_{i=1}^{N_r} h_i\left(\mathbf{z}\right) \left(\boldsymbol{\mathcal{A}}_{21,i}\, \hat{\mathbf{x}}_1 \, + \, \boldsymbol{\mathcal{A}}_{22,i}\, \hat{\mathbf{y}} \, + \boldsymbol{\mathcal{B}}_{2,i}\, \mathbf{u} \, - \, \left(\boldsymbol{\mathcal{A}}_{22,i} \, - \boldsymbol{\mathcal{A}}_{22}^s\right) \tilde{\mathbf{e}}_y \, + \, \boldsymbol{\nu}\right) \, , && (6.6) \end{aligned}$$

where

$$\tilde{\mathbf{e}}_y := \hat{\mathbf{y}} - \left(\mathbf{y} + \mathbf{f}_s\right) \tag{6.7}$$

---

[1] see Appendix A.4.2 and [Edwards and Spurgeon, 1998; Gerland et al., 2010b,a] for a description of the transformation matrices.

denotes the output error including sensor faults and $\boldsymbol{\mathcal{A}}_{22}^{s}$ is a stable design matrix.

The discontinuous term $\boldsymbol{\nu}$ necessary to establish and maintain a sliding motion is given by

$$\boldsymbol{\nu} = -\rho \frac{\mathbf{P}_2 \, \mathbf{e}_y}{\|\mathbf{P}_2 \, \mathbf{e}_y\|} \quad , \quad \text{if } \mathbf{e}_y \neq \mathbf{0} \, , \tag{6.8}$$

where ρ is a scalar gain factor and $\mathbf{P}_2$ is the unique symmetric positive definite solution of the Lyapunov equation

$$\mathbf{P}_2 \, \boldsymbol{\mathcal{A}}_{22}^{s} + \boldsymbol{\mathcal{A}}_{22}^{s\;T} \, \mathbf{P}_2 = -\mathbf{Q}_2 \, , \tag{6.9}$$

where $\mathbf{Q}_2$ is a symmetric positive semidefinite design matrix. An obvious choice for $\boldsymbol{\mathcal{A}}_{22}^{s}$ is a diagonal matrix where the elements are the desired poles for the output error dynamics [Groß, 2010].

Note that the discontinuous term $\boldsymbol{\nu}$ in Equation (6.8) is undefined in the case of $\mathbf{e}_y = \mathbf{0}$.

Once the sliding surface

$$\mathcal{S} = \left\{ \mathbf{e}\,(t) \in \mathbb{R}^n \, : \tilde{\mathbf{e}}_y = \boldsymbol{\mathcal{C}} \, \begin{pmatrix} \mathbf{e}_1^T & \tilde{\mathbf{e}}_y^T \end{pmatrix}^T = \mathbf{0} \right\} \tag{6.10}$$

is reached, the TS SM observer tries to maintain the sliding motion on $\mathcal{S}$.

Using the inverse transformation $\mathbf{T}_i^{-1}$, the observer gains can be obtained in the original coordinates as

$$\mathbf{G}_{l,i} = \mathbf{T}_i^{-1} \begin{pmatrix} \boldsymbol{\mathcal{A}}_{12,i} \\ \boldsymbol{\mathcal{A}}_{22,i} - \boldsymbol{\mathcal{A}}_{22}^{s} \end{pmatrix} \, , \quad \mathbf{G}_{n,i} = \mathbf{T}_i^{-1} \begin{pmatrix} \mathbf{0}_{(n-p)\times p} \\ \mathbf{I}_p \end{pmatrix} \, . \tag{6.11}$$

such that the TS sliding mode observer in the original coordinates has the general structure [Gerland et al., 2010a]

$$\dot{\hat{\mathbf{x}}} = \sum_{i=1}^{N_r} h_i\,(\mathbf{z})\,(\mathbf{A}_i \hat{\mathbf{x}} + \mathbf{B}_i \mathbf{u} - \mathbf{G}_{l,i} \, \mathbf{e}_y + \mathbf{G}_{n,i} \, \boldsymbol{\nu}) \, . \tag{6.12}$$

A note of caution must be mentioned in connection with using the inverse transformation to obtain the observer in structure (6.12): In case the transformation matrices $\mathbf{T}_i$ obtained from (6.2) are not equal for each index i, the transformation is no canonical transformation. It is a question of current research in which cases a well-defined transformation can be achieved [Schulte and Georg, 2014].

Apart from the three existence conditions for the TS SMO on page 126, a further assumption is that the error vector $\mathbf{e}_1 = \hat{\mathbf{x}}_1 - \mathbf{x}_1$ of the transformed unmeasurable states is bounded: $\|\mathbf{e}_1\| < \Gamma$. When investigating the stability of the error system $\mathbf{e}_1$, an LMI design condition is obtained (see Appendix A.4.3).

6.2 TS Sliding Mode Observer for Fault Reconstruction

As demonstrated in [Edwards and Spurgeon, 1998], sliding mode observers can be effectively applied for the reconstruction of external disturbances as well as actuator and sensor faults, where the fault signals are reconstructed using the so-called equivalent output injection signal. Similarly to the equivalent control action in sliding mode control structures, for sliding mode observers the equivalent output injection signal [Edwards et al., 2000] describes the average behaviour of the discontinuous component $\boldsymbol{\nu}$ [Utkin, 1992]. It is a measure for the effort to maintain the sliding motion [Edwards et al., 2000] on the sliding surface and can be approximated to arbitrary precision by introducing a small positive scalar δ into the discontinuous component (6.8):

$$\boldsymbol{\nu}_{\mathrm{eq}} = -\rho \, \frac{\mathbf{P}_2 \, \tilde{\mathbf{e}}_y}{\|\mathbf{P}_2 \, \tilde{\mathbf{e}}_y\| + \delta} \,, \tag{6.13}$$

where δ should be chosen as small as possible, because it influences the quality of the fault reconstruction (see Section 6.3.3).

It was shown in [Gerland et al., 2010a] and [Schulte et al., 2012] that for nonlinear systems, TS sliding mode observers can be used in the same way for fault reconstruction.

Actuator Fault Reconstruction

Consider Equations (6.3) and (6.5) for the non-measurable states of the TS system and of the TS sliding mode observer in transformed form. The derivative of the error term $\mathbf{e}_1 = \hat{\mathbf{x}}_1 - \mathbf{x}_1$ can be derived from these equations and for the case of no sensor faults ($\mathbf{f}_s = \mathbf{0}$) is given by

$$\dot{\mathbf{e}}_1 = \sum_{i=1}^{N_r} h_i\,(\mathbf{z})\; \boldsymbol{\mathcal{A}}_{11,i}\,\mathbf{e}_1 \,. \tag{6.14}$$

Since $\sum_{i=1}^{N_r} h_i\,(\mathbf{z})\; \boldsymbol{\mathcal{A}}_{11,i}$ is a stable matrix, in steady state it holds that $\mathbf{e}_1 = 0$.

From Equations (6.4), (6.6) and the definition (6.7) of the error term $\tilde{\mathbf{e}}_y$, the derivative of $\tilde{\mathbf{e}}_y$ is obtained as

$$\dot{\tilde{\mathbf{e}}}_y = \sum_{i=1}^{N_r} h_i\,(\mathbf{z}) \left(\boldsymbol{\mathcal{A}}_{21,i}\,\mathbf{e}_1 + \boldsymbol{\mathcal{A}}_{22,i}\,\mathbf{f}_s + \boldsymbol{\mathcal{A}}_{22}^{s}\,\tilde{\mathbf{e}}_y - [\boldsymbol{\mathcal{D}}_{2,i}\;\boldsymbol{\mathcal{F}}_{2,i}] \begin{pmatrix} \boldsymbol{\xi} \\ \mathbf{f}_a \end{pmatrix} + \boldsymbol{\nu} \right) - \dot{\mathbf{f}}_s \,. \tag{6.15}$$

Once a sliding motion has been established, $\tilde{\mathbf{e}}_y = 0$, and $\dot{\tilde{\mathbf{e}}}_y = 0$, such that equation (6.15) in the case without sensor faults simplifies to

$$\dot{\tilde{\mathbf{e}}}_y = 0 = \sum_{i=1}^{N_r} h_i\,(\mathbf{z}) \left(\boldsymbol{\mathcal{A}}_{21,i}\,\mathbf{e}_1 + \boldsymbol{\nu}_{\mathrm{eq}} - [\boldsymbol{\mathcal{D}}_{2,i}\;\boldsymbol{\mathcal{F}}_{2,i}] \begin{pmatrix} \boldsymbol{\xi} \\ \mathbf{f}_a \end{pmatrix} \right) . \tag{6.16}$$

In this case, the discontinuous term $\boldsymbol{\nu}$ was replaced by the equivalent output injection signal $\boldsymbol{\nu}_{\mathrm{eq}}$, since the latter describes the action on the sliding surface, where $\boldsymbol{\nu}$ is undefined.

Substituting the steady-state solution of equation (6.14) into (6.16) yields

$$0 = \sum_{i=1}^{N_r} h_i\,(\mathbf{z}) \left(\boldsymbol{\nu}_{\mathrm{eq}} - [\boldsymbol{\mathcal{D}}_{2,i}\;\boldsymbol{\mathcal{F}}_{2,i}] \begin{pmatrix} \boldsymbol{\xi} \\ \mathbf{f}_a \end{pmatrix} \right) . \tag{6.17}$$

Thus, the equivalent output injection signal is given by

$$\boldsymbol{\nu}_{\text{eq}} = \sum_{i=1}^{N_r} h_i(\mathbf{z}) \, [\boldsymbol{\mathcal{D}}_{2,i} \, \boldsymbol{\mathcal{F}}_{2,i}] \begin{pmatrix} \boldsymbol{\xi} \\ \mathbf{f}_a \end{pmatrix} . \tag{6.18}$$

From (6.18), the unmodelled dynamics and actuator faults can be reconstructed using the relation

$$\begin{pmatrix} \hat{\boldsymbol{\xi}} \\ \hat{\mathbf{f}}_a \end{pmatrix} = [\boldsymbol{\mathcal{D}}_2(\mathbf{z}) \;\; \boldsymbol{\mathcal{F}}_2(\mathbf{z})]^{+} \, \boldsymbol{\nu}_{\text{eq}} , \tag{6.19}$$

where $(\cdot)^{+}$ denotes the pseudo-inverse of the corresponding matrix and the matrices

$$\boldsymbol{\mathcal{D}}_2(\mathbf{z}) := \sum_{i=1}^{N_r} h_i(\mathbf{z}) \, \boldsymbol{\mathcal{D}}_{2,i} \qquad , \qquad \boldsymbol{\mathcal{F}}_2(\mathbf{z}) := \sum_{i=1}^{N_r} h_i(\mathbf{z}) \, \boldsymbol{\mathcal{F}}_{2,i} , \tag{6.20}$$

have been introduced for the sake of clarity.

Sensor Fault Reconstruction

The equivalent output injection signal $\boldsymbol{\nu}_{\text{eq}}$ can also be used for the reconstruction of sensor faults. The derivation of a fault reconstruction scheme works analogously to the one in [Edwards and Spurgeon, 1998], but adapted to a TS sliding mode observer. Consider equations (6.3) and (6.5) for the non-measurable states of the TS system and of the TS sliding mode observer in transformed form. The derivative of the error term $\mathbf{e}_1 = \hat{\mathbf{x}}_1 - \mathbf{x}_1$ can be derived from these equations as

$$\dot{\mathbf{e}}_1 = \sum_{i=1}^{N_r} h_i(\mathbf{z}) \left(\boldsymbol{\mathcal{A}}_{11,i} \, \mathbf{e}_1 + \boldsymbol{\mathcal{A}}_{12,i} \, \mathbf{f}_s \right) . \tag{6.21}$$

For the sake of clarity, the weighted TS matrices are written as $\boldsymbol{\mathcal{A}}_{jk}(\mathbf{z}) := \sum_{i=1}^{N_r} h_i(\mathbf{z}) \, \boldsymbol{\mathcal{A}}_{jk,i}$.

Consider a case without external disturbances or actuator faults ($\mathbf{D}_i = \mathbf{0}$, $\mathbf{F}_i = \mathbf{0}$). The time-derivatives of the error term $\mathbf{e}_1$ and of the sensor faults are assumed to be approximately zero ($\dot{\mathbf{e}}_1 \approx 0$, $\dot{\mathbf{f}}_s \approx 0$). In this case, the sensor fault $\mathbf{f}_s$ is constant or at least slowly varying and from Equation (6.21) one obtains

$$\mathbf{e}_1 = -\boldsymbol{\mathcal{A}}_{11}^{-1}(\mathbf{z}) \, \boldsymbol{\mathcal{A}}_{12}(\mathbf{z}) \, \mathbf{f}_s . \tag{6.22}$$

From (6.15), the derivative of $\tilde{\mathbf{e}}_y$ for $\mathbf{D}_i = \mathbf{0}$, $\mathbf{F}_i = \mathbf{0}$ and $\dot{\mathbf{f}}_s \approx 0$ is obtained as

$$\dot{\tilde{\mathbf{e}}}_y = \sum_{i=1}^{N_r} h_i(\mathbf{z}) \left(\boldsymbol{\mathcal{A}}_{21,i} \, \mathbf{e}_1 + \boldsymbol{\mathcal{A}}_{22,i} \, \mathbf{f}_s + \boldsymbol{\mathcal{A}}_{22}^{s} \, \tilde{\mathbf{e}}_y + \boldsymbol{\nu} \right) . \tag{6.23}$$

Once a sliding motion has been established, it holds that $\tilde{\mathbf{e}}_y = 0$, and $\dot{\tilde{\mathbf{e}}}_y = 0$, such that equation (6.23) simplifies to

$$\dot{\tilde{\mathbf{e}}}_y = 0 = \sum_{i=1}^{N_r} h_i(\mathbf{z}) \left(\boldsymbol{\mathcal{A}}_{21,i} \, \mathbf{e}_1 + \boldsymbol{\mathcal{A}}_{22,i} \, \mathbf{f}_s + \boldsymbol{\nu} \right) = \boldsymbol{\mathcal{A}}_{21}(\mathbf{z}) \, \mathbf{e}_1 + \boldsymbol{\mathcal{A}}_{22}(\mathbf{z}) \, \mathbf{f}_s + \boldsymbol{\nu}_{\text{eq}} . \tag{6.24}$$

Inserting (6.22) in Equation (6.24) yields

$$0 \;=\; -\boldsymbol{\mathcal{A}}_{21}\left(\mathbf{z}\right)\,\boldsymbol{\mathcal{A}}_{11}^{-1}\left(\mathbf{z}\right)\,\boldsymbol{\mathcal{A}}_{12}\left(\mathbf{z}\right)\,\mathbf{f}_s \;+\; \boldsymbol{\mathcal{A}}_{22}\left(\mathbf{z}\right)\,\mathbf{f}_s \;+\; \boldsymbol{\nu}_{\text{eq}}\,. \tag{6.25}$$

Thus, the equivalent output injection signal is given by

$$\boldsymbol{\nu}_{\text{eq}} \;=\; -\left(\boldsymbol{\mathcal{A}}_{22}\left(\mathbf{z}\right) \;-\; \boldsymbol{\mathcal{A}}_{21}\left(\mathbf{z}\right)\,\boldsymbol{\mathcal{A}}_{11}^{-1}\left(\mathbf{z}\right)\,\boldsymbol{\mathcal{A}}_{12}\left(\mathbf{z}\right)\right)\,\mathbf{f}_s\,. \tag{6.26}$$

From (6.26), the sensor fault signal $\mathbf{f}_s$ can be reconstructed if the matrix $\boldsymbol{\mathcal{A}}_{\text{FDI}}\left(\mathbf{z}\right) := \left(\boldsymbol{\mathcal{A}}_{22}\left(\mathbf{z}\right) \;-\; \boldsymbol{\mathcal{A}}_{21}\left(\mathbf{z}\right)\,\boldsymbol{\mathcal{A}}_{11}^{-1}\left(\mathbf{z}\right)\,\boldsymbol{\mathcal{A}}_{12}\left(\mathbf{z}\right)\right)$ is non-singular:

$$\hat{\mathbf{f}}_s \;=\; -\boldsymbol{\mathcal{A}}_{\text{FDI}}^{-1}\left(\mathbf{z}\right)\,\boldsymbol{\nu}_{\text{eq}}\,. \tag{6.27}$$

Note: It is important to recognise that the matrix $\boldsymbol{\mathcal{A}}_{\text{FDI}}\left(\mathbf{z}\right)$ is constructed in such a way that the TS sum over the membership functions is calculated first, before inverting and multiplying the individual matrices. By contrast, in [Gerland, 2011; Schulte et al., 2012], the matrix operations are applied to the constant submatrices of the TS system and the sum over the membership functions is calculated afterwards: ($\boldsymbol{\mathcal{A}}_{\text{FDI},ij}$ is defined as $\boldsymbol{\mathcal{A}}_{21,i}\,\boldsymbol{\mathcal{A}}_{11,i}^{-1}\,\boldsymbol{\mathcal{A}}_{12,i} \,-\, \boldsymbol{\mathcal{A}}_{22,i}$ and the reconstructed sensor fault signal is calculated as $\hat{\mathbf{f}}_s = \left[\sum\limits_{i=1}^{N_r}\sum\limits_{j=1}^{N_r} h_i\left(\mathbf{z}\right)\,h_j\left(\mathbf{z}\right)\,\boldsymbol{\mathcal{A}}_{\text{FDI},ij}\right]^{-1}\boldsymbol{\nu}_{\text{eq}}$).

However, this is not valid, as this would require that $\left(\sum\limits_{i=1}^{N_r}\boldsymbol{\mathcal{A}}_{11,i}\right)^{-1} = \sum\limits_{i=1}^{N_r}\boldsymbol{\mathcal{A}}_{11,i}^{-1}$, which in general is not the case.

Dynamic Sensor Fault Reconstruction

For sensor faults, the direct reconstruction scheme (6.27) requires that the sensor faults and the errors of the unmeasurable system states are constant or at least slowly varying ($\dot{\mathbf{e}}_1 \approx 0,\, \dot{\mathbf{f}}_s \approx 0$), otherwise the reconstruction can be corrupted [Alwi et al., 2011]. To avoid these downsides, Tan and Edwards have introduced a dynamic sensor fault reconstruction method were sensor faults can be treated as actuator faults within an augmented system [Tan and Edwards, 2003].

In this section, this dynamic sensor fault reconstruction method from [Tan and Edwards, 2003] is derived for nonlinear TS structures such that sensor faults can be reconstructed similarly to actuator faults. To this end, the system outputs are first filtered and the system is augmented with the filtered outputs.

Consider a system with sensor faults in selected output channels. The selection is achieved by introducing a selection matrix $\mathbf{N} \in \mathbb{R}^{p\times r}$ $(n \geq p \geq r)$, where n is the number of system states and p the number of outputs:

$$\begin{aligned}\dot{\mathbf{x}} &= \sum_{i=1}^{N_r} h_i\left(\mathbf{z}\right)\left(\mathbf{A}_i\,\mathbf{x} \,+\, \mathbf{B}_i\,\mathbf{u}\right)\,,\\ \tilde{\mathbf{y}} &= \mathbf{C}\,\mathbf{x} \,+\mathbf{N}\,\mathbf{f}_s\,.\end{aligned} \tag{6.28}$$

Introducing a delay matrix $\mathbf{A}_f = \text{diag}\left(-\frac{1}{\tau_1}\,\cdots\,-\frac{1}{\tau_p}\right)$, with the individual delay time constants τ_i $(i \in \{1,\,\ldots,\,p\})$ for each output, the dynamics of the filtered output vector $\mathbf{y}_f$ is given by:

$$\begin{aligned}\dot{\mathbf{y}}_f &= -\mathbf{A}_f\,\mathbf{y}_f + \mathbf{A}_f\,\tilde{\mathbf{y}} \\ &= -\mathbf{A}_f\,\mathbf{y}_f + \mathbf{A}_f\,\mathbf{C}\,\mathbf{x} + \mathbf{A}_f\,\mathbf{N}\,\mathbf{f}_s\,. \end{aligned} \tag{6.29}$$

Both systems (6.28) and (6.29) can be combined into an augmented system with system vector $\mathbf{x}_a$, where the sensor faults now appear as actuator faults:

$$\begin{pmatrix}\dot{\mathbf{x}}\\ \dot{\mathbf{y}}_f\end{pmatrix} = \sum_{i=1}^{N_r} h_i(\mathbf{z}) \left(\underbrace{\begin{bmatrix}\mathbf{A}_i & \mathbf{0}\\ \mathbf{A}_f\,\mathbf{C} & -\mathbf{A}_f\end{bmatrix}}_{=:\mathbf{A}_{a,i}} \underbrace{\begin{pmatrix}\mathbf{x}\\ \mathbf{y}_f\end{pmatrix}}_{=:\mathbf{x}_a} + \underbrace{\begin{bmatrix}\mathbf{B}_i\\ \mathbf{0}\end{bmatrix}}_{=:\,\mathbf{B}_{a,i}} \mathbf{u} \right) + \underbrace{\begin{bmatrix}\mathbf{0}\\ \mathbf{A}_f\,\mathbf{N}\end{bmatrix}}_{=:\,\mathbf{F}_a} \mathbf{f}_s \tag{6.30}$$

$$\mathbf{y}_f = \underbrace{\begin{bmatrix}\mathbf{0} & \mathbf{I}_{p\times p}\end{bmatrix}}_{=:\,\mathbf{C}_a} \begin{pmatrix}\mathbf{x}\\ \mathbf{y}_f\end{pmatrix}\,. \tag{6.31}$$

In analogy to (6.5) and (6.6), the TS sliding mode observer in transformed form for the augmented system (6.30)-(6.31) is given by

$$\dot{\hat{\mathbf{x}}}_{a,1} = \sum_{i=1}^{N_r} h_i(\mathbf{z})\,\left(\boldsymbol{\mathcal{A}}_{a,11,i}\,\hat{\mathbf{x}}_{a,1} + \boldsymbol{\mathcal{A}}_{a,12,i}\,\hat{\mathbf{y}}_f + \boldsymbol{\mathcal{B}}_{a,1,i}\,\mathbf{u} - \boldsymbol{\mathcal{A}}_{a,12,i}\,\tilde{\mathbf{e}}_y\right)\,, \tag{6.32}$$

$$\dot{\hat{\mathbf{y}}}_f = \sum_{i=1}^{N_r} h_i(\mathbf{z})\,\left(\boldsymbol{\mathcal{A}}_{a,21,i}\,\hat{\mathbf{x}}_{a,1} + \boldsymbol{\mathcal{A}}_{a,22,i}\,\hat{\mathbf{z}}_f + \boldsymbol{\mathcal{B}}_{a,2,i}\,\mathbf{u} - (\boldsymbol{\mathcal{A}}_{a,22,i} - \boldsymbol{\mathcal{A}}^s_{22})\,\tilde{\mathbf{e}}_y + \boldsymbol{\nu}\right)\,, \tag{6.33}$$

where $\tilde{\mathbf{e}}_y = \hat{\mathbf{y}}_f - \mathbf{y}_f$ denotes the error vector, which is now calculated from the filtered output components of the TS SMO and the sensor faults can now be reconstructed like actuator faults using the pseudo-inverse of the transformed form of the augmented matrix $\boldsymbol{\mathcal{F}}_{a,2}$:

$$\hat{\mathbf{f}}_s = \boldsymbol{\mathcal{F}}^{+}_{a,2}\,\boldsymbol{\nu}_{\text{eq}}\,. \tag{6.34}$$

6.3 Sliding Motion for Modified Discontinuous Term

In [Gerland et al., 2010a] it was shown that for a system structure (6.1), a sliding motion exists and is reached in finite time when using a TS sliding mode observer (6.5)-(6.6) with the discontinuous term

$$\boldsymbol{\nu} = -\rho \frac{\mathbf{P}_2 \,\tilde{\mathbf{e}}_y}{\|\mathbf{P}_2 \,\tilde{\mathbf{e}}_y\|} \quad , \quad \text{if } \tilde{\mathbf{e}}_y \neq \mathbf{0} \,, \tag{6.35}$$

where a constant scalar ρ is used as a gain factor.

For the reconstruction of both actuator and sensor faults in simulations of the wind turbine model, the structure (6.35) for the discontinuous term imposes a limitation, which is illustrated by the following example.

Insufficient Reconstruction of Pitch and Generator Torque Actuator Faults

A TS sliding mode observer of model order 5 is designed for the reconstruction of actuator faults, with the observer state vector $\mathbf{x} = (\theta_s \quad \omega_r \quad \omega_g \quad \beta \quad T_g)^T$. All states except for the torsional angle θ_s are measurable: $\mathbf{y} = (\omega_r \quad \omega_g \quad \beta \quad T_g)^T$.

The following observer parameters were used: $\boldsymbol{\mathcal{A}}^s_{22} = \operatorname{diag}(-10 \cdots -10)$, $\delta = 0.05$, $\alpha = 0$, $\rho = 200$.

Two case studies were conducted:
First, only an additive pitch angle fault ($f_a = 0.1$ rad, active between 30 s and 70 s) was simulated using the nominal model (2.29). The reconstructed fault is depicted in Figure 6.1, together with the actual fault.

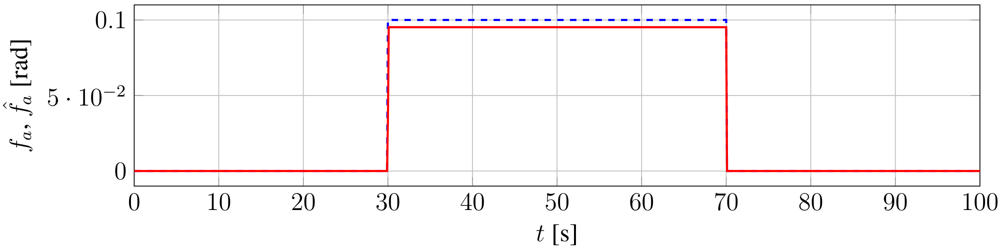
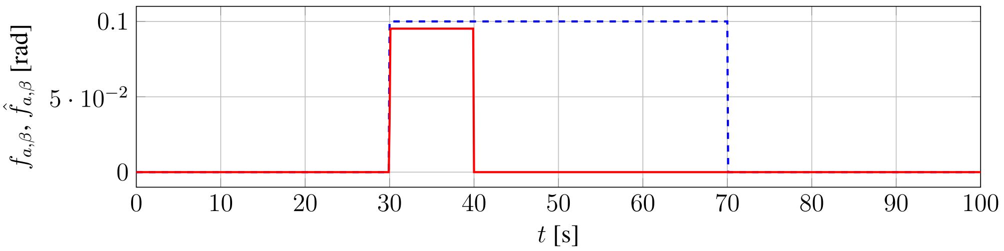


Figure 6.1: Actual and reconstructed pitch angle actuator fault in a simulation of the nominal WEA model (2.29); blue dashed line: actual fault; red line reconstructed fault.

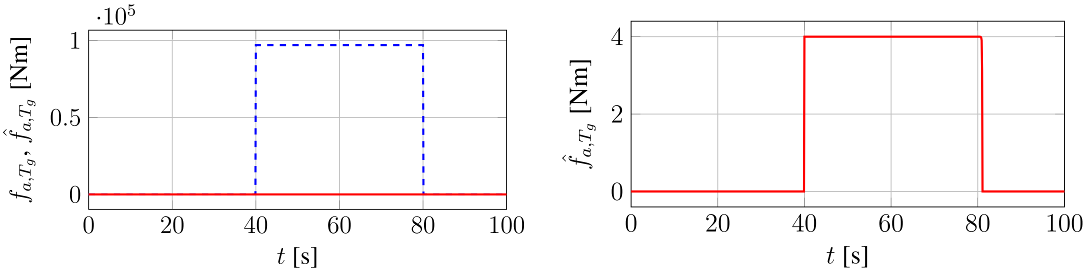
As can be seen from Figure 6.1, a good reconstruction is achieved, both in terms of rapidity and reconstructed fault magnitude. In the above example, the actual fault magnitude is reached to a degree of about 95 %.

In the second case study, in addition to the same pitch angle fault as in the previous example, an additive generator torque actuator fault ($f_{a,T_g} = 97000$ Nm, active between 40 s and 80 s) was simulated. The actual and reconstructed faults are depicted in Figure 6.2.

In this case, neither the pitch angle nor the torque fault is well reconstructed. As can be seen from Figure 6.2a, a good reconstruction of the pitch angle is initially achieved. However, at the onset of the torque fault, the reconstructed pitch angle fault signal is lost. Also, the torque fault is only reconstructed to a degree of 0.004 % of its actual magnitude (see Figure 6.2c).



(a) Actual and reconstructed pitch angle fault



(b) Actual and reconstructed generator torque fault

(c) Reconstructed generator torque fault

Figure 6.2: Actual and reconstructed pitch angle and generator torque actuator faults in a simulation of the nominal WEA model (2.29); blue lines: actual faults; red lines reconstructed faults.

6.3.1 Weighted Switching Term

Explanation for the Insufficient Fault Reconstruction For the wind turbine model, the structure (6.35) for the discontinuous term imposes a limitation for the reconstruction of both actuator and sensor faults, due to the following reason.
The orders of magnitude of the two (main) control inputs, pitch angle and generator torque, vary significantly. The pitch angle can vary between 0 and 1.57 rad, whereas the generator torque can vary between 0 and $\approx\ 4.2 \cdot 10^6$ Nm (for the model without gearbox). These ranges can be assumed as the maximum ranges for both actuator or sensor faults concerning pitch angle and generator torque. Since the generator torque is also included as a system state and output in the TS SMO model, in the case of torque actuator/sensor faults, the T_g-component of the output error $\tilde{\mathbf{e}}_y$ therefore assumes values that largely outweigh the other components of $\tilde{\mathbf{e}}_y$. In this case, due to the normalisation of the output error in equation (6.35) for the discontinuous term $\boldsymbol{\nu}$, only or nearly only the T_g component $\tilde{\mathbf{e}}_y$ contributes to the vector $\frac{\mathbf{P}_2\mathbf{e}_y}{\|\mathbf{P}_2\mathbf{e}_y\|}$ and thereby to $\boldsymbol{\nu}$, unless the matrix $\mathbf{P}_2$ introduces additional weighting, which is usually not the case. The gain factor ρ determines the maximum magnitude of the vector $\boldsymbol{\nu}$ and thereby the maximum magnitude for fault reconstruction, which may quickly reach its limit in a scenario just described. Reconstructing faults with large orders of magnitude would thus require a large value of ρ. This, however, may lead to an erroneous observer behaviour.

To overcome this limitation, two modifications are applied in this work for the calculation of the discontinuous term $\boldsymbol{\nu}$. The specific choice of the modifications was found and adjusted through simulation tests.

1. Modification : The output error vector is multiplied by a diagonal, positive definite weighting matrix for the individual output components:

$$\tilde{\mathbf{e}}_y \ \rightarrow\ \mathbf{W}\,\tilde{\mathbf{e}}_y\,, \tag{6.36}$$

where the elements of the weighting matrix $\mathbf{W}$ are defined as the reciprocal values of the estimated maximum absolute values of the output vector components:

$$\mathbf{W} = \operatorname{diag}(W_1 \cdots W_p) = \operatorname{diag}\left(\frac{1}{|y_{\max,1}|} \cdots \frac{1}{|y_{\max,p}|}\right) . \tag{6.37}$$

2. Modification : The scalar sliding mode gain ρ is replaced by a diagonal, positive definite gain matrix $\boldsymbol{\rho}$, such that the gains can be adjusted separately for each output component:

$$\boldsymbol{\rho} = \operatorname{diag}(\rho_1 \cdots \rho_p) . \tag{6.38}$$

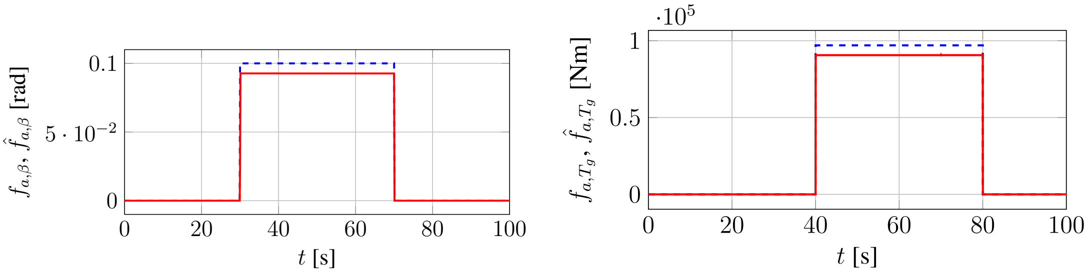
The standard relation (6.8) for the discontinuous term $\boldsymbol{\nu}$ is thus replaced by the weighted term

$$\boldsymbol{\nu} = -\boldsymbol{\rho}\,\frac{\mathbf{W}\,\mathbf{P}_2\,\tilde{\mathbf{e}}_y}{\|\mathbf{W}\,\mathbf{P}_2\,\tilde{\mathbf{e}}_y\|} \quad , \quad \text{if } \tilde{\mathbf{e}}_y \neq \mathbf{0} , \tag{6.39}$$

The modified equivalent output injection signal, i.e. the continuous approximation of (6.39) for simulations is given by:

$$\boldsymbol{\nu}_{\text{eq}} = -\boldsymbol{\rho}\,\frac{\mathbf{W}\,\mathbf{P}_2\,\tilde{\mathbf{e}}_y}{\|\mathbf{W}\,\mathbf{P}_2\,\tilde{\mathbf{e}}_y\| + \delta} . \tag{6.40}$$

Simulating the same pitch angle and torque actuator faults as described on page 78 using the modified equivalent output injection signal (6.40), the obtained fault reconstruction results are significantly better, which can be seen in Figure 6.3.



(a) Actual and reconstructed pitch angle fault

(b) Actual and reconstructed generator torque fault

Figure 6.3: Actual and reconstructed pitch angle and generator torque actuator faults in a simulation of the nominal WEA model (2.29) using the modified equivalent output injection signal (6.40); blue lines: actual faults; red lines reconstructed faults.

Here, the sliding mode gain matrix was chosen as

$\boldsymbol{\rho} = \operatorname{diag}\begin{pmatrix}200 & 200 & 200 & 6 \cdot 10^8\end{pmatrix}$

and the maximum values for the weighting matrix $\mathbf{W}$ were estimated as

$\omega_{r,\max} = \omega_{g,\max} = 1.6\,\frac{\text{rad}}{\text{s}}$, $\beta_{\max} = 1.5708\,\text{rad}$, $T_{g,\max} = 4.2 \cdot 10^6\,\text{Nm}$.

The additional design freedom that is available by introducing the weighting matrix $\mathbf{W}$ and the gain matrix $\boldsymbol{\rho}$ allows to tune the switching term for fault reconstruction even for systems where the orders of magnitude of the individual output components differ significantly. This is not only limited to the wind turbine example but opens up new design possibilities for sliding mode observers in general.

A further possible augmentation of the discontinuous switching term would be to choose different $\boldsymbol{\rho}$-matrices for each TS submodel [Pöschke, Georg, and Schulte, 2014], i.e.:

$$\boldsymbol{\nu} = -\sum_{i=1}^{N_r} h_i(\mathbf{z})\, \boldsymbol{\rho}_i \frac{\mathbf{W}\,\mathbf{P}_2\,\tilde{\mathbf{e}}_y}{\|\mathbf{W}\,\mathbf{P}_2\,\tilde{\mathbf{e}}_y\|} \quad , \quad \text{if } \tilde{\mathbf{e}}_y \neq \mathbf{0}\,. \tag{6.41}$$

6.3.2 Existence of an Ideal Sliding Motion

In this section, a formal proof is given that a sliding motion can be established using a TS SMO with the modified switching term (6.39) and that the sliding surface is reached in finite time. The proof follows along the lines of those given in [Edwards and Spurgeon, 1994, 1998] for the sliding mode observer and in [Gerland et al., 2010a] for the TS SMO with the standard discontinuous term (6.8).

Proposition

For the TS structure (6.3)-(6.4) and the TS SMO (6.5)-(6.6) with the modified discontinuous term (6.39), an ideal sliding motion takes place on
$\mathcal{S} = \{(\mathbf{e}_1\,,\,\tilde{\mathbf{e}}_y) : \tilde{\mathbf{e}}_y = 0\}$ and the sliding surface is reached in finite time.
Necessary conditions are: $\boldsymbol{\mathcal{A}}_{22}^s$, $\mathbf{P}_2$, $\mathbf{W}$, and $\boldsymbol{\rho}$ must be diagonal matrices. The conditions that $\boldsymbol{\mathcal{A}}_{22}^s$ and $\mathbf{P}_2$ must be diagonal is not an overly restrictive constraint in practice, since a natural choice for $\boldsymbol{\mathcal{A}}_{22}^s$ is a diagonal matrix with the desired poles for the dynamics of $\mathbf{e}_1$ [Gerland et al., 2010a]. Choosing also $\mathbf{Q}_2$ as diagonal in the Lyapunov equation (6.9) for $\boldsymbol{\mathcal{A}}_{22}^s$ yields a diagonal $\mathbf{P}_2$.

Proof 1 (Reachability)

It must be shown that a Lyapunov function $V_s(\tilde{\mathbf{e}}_y)$ exists that fulfills an η-reachability condition. Let $\tilde{\mathbf{P}}_2$ be defined as $\tilde{\mathbf{P}}_2 := \mathbf{P}_2\mathbf{W}$ and a candidate Lyapunov function as $V_s = \tilde{\mathbf{e}}_y^T\,\tilde{\mathbf{P}}_2\,\tilde{\mathbf{e}}_y$. Since $\mathbf{W}$ and $\mathbf{P}_2$ are positive definite, $\tilde{\mathbf{P}}_2$ is also positive definite. It directly follows that $V_s > 0 \quad (\tilde{\mathbf{e}}_y \neq 0)$.
For the proof of $\dot{V}_s < 0$, the following relation will first be shown:

$$2\,\tilde{\mathbf{e}}_y^T\,\tilde{\mathbf{P}}_2\,\boldsymbol{\mathcal{A}}_{22}^s\,\tilde{\mathbf{e}}_y = -\,\tilde{\mathbf{e}}_y^T\,\mathbf{Q}_2\,\mathbf{W}\,\tilde{\mathbf{e}}_y \leq 0\,. \tag{6.42}$$

The second inequality is obvious, as $\mathbf{Q}_2$ and $\mathbf{W}$ are positive definite.
Proof of the first equality making use of the Lyapunov equation (6.9):

$$\begin{aligned}
2\,\tilde{\mathbf{e}}_y^T\,\tilde{\mathbf{P}}_2\,\boldsymbol{\mathcal{A}}_{22}^s\,\tilde{\mathbf{e}}_y &= \tilde{\mathbf{e}}_y^T\,\tilde{\mathbf{P}}_2\,\boldsymbol{\mathcal{A}}_{22}^s\,\tilde{\mathbf{e}}_y + \tilde{\mathbf{e}}_y^T\left(\tilde{\mathbf{P}}_2\,\boldsymbol{\mathcal{A}}_{22}^s\right)^T\tilde{\mathbf{e}}_y \\
&= \tilde{\mathbf{e}}_y^T\,\mathbf{P}_2\,\mathbf{W}\,\boldsymbol{\mathcal{A}}_{22}^s\,\tilde{\mathbf{e}}_y + \tilde{\mathbf{e}}_y^T\,(\boldsymbol{\mathcal{A}}_{22}^s)^T\,\mathbf{W}\,\mathbf{P}_2\,\tilde{\mathbf{e}}_y \\
&= \tilde{\mathbf{e}}_y^T\,\underbrace{\left(\mathbf{P}_2\,\boldsymbol{\mathcal{A}}_{22}^s + (\boldsymbol{\mathcal{A}}_{22}^s)^T\,\mathbf{P}_2\right)}_{=-\mathbf{Q}_2}\,\mathbf{W}\,\tilde{\mathbf{e}}_y \\
&= -\tilde{\mathbf{e}}_y^T\,\mathbf{Q}_2\,\mathbf{W}\,\tilde{\mathbf{e}}_y \leq 0\,.
\end{aligned}$$

Here it was exploited that $\mathbf{W}^T = \mathbf{W}$, $\mathbf{W}\,\mathbf{P}_2 = \mathbf{P}_2\,\mathbf{W}$ and $\mathbf{W}\,\boldsymbol{\mathcal{A}}_{22}^s = \boldsymbol{\mathcal{A}}_{22}^s\,\mathbf{W}$, since $\mathbf{W}$, $\mathbf{P}_2$, and $\boldsymbol{\mathcal{A}}_{22}^s$ are diagonal matrices. For V_s to be a Lyapunov function, the relation $\dot{V}_s < 0$ must hold. From the definition of V_s it follows:

$$\dot{V}_s = 2\,\tilde{\mathbf{e}}_y^T\,\tilde{\mathbf{P}}_2\,\dot{\tilde{\mathbf{e}}}_y\,. \tag{6.43}$$

From equations (6.4), (6.6) and $\tilde{\mathbf{e}}_y = \hat{\mathbf{y}} - \tilde{\mathbf{y}}$, the derivative of the output error is obtained as

$$\dot{\tilde{\mathbf{e}}}_y \;=\; \sum_{i=1}^{N_r} h_i\,(\mathbf{z})\,\left(\boldsymbol{\mathcal{A}}_{21,i}\,\mathbf{e}_1 \,+\, \boldsymbol{\mathcal{A}}_{22,i}\,\mathbf{f}_s \,+\, \boldsymbol{\mathcal{A}}_{22}^s\,\tilde{\mathbf{e}}_y \,+\, \boldsymbol{\nu} \,-\, [\boldsymbol{\mathcal{D}}_{2,i}\,\boldsymbol{\mathcal{E}}_{2,i}]\begin{pmatrix}\boldsymbol{\xi}\\ \mathbf{f}_a\end{pmatrix} - \dot{\mathbf{f}}_s\right). \tag{6.44}$$

Inserting (6.44) into (6.43) yields

$$\dot{V}_s \;=\; \sum_{i=1}^{N_r} h_i\,(\mathbf{z})\;2\,\tilde{\mathbf{e}}_y^T\tilde{\mathbf{P}}_2\,\left(\boldsymbol{\mathcal{A}}_{21,i}\,\mathbf{e}_1 \,+\, \boldsymbol{\mathcal{A}}_{22,i}\,\mathbf{f}_s \,+\, \boldsymbol{\mathcal{A}}_{22}^s\,\tilde{\mathbf{e}}_y \,+\, \boldsymbol{\nu} \,-\, [\boldsymbol{\mathcal{D}}_{2,i}\,\boldsymbol{\mathcal{E}}_{2,i}]\begin{pmatrix}\boldsymbol{\xi}\\ \mathbf{f}_a\end{pmatrix} - \dot{\mathbf{f}}_s\right). \tag{6.45}$$

Using relation (6.42) it thus follows:

$$\dot{V}_s \;\leq\; \sum_{i=1}^{N_r} h_i\,(\mathbf{z})\,\left(2\,\tilde{\mathbf{e}}_y^T\tilde{\mathbf{P}}_2\left[\boldsymbol{\mathcal{A}}_{21,i}\,\mathbf{e}_1 \,+\, \boldsymbol{\mathcal{A}}_{22,i}\,\mathbf{f}_s \,+\, \boldsymbol{\nu} -\, [\boldsymbol{\mathcal{D}}_{2,i}\,\boldsymbol{\mathcal{E}}_{2,i}]\begin{pmatrix}\boldsymbol{\xi}\\ \mathbf{f}_a\end{pmatrix} - \dot{\mathbf{f}}_s\right]\right). \tag{6.46}$$

Defining the vector

$$\boldsymbol{\kappa}_i \;:=\; \boldsymbol{\mathcal{A}}_{21,i}\,\mathbf{e}_1 \,+\, \boldsymbol{\mathcal{A}}_{22,i}\,\mathbf{f}_s \,-\, [\boldsymbol{\mathcal{D}}_{2,i}\,\boldsymbol{\mathcal{E}}_{2,i}]\begin{pmatrix}\boldsymbol{\xi}\\ \mathbf{f}_a\end{pmatrix} - \dot{\mathbf{f}}_s\,, \tag{6.47}$$

the following estimate for $\dot{V}_s$ holds:

$$\dot{V}_s \;\leq\; \sum_{i=1}^{N_r} h_i(\mathbf{z})\,2\,\left\|\tilde{\mathbf{P}}_2\,\tilde{\mathbf{e}}_y\right\|\;\|\boldsymbol{\kappa}_i\| \,+\, 2\,\tilde{\mathbf{e}}_y^T\,\tilde{\mathbf{P}}_2\,\boldsymbol{\nu}\,. \tag{6.48}$$

Let $\eta > 0$ be a small scalar. If

$$2\,\tilde{\mathbf{e}}_y^T\,\tilde{\mathbf{P}}_2\,\boldsymbol{\nu} = \,-2\sum_{i=1}^{N_r} h_i(\mathbf{z})\left(\left\|\tilde{\mathbf{P}}_2\,\tilde{\mathbf{e}}_y\right\|\,\eta \,+\, \left\|\tilde{\mathbf{P}}_2\,\tilde{\mathbf{e}}_y\right\|\;\|\boldsymbol{\kappa}_i\|\right), \tag{6.49}$$

the following η-reachability condition is fulfilled:

$$\dot{V}_s \;\leq\; -2\,\eta\,\left\|\tilde{\mathbf{P}}_2\,\tilde{\mathbf{e}}_y\right\|\,. \tag{6.50}$$

A condition for the diagonal matrix $\boldsymbol{\rho}$ must now be determined such that both equation (6.49) and thereby relation (6.50) hold. Inserting the definition for the modified discontinuous term (6.39) in equation (6.49) yields (for the case of $\tilde{\mathbf{e}}_y \neq \mathbf{0}$)

$$-2\,\tilde{\mathbf{e}}_y^T\,\tilde{\mathbf{P}}_2\,\boldsymbol{\rho}\,\frac{\tilde{\mathbf{P}}_2\,\tilde{\mathbf{e}}_y}{\left\|\tilde{\mathbf{P}}_2\,\tilde{\mathbf{e}}_y\right\|} \overset{(6.49)}{=} -2\sum_{i=1}^{N_r} h_i(\mathbf{z})\,\left(\left\|\tilde{\mathbf{P}}_2\,\tilde{\mathbf{e}}_y\right\|\,\eta + \left\|\tilde{\mathbf{P}}_2\,\tilde{\mathbf{e}}_y\right\|\,\|\boldsymbol{\kappa}_i\|\right)$$

$$\Leftrightarrow \qquad \tilde{\mathbf{e}}_y^T\,\tilde{\mathbf{P}}_2\,\boldsymbol{\rho}\,\tilde{\mathbf{P}}_2\,\tilde{\mathbf{e}}_y = \sum_{i=1}^{N_r} h_i(\mathbf{z})\,\left(\eta + \|\boldsymbol{\kappa}_i\|\right)\,\left\|\tilde{\mathbf{P}}_2\,\tilde{\mathbf{e}}_y\right\|^2$$

$$\Leftrightarrow \qquad \tilde{\mathbf{e}}_y^T\,\tilde{\mathbf{P}}_2\,\boldsymbol{\rho}\,\tilde{\mathbf{P}}_2\,\tilde{\mathbf{e}}_y = \sum_{i=1}^{N_r} h_i(\mathbf{z})\,\tilde{\mathbf{e}}_y^T\,\tilde{\mathbf{P}}_2\,\left(\eta + \|\boldsymbol{\kappa}_i\|\right)\,\tilde{\mathbf{P}}_2\,\tilde{\mathbf{e}}_y$$

A small enough constant η can always be found for the last identify to be fulfilled if the following inequality holds:

$$\Rightarrow \qquad \tilde{\mathbf{e}}_y^T\,\tilde{\mathbf{P}}_2\,\boldsymbol{\rho}\,\tilde{\mathbf{P}}_2\,\tilde{\mathbf{e}}_y > \tilde{\mathbf{e}}_y^T\,\tilde{\mathbf{P}}_2\,\left(\sum_{i=1}^{N_r} h_i(\mathbf{z})\,\|\boldsymbol{\kappa}_i\|\right)\,\tilde{\mathbf{P}}_2\,\tilde{\mathbf{e}}_y$$

$$\Leftrightarrow \qquad 0 < \tilde{\mathbf{e}}_y^T\,\left[\left(\boldsymbol{\rho} - \sum_{i=1}^{N_r} h_i(\mathbf{z})\,\|\boldsymbol{\kappa}_i\|\,\mathbf{I}_{p\times p}\right)\tilde{\mathbf{P}}_2^2\right]\,\tilde{\mathbf{e}}_y\,. \tag{6.51}$$

The last calculation step could be done because $\tilde{\mathbf{P}}_2$ and $\boldsymbol{\rho}$ are diagonal matrices. Since $\tilde{\mathbf{P}}_2$ is positive definite, a sufficient, but not necessary, condition for (6.51) to be fulfilled is that the diagonal matrix $\left(\boldsymbol{\rho} - \sum_{i=1}^{N_r} h_i(\mathbf{z})\,\|\boldsymbol{\kappa}_i\|\,\mathbf{I}_{p\times p}\right)$ is positive definite, which is fulfilled if and only if for every element of $\boldsymbol{\rho}$ it holds that

$$\rho_k > \sum_{i=1}^{N_r} h_i(\mathbf{z})\,\|\boldsymbol{\kappa}_i\| \qquad (k \in \{1,\ldots,p\})\,. \tag{6.52}$$

Then, (6.49) holds and thus the reachability condition (6.50) is fulfilled.

The procedure that led to condition (6.52) is less conservative than the one presented in [Georg and Schulte, 2014c], were the estimations of $\dot{V}_s$ were done using an upper norm bound $\mathcal{K}_{\text{max}} := \|\boldsymbol{\mathcal{A}}_{21}\|_{\text{max}}\,\Gamma + \|\boldsymbol{\mathcal{A}}_{22}\|_{\text{max}}\,\Psi + \|[\boldsymbol{\mathcal{D}}_2\,\boldsymbol{\mathcal{E}}_2]\|_{\text{max}}\,\Xi + \Psi_d\,.$

Proof 2 (Sliding Surface Reached in Finite Time)

Arguing as in [Edwards and Spurgeon, 1998], it is now shown that the sliding surface $\mathcal{S}$ is reached in finite time. Using the condition

$$\left\|\tilde{\mathbf{P}}_2\,\tilde{\mathbf{e}}_y\right\|^2 = \left(\tilde{\mathbf{P}}_2^{\frac{1}{2}}\,\tilde{\mathbf{P}}_2^{\frac{1}{2}}\,\tilde{\mathbf{e}}_y\right)^T\left(\tilde{\mathbf{P}}_2^{\frac{1}{2}}\,\tilde{\mathbf{P}}_2^{\frac{1}{2}}\,\tilde{\mathbf{e}}_y\right) = \left(\tilde{\mathbf{e}}_y^T\,\tilde{\mathbf{P}}_2^{\frac{1}{2}}\right)\,\tilde{\mathbf{P}}_2\,\underbrace{\left(\tilde{\mathbf{P}}_2^{\frac{1}{2}}\,\tilde{\mathbf{e}}_y\right)}_{=:\,\tilde{\mathbf{e}}_{yP}} = \tilde{\mathbf{e}}_{yP}^T\,\tilde{\mathbf{P}}_2\,\tilde{\mathbf{e}}_{yP}\,,$$

it holds that $\|\tilde{\mathbf{e}}_{yP}\|^2 = \tilde{\mathbf{e}}_y^T\,\tilde{\mathbf{P}}_2\,\tilde{\mathbf{e}}_y = V_s$. For any symmetric matrix $\mathbf{A}$, the Rayleigh principle is fulfilled: $\lambda_{\text{min}}(\mathbf{A})\,\|\mathbf{x}\|^2 \le \mathbf{x}^T\,\mathbf{A}\,\mathbf{x} \le \lambda_{\text{max}}(\mathbf{A})\,\|\mathbf{x}\|^2$, where λ_{min} and λ_{max} denote the minimum and maximum eigenvalues of $\mathbf{A}$. Applying the Rayleigh principle to $\tilde{\mathbf{e}}_{yP}$ and $\tilde{\mathbf{P}}_2$ yields: $\lambda_{\text{min}}\left(\tilde{\mathbf{P}}_2\right)\,\|\tilde{\mathbf{e}}_{yP}\|^2 \le \tilde{\mathbf{e}}_{yP}^T\,\tilde{\mathbf{P}}_2\,\tilde{\mathbf{e}}_{yP} \le \lambda_{\text{max}}\left(\tilde{\mathbf{P}}_2\right)\,\|\tilde{\mathbf{e}}_{yP}\|^2$ and thus:

$$\lambda_{\text{min}}\left(\tilde{\mathbf{P}}_2\right)\,V_s \le \left\|\tilde{\mathbf{P}}_2\,\tilde{\mathbf{e}}_y\right\|^2 \le \lambda_{\text{max}}\left(\tilde{\mathbf{P}}_2\right)\,V_s\,. \tag{6.53}$$

Using relation (6.53), the reachability condition (6.50) can be written independently of the output error $\tilde{\mathbf{e}}_y$:

$$\dot{V}_s \leq -2\,\eta \left\| \tilde{\mathbf{P}}_2\, \tilde{\mathbf{e}}_y \right\| \leq -2\,\eta \sqrt{\lambda_{\text{min}}\left(\tilde{\mathbf{P}}_2\right)}\, \sqrt{V_s}\,. \tag{6.54}$$

Applying the chain rule for differentiation, the following equality holds: $\frac{d}{dt}\sqrt{V_s} = \frac{1}{2\sqrt{V_s}}\,\dot{V}_s$.
Defining t_r as the time when the sliding surface is reached and making use of
$V_s(t_r) = \tilde{\mathbf{e}}_y^T\,(t_r)\,\tilde{\mathbf{P}}_2\,\tilde{\mathbf{e}}_y\,(t_r) = 0$, the reachability condition (6.54) can be integrated and rearranged to obtain an estimate for t_r:

$$\begin{aligned}
& & \frac{1}{2\sqrt{V_s}}\,\dot{V}_s \quad &\leq \quad -\eta\sqrt{\lambda_{\text{min}}\left(\tilde{\mathbf{P}}_2\right)} \\
&\Rightarrow & \int\limits_0^{t_r} \frac{d}{dt}\left(\sqrt{V_s}\right)\,dt \quad &\leq \quad -\eta\sqrt{\lambda_{\text{min}}\left(\tilde{\mathbf{P}}_2\right)}\int\limits_0^{t_r} dt \\
&\Rightarrow & \left[\underbrace{\sqrt{V_s\,(t_r)}}_{=\,0} - \sqrt{V_s\,(0)}\right] \quad &\leq \quad -\eta\sqrt{\lambda_{\text{min}}\left(\tilde{\mathbf{P}}_2\right)}\,t_r \\
&\Leftrightarrow & t_r \quad &\leq \quad \frac{1}{\eta}\,\frac{\sqrt{V_s\,(0)}}{\sqrt{\lambda_{\text{min}}\left(\mathbf{W}\,\mathbf{P}\right)}}\,.
\end{aligned}$$

The sliding surface $\mathcal{S}$ is thus reached in finite time. This completes the proof of the proposition on page 81.

■

The weighted switching term introduced in Section 6.3.1 and the main part of the proof in Section 6.3.2 were first published in [Georg and Schulte, 2014c] [2].

6.3.3 Choice of the Sliding Mode Gain Matrix

For the example of pitch angle and torque actuator fault described on page 78, condition (6.52) is not fulfilled, which can be seen from a plot of the term $\mathcal{K}_S := \sum_{i=1}^{N_r} h_i(\mathbf{z})\,\|\boldsymbol{\kappa}_i\|$ in Figure 6.4.

Between a simulation time of 40 s and 80 s, i.e., when the generator torque fault is active, $\mathcal{K}_S$ assumes values approximately between 2000 and 4000, which is clearly larger than the first three diagonal components of $\boldsymbol{\rho} = \text{diag}\begin{pmatrix}200 & 200 & 200 & 6\cdot 10^8\end{pmatrix}$. Still, as was shown in Figures 6.3, the achieved fault reconstruction with this selection of $\boldsymbol{\rho}$ is very good and the observer shows a stable behaviour. In fact, when setting the first three diagonal elements of $\boldsymbol{\rho}$ to values greater than 420, the observer behaviour becomes erroneous and no meaningful fault reconstruction is achieved.

These findings indicate that condition (6.52) is a theoretically correct, but very conservative and not necessarily practical condition for the TS sliding mode observer design. Given that the TS SM observer yields good results using the design parameters as on page 78, which do not satisfy condition 6.52, it is therefore necessary to revisit condition (6.51) to gain further insight as to when the η-reachability condition (6.50) is actually fulfilled.

---

[2] Sören Georg and Horst Schulte. Takagi-Sugeno Sliding Mode Observer with a Weighted Switching Action and Application to Fault Diagnosis for Wind Turbines. In Józef Korbicz and Marek Kowal, editors, *Intelligent Systems in Technical and Medical Diagnostics*, volume 230 of *Advances in Intelligent Systems and Computing*, pages 41-52. © Springer-Verlag Berlin Heidelberg 2014, with kind permission from Springer Science and Business Media.

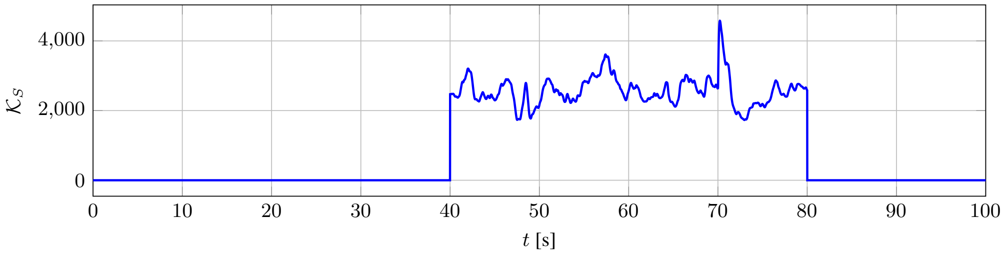


Figure 6.4: Term $\mathcal{K}_S = \sum_{i=1}^{N_r} h_i(\mathbf{z}) \, \|\boldsymbol{\kappa}_i\|$ for a simulation with pitch angle and torque actuator faults as described on page 78

Starting from condition (6.46), a less conservative, approximate design condition can be derived. If it holds that

$$2\,\tilde{\mathbf{e}}_y^T\,\tilde{\mathbf{P}}_2\,\boldsymbol{\nu} = -2\sum_{i=1}^{N_r} h_i(\mathbf{z})\left(\left\|\tilde{\mathbf{P}}_2\,\tilde{\mathbf{e}}_y\right\|\,\eta + \tilde{\mathbf{e}}_y^T\,\tilde{\mathbf{P}}_2\,\boldsymbol{\kappa}_i\right)\,, \tag{6.55}$$

the reachability condition (6.50) is fulfilled.

Inserting the definition (6.39) for the discontinuous term in equation (6.55) (for the case of $\tilde{\mathbf{e}}_y \neq \mathbf{0}$) yields

$$\begin{aligned} -2\,\tilde{\mathbf{e}}_y^T\,\tilde{\mathbf{P}}_2\,\boldsymbol{\rho}\,\frac{\tilde{\mathbf{P}}_2\,\tilde{\mathbf{e}}_y}{\left\|\tilde{\mathbf{P}}_2\,\tilde{\mathbf{e}}_y\right\|} &\overset{(6.49)}{=} -2\sum_{i=1}^{N_r} h_i(\mathbf{z})\,\left(\left\|\tilde{\mathbf{P}}_2\,\tilde{\mathbf{e}}_y\right\|\,\eta + \tilde{\mathbf{e}}_y^T\,\tilde{\mathbf{P}}_2\,\boldsymbol{\kappa}_i\right) \\ \Leftrightarrow \qquad \tilde{\mathbf{e}}_y^T\,\tilde{\mathbf{P}}_2\,\boldsymbol{\rho}\,\tilde{\mathbf{P}}_2\,\tilde{\mathbf{e}}_y &= \sum_{i=1}^{N_r} h_i(\mathbf{z})\,\left(\eta\,\left\|\tilde{\mathbf{P}}_2\,\tilde{\mathbf{e}}_y\right\|^2 + \tilde{\mathbf{e}}_y^T\tilde{\mathbf{P}}_2\,\boldsymbol{\kappa}_i\,\left\|\tilde{\mathbf{P}}_2\,\tilde{\mathbf{e}}_y\right\|\right) \end{aligned}$$

A small enough constant η can always be found for the last identify to be fulfilled if the following inequality holds:

$$\tilde{\mathbf{e}}_y^T\,\tilde{\mathbf{P}}_2\,\left(\boldsymbol{\rho}\,\tilde{\mathbf{P}}_2\,\tilde{\mathbf{e}}_y - \sum_{i=1}^{N_r} h_i(\mathbf{z})\,\boldsymbol{\kappa}_i\,\left\|\tilde{\mathbf{P}}_2\,\tilde{\mathbf{e}}_y\right\|\right) > 0. \tag{6.56}$$

Defining the vector sum $\boldsymbol{\kappa}_s := \sum_{i=1}^{N_r} h_i(\mathbf{z})\,\boldsymbol{\kappa}_i$, a sufficient condition for (6.56) to be fulfilled is

$$\tilde{\mathbf{e}}_y^T\,\tilde{\mathbf{P}}_2\,\boldsymbol{\rho}\,\tilde{\mathbf{P}}_2\,\tilde{\mathbf{e}}_y > \tilde{\mathbf{e}}_y^T\,\tilde{\mathbf{P}}_2\,\boldsymbol{\kappa}_s\,\left\|\tilde{\mathbf{P}}_2\,\tilde{\mathbf{e}}_y\right\|\,. \tag{6.57}$$

To exploit the design freedom of the matrix structure of $\boldsymbol{\rho}$, it would be desirable to obtain a separate design condition for each matrix entry of $\boldsymbol{\rho}$, only in terms of the vector components of $\boldsymbol{\kappa}_s$, without explicit dependence on the vector $\mathbf{e}_P := \tilde{\mathbf{P}}_2\,\tilde{\mathbf{e}}_y$.

A sufficient condition for (6.57) to be fulfilled is

$$\rho_j\,e_{P,j}^2 > |e_{P,j}|\;|\kappa_{s,j}|\;\|\mathbf{e}_P\| \qquad (j \in \{1,\ldots,p\})\,, \tag{6.58}$$

where p denotes the number of output vector components. Dividing by the positive value $e_{P,j}^2$, and introducing the (very rough) estimate $\left|\frac{\|\mathbf{e}_P\|}{e_{P,j}}\right| \approx p$, the following relation can be used as a first design condition:

$$\rho_j > p\,|\kappa_{s,j}| \ . \tag{6.59}$$

To estimate the components of $\boldsymbol{\kappa}_i$, the pitch and torque actuator faults were simulated using a zero sliding mode gain matrix $\boldsymbol{\rho} = \mathrm{diag}\begin{pmatrix}0 & 0 & 0 & 0\end{pmatrix}$. The obtained vector components, which are both equal to the vector sum $\boldsymbol{\kappa}_s$ in this case, are depicted in Figure 6.5.[3]

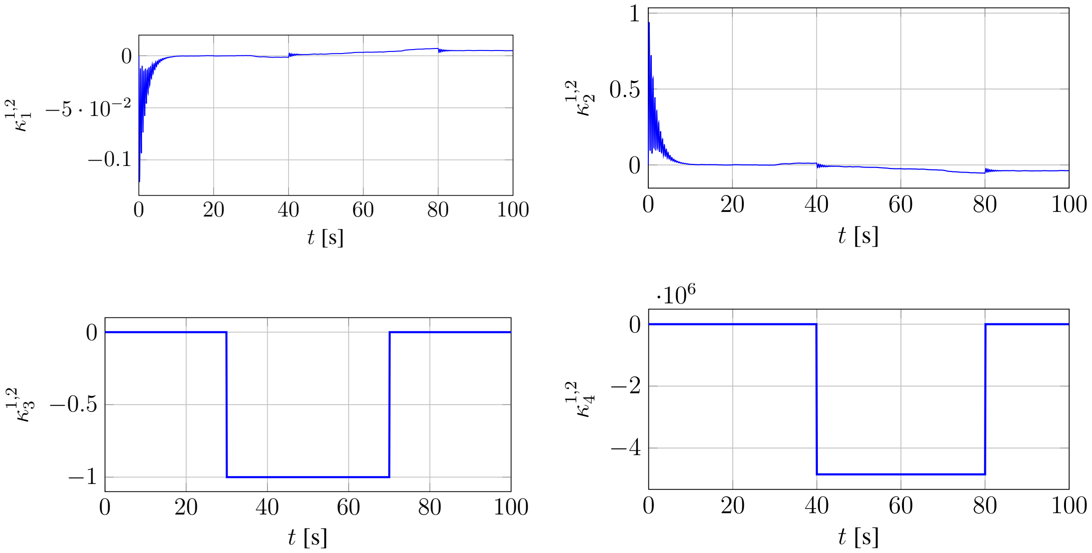


Figure 6.5: Components of the vector $\boldsymbol{\kappa}_i$ as defined in (6.47) for pitch and torque actuator faults as defined on page 78 and a zero sliding mode gain matrix $\boldsymbol{\rho} = \mathrm{diag}\begin{pmatrix}0 & 0 & 0 & 0\end{pmatrix}$.

Inspecting the $\boldsymbol{\kappa}_i$-components in Figure 6.5 and with $p = 4$, a first choice of the sliding mode gain matrix that fulfills (6.59) could be

$$\boldsymbol{\rho} = \mathrm{diag}\begin{pmatrix}1 & 10 & 5 & 16 \cdot 10^6\end{pmatrix} \ . \tag{6.60}$$

Simulating the actuator faults using this design matrix, one already obtains reconstructed actuator faults, however the reconstructed fault magnitudes are still far from the actual ones (see Figure 6.6).

Starting from the first design matrix (6.60), one can increase the matrix components to achieve the desired reconstruction quality. Choosing the following gain matrix yields the best compromise between reconstructed fault magnitude and few erroneous peaks.

$$\boldsymbol{\rho} = \mathrm{diag}\begin{pmatrix}200 & 200 & 200 & 6 \cdot 10^8\end{pmatrix} \ , \tag{6.61}$$

The reconstructed pitch angle and generator torque faults are depicted in Figure 6.7.

In light of the previous remarks, some practical considerations can be made about the choice of the sliding mode gain matrix $\boldsymbol{\rho}$. First, as just described, it is sensible to estimate the vector components of $\boldsymbol{\kappa}_i$ and to

---

[3] In the above example, the vector $\boldsymbol{\kappa}_i$ is equal for both $i = 1$ and $i = 2$, since the matrices $\boldsymbol{\mathcal{A}}_{21,i}$ and $\boldsymbol{\mathcal{F}}_{2,i}$ are equal for both values of i. Thus, it holds that $\boldsymbol{\kappa}_s = \boldsymbol{\kappa}_1 = \boldsymbol{\kappa}_2$.

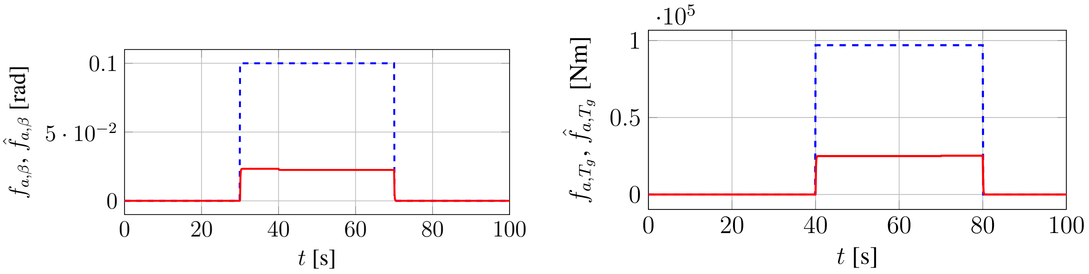


Figure 6.6: Actual and reconstructed actuator faults using the design gain matrix (6.60)

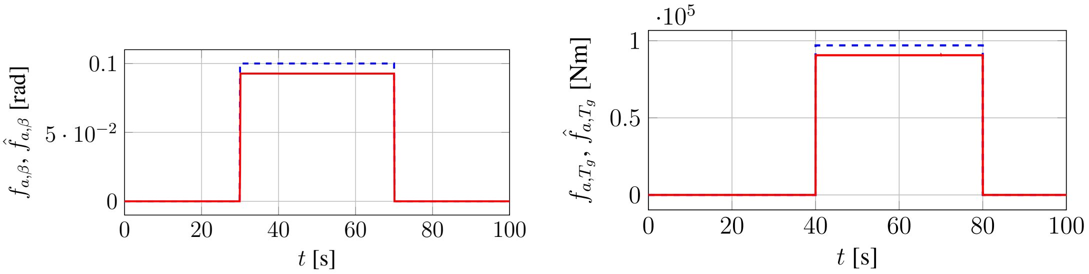


Figure 6.7: Actual and reconstructed actuator faults using the gain matrix (6.61).

choose the elements of $\boldsymbol{\rho}$ according to the first design condition (6.60). However, this will probably not guarantee a satisfying reconstruction of faults. The elements of $\boldsymbol{\rho}$ may then be adjusted until a sufficient reconstruction quality is achieved. The components of $\boldsymbol{\rho}$ must not be too large, either, as this might result in an unstable or at least inadequate observer behaviour.

The question remains whether the reachability condition (6.50) is fulfilled for the above choice of $\boldsymbol{\rho}$. Since $\dot{V}_s = 2\,\mathbf{e}_y^T\,\tilde{\mathbf{P}}_2\,\dot{\mathbf{e}}_y$ and since $\mathbf{e}_y = \mathbf{0}$, $\dot{\mathbf{e}}_y = \mathbf{0}$ during an ideal sliding motion, it holds that $\dot{V}_s = 0$ when a sliding motion has been established. In Figure 6.8, the derivative $\dot{V}_s$ of the Lyapunov function is depicted for different values of the constant δ, where $\dot{V}_s$ was not calculated directly using (6.43) but using (6.45). It is visible that $\dot{V}_s \to 0$ for $\delta \to 0$. This means that a sliding motion is established for most parts of the simulation time except at the single occurrences when the faults are switched on and off. Thus, instead of analysing the reachability condition, it is better to check whether an ideal sliding motion can be obtained with the selected gain matrix $\boldsymbol{\rho}$ by checking whether $\dot{V}_s \to 0$ for $\delta \to 0$.

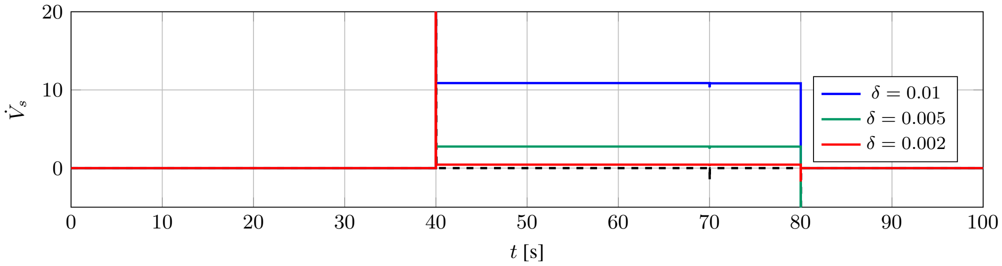


Figure 6.8: Derivatives $\dot{V}_s$ of the Lyapunov function for a simulation with pitch angle and torque actuator faults as described on page 78 for different values of the constant δ. Dashed line: $\dot{V}_s$ calculated with (6.43). Solid lines: derivatives calculated with (6.45).

Apart from $\dot{V}_s$, the choice of δ also influences the quality of the fault reconstruction. This can be seen in Figure 6.9, where the reconstructed pitch angle fault is depicted for values of $\delta = 0.05$ and $\delta = 0.002$. In the latter case, there are only minor differences between the actual and the reconstructed fault, which are hardly visible. However, when reducing δ, one must also reduce the step size in the numerical calculations. Otherwise, an erroneous observer behaviour would result. For the choice of $\delta = 0.05$, the step size was set to $\Delta t = 0.01$ s, whereas for $\delta = 0.002$, the step size had to be reduced to $\Delta t = 0.0005$ s in order to achieve a reliable observer behaviour. Such a small step size would not be appropriate for simulating the observer in real-time on an industrial control system. This means that there is a practical limit for the reconstruction quality, although theoretically each fault could be reconstructed to arbitrary precision.

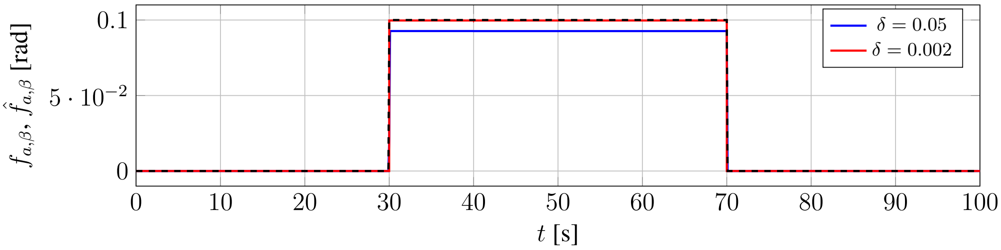


Figure 6.9: Actual and reconstructed pitch angle fault. Black dashed line: actual fault. Blue solid line: Reconstructed fault for $\delta = 0.05$. Red solid line: Reconstructed fault for $\delta = 0.002$.

6.4 Actuator Fault Reconstruction

In this section, some scenarios of actuator faults are reconstructed by means of a TS sliding mode observer. The two main types of faults treated here are input faults and parameter faults (altered actuator dynamics), which require different approaches. Whereas the input faults can be reconstructed directly, the altered actuator dynamics constants, which are inherently available in the additive fault components, have to be extracted using special algebraic relations (see Section 6.4.2).

TS Sliding Mode Observer Configuration

A sliding mode observer of model order 5 is used, with the observer state vector $\mathbf{x} = (\theta_s \quad \omega_r \quad \omega_g \quad \beta \quad T_g)^T$. All states except for the torsional angle θ_s are measurable: $\mathbf{y} = (\omega_r \quad \omega_g \quad \beta \quad T_g)^T$.

The TS SM parameters were chosen as follows:

$\boldsymbol{\mathcal{A}}_{22}^s = \text{diag}\,(-10 \cdots -10)$, $\delta = 0.03$, $\alpha = 0$.

The sliding mode gain matrix was chosen as

$\boldsymbol{\rho} = \text{diag}\begin{pmatrix}200 & 200 & 200 & 6 \cdot 10^8\end{pmatrix}$

and the maximum values for the weighting matrix $\mathbf{W}$ were estimated as:

$\omega_{r,\text{max}} = \omega_{g,\text{max}} = 1.6\,\frac{\text{rad}}{\text{s}}$, $\beta_{\text{max}} = 1.5708\,\text{rad}$, $T_{g,\text{max}} = 4.2 \cdot 10^6\,\text{Nm}$.

Only an actuator fault matrix $\mathbf{F}$, but no $\mathbf{D}$-matrix for unmodelled dynamics is included in the TS sliding mode observer design, as this yielded no improved fault reconstruction.

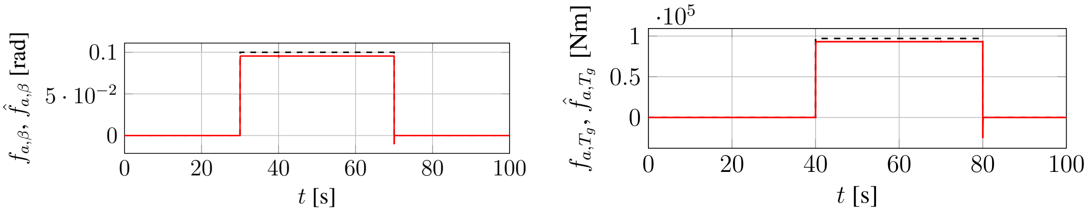
6.4.1 Actuator Faults as Input Faults

The most simple way to consider actuator faults is by adding an offset to the controller output $\mathbf{u}_c$: $\mathbf{u} = \mathbf{u}_c + \mathbf{f}_a$. In this case, there is only one common actuator fault matrix $\mathbf{F}$ in equation (6.1), which is equal to the input matrix $\mathbf{B}$.

Additive Actuator Faults (Offsets)

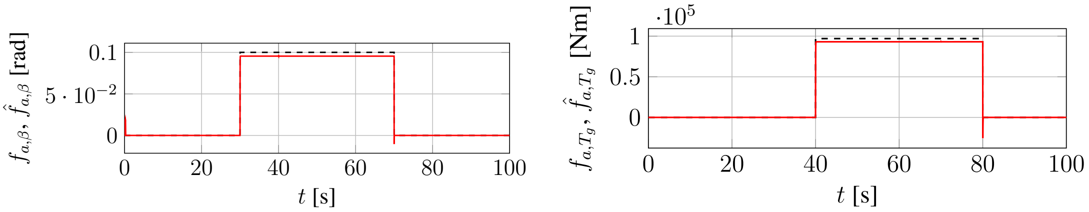
As a first example, the same pitch angle and generator torque offset faults as on page 78 were considered ($f_{a,\beta} = 0.1$ rad, $f_{a,T_g} = 97000$ Nm). The torque offset fault of 97000 Nm in the TS SMO design model without gearbox corresponds to an offset of 1000 Nm in the FAST model. This fault value was taken from the FDI benchmark model [Odgaard and Johnson, 2013].

Simulations were conducted using both the nominal model and FAST. Between 40 s and 70 s, both faults are active. The actual and reconstructed faults are depicted in Figure 6.10.



(a) Pitch fault; simulation with nominal model

(b) Torque fault; simulation with nominal model



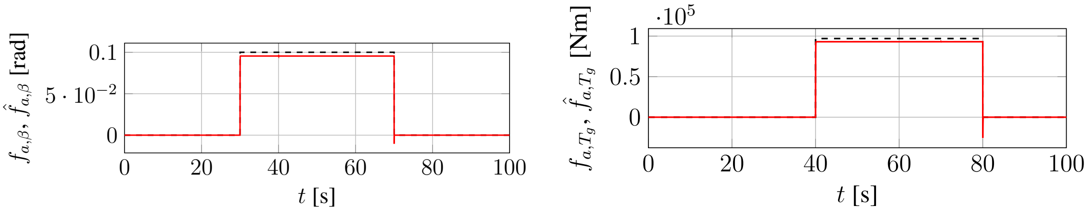
(c) Pitch fault; FAST simulation

(d) Torque fault; FAST simulation

Figure 6.10: Actuator input faults in turbulent wind simulations (mean wind speed: 18 m/s, turbulence intensity: 15 % (NTM), Kaimal model). Black dashed lines: actual faults; red solid lines: reconstructed faults.

It can be seen from Figure 6.10 that both faults are reasonably well reconstructed. Astonishingly, (almost) identical results are obtained in both simulation models, apart from a small initial peak in the reconstruction of the pitch angle fault in the FAST simulation. Compared to the simulation results in Figure 6.7, the discrepancy between the actual and the reconstructed fault values is smaller, which is due to the smaller sliding mode parameter $\delta = 0.03$ compared to $\delta = 0.05$ in the previous section. However, since the simulation time-step was left unaltered ($\Delta t = 0.01$ s), a small negative peak is visible when the faults are switched off, which was not visible in the simulation results in Figure 6.7. This further reveals the compromise that must be made between reconstruction accuracy and a feasible simulation time-step.

The same fault scenarios were simulated in the partial load region, at a mean wind speed of 10 m/s (see Figure 6.11), using FAST. Again, the same reconstruction quality as in the case of 18 m/s is achieved.



(a) Actual and reconstructed pitch angle fault (b) Actual and reconstructed generator torque fault

Figure 6.11: Actuator input faults in a FAST simulation with turbulent wind input (mean wind speed: 10 m/s, turbulence intensity: 18 % (NTM), Kaimal model). Black dashed lines: actual faults; red solid lines: reconstructed faults.

Scaling Actuator Faults

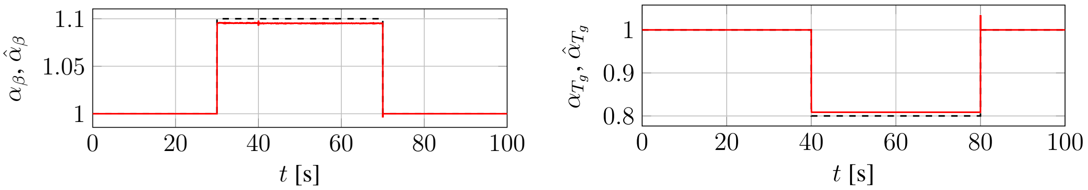
Scaling faults can also be modelled and detected using the additive structure $\mathbf{F}_i\,\mathbf{f}_a$ in the TS sliding mode observer design. The scaled actuator signal $\tilde{u} = \alpha\, u$ can be written using the additive fault signal f_a:

$$\tilde{u} \;=\; \alpha\, u \;=\; u \;+\; f_a\,, \tag{6.62}$$

and thus the scaling constant α can be reconstructed using the reconstructed additive fault signal $\hat{\mathbf{f}}_a$:

$$\hat{\alpha} \;=\; \frac{u + \hat{f}_a}{u}\,. \tag{6.63}$$

A pitch angle (scaling constant: 1.1, active between 30 s and 70 s) and a generator torque scaling fault (scaling constant: 0.8, active between 40 s and 80 s) were simulated in FAST with turbulent wind input. The scaling constants, which are both reconstructed fast and precise, are depicted in figure 6.12.



(a) Actual and reconstructed pitch angle scaling constant (b) Actual and reconstructed generator torque scaling constant

Figure 6.12: Actuator input faults in FAST simulations with turbulent wind input (mean wind speed: 18 m/s, turbulence intensity: 15 % (NTM), Kaimal model). Black dashed lines: actual (faulty) scaling constants; red solid lines: reconstructed scaling constants.

Apart from scaling faults, other fault types like fixed values, or stuck faults can also traced back to the additive fault structure (see Section 6.5 and [Chen and Patton, 1999; Sami and Patton, 2012c]).

6.4.2 Altered Actuator Dynamics

Input faults are one special fault category. However, since the pitch system and the generator/converter system are complex, inherent faults in these systems may also alter the pitch and converter dynamics, which in this work are represented by the first-order delay models (6.64) and (6.65).

$$\begin{aligned} \dot{\beta} &= -\frac{1}{\tau}\beta + \frac{1}{\tau}\beta_d\,, & (6.64)\\ \dot{T}_g &= -\frac{1}{\tau_g}T_g + \frac{1}{\tau_g}T_{g,d}\,. & (6.65) \end{aligned}$$

Alterations in the dynamics of the actuator delay models will be reflected in changed time delay constants. In such a case, the goal of a fault reconstruction scheme would be to reconstruct the faulty time delay constant.

Consider a change of the pitch delay time constant: $\tau \to \tilde{\tau}$. Since the delay time constant appears in the denominator in the actuator model (6.64), an altered constant is best treated by considering its reciprocal value: $\tilde{a} := \frac{1}{\tilde{\tau}}$. Then, the alteration can be represented as the sum of the nominal value $a = \frac{1}{\tau}$ and an offset value Δa: $\tilde{a} = a + \Delta a$.

The altered dynamics is then given by

$$\begin{aligned} \dot{\beta} &= -\tilde{a}\,\beta + \tilde{a}\,\beta_d \\ &= -(a+\Delta a)\,\beta + (a+\Delta a)\,\beta_d \\ &= -a\,\beta + a\,\beta_d - \Delta a\,\beta + \Delta a\,\beta_d\,. & (6.66) \end{aligned}$$

Analogously, altered generator torque dynamics can be considered using the reciprocal a_g of the delay time constant τ_g:

$$\dot{T}_g = -a_g\,T_g + a_g\,T_{g,d} - \Delta a_g\,T_g + \Delta a_g\,T_{g,d}\,. \qquad (6.67)$$

Both offset values are time-dependent: $\Delta a = \Delta a\,(t)$, $\Delta a_g = \Delta a_g\,(t)$

Defining the offset matrices $\Delta\mathbf{A}$ and $\Delta\mathbf{B}$ as

$$\Delta\mathbf{A} = \begin{pmatrix} 0 & \cdots & 0 & 0 \\ & \ddots & & \\ 0 & \cdots & -\Delta a & 0 \\ 0 & \cdots & 0 & -\Delta a_g \end{pmatrix} \qquad \Delta\mathbf{B} = \begin{pmatrix} 0 & 0 \\ \vdots & \vdots \\ \Delta a & 0 \\ 0 & \Delta a_g \end{pmatrix}, \qquad (6.68)$$

the system dynamics with altered actuator dynamics can be written as

$$\dot{\mathbf{x}} = \sum_{i=1}^{N_r} h_i\,(\mathbf{z})\,(\mathbf{A}_i\mathbf{x} + \mathbf{D}_i\,\boldsymbol{\xi}) + \mathbf{Bu} + \Delta\mathbf{A}\,\mathbf{x} + \Delta\,\mathbf{B}\,\mathbf{u}\,. \qquad (6.69)$$

In order to reconstruct the altered actuator dynamics parameters Δa and Δa_g, a fault matrix $\mathbf{F}$ and an actuator fault vector $\mathbf{f}_a$ must be found such that

$$\begin{aligned} \mathbf{F}\,\mathbf{f}_a &= \Delta\mathbf{A}\,\mathbf{x} + \Delta\,\mathbf{B}\,\mathbf{u} \\ &= \begin{pmatrix} 0 \\ \vdots \\ 0 \\ \Delta a\ (\beta_d - \beta) \\ \Delta a_g\ (T_{g,d} - T_g) \end{pmatrix} . \end{aligned} \tag{6.70}$$

Choosing

$$\mathbf{F} = \begin{pmatrix} 0 & 0 \\ \vdots & \vdots \\ 1 & 0 \\ 0 & 1 \end{pmatrix} \qquad \text{and} \qquad \mathbf{f}_a = \begin{pmatrix} \Delta a\ (\beta_d - \beta) \\ \Delta a_g\ (T_{g,d} - T_g) \end{pmatrix} , \tag{6.71}$$

condition (6.70) is fulfilled. Furthermore, in the case when no unmodelled dynamics are considered ($\mathbf{D} = \mathbf{0}$), this choice of $\mathbf{F}$ guarantees that the existence conditions for the TS sliding mode observer are fulfilled:

1. The number of measurable states p must be greater than q, the number of columns of $\mathbf{F}$. This holds, because at least ω_r, ω_g, β and T_g are measurable.
2. $q = \text{rank}\,(\mathbf{F})$.

The reconstructed actuator fault vector is thus given by

$$\hat{\mathbf{f}}_a = \begin{pmatrix} \Delta\hat{a}\ (\beta_d - \beta) \\ \Delta\hat{a}_g\ (T_{g,d} - T_g) \end{pmatrix} , \tag{6.72}$$

and is reconstructed using the equivalent output injection signal $\boldsymbol{\nu}_{\text{eq}}$ via the relation $\hat{\mathbf{f}}_a = \boldsymbol{\mathcal{F}}_2\,(\mathbf{z})^+\ \boldsymbol{\nu}_{\text{eq}}$. such that the individual reconstructed altered actuator dynamics parameters can be obtained as

$$\Delta\hat{a} = \begin{cases} \frac{\hat{f}_{a,1}}{\beta_d - \beta} & , \beta_d \neq \beta \\ 0 & , \beta_d = \beta \end{cases} \tag{6.73}$$

$$\Delta\hat{a}_g = \begin{cases} \frac{\hat{f}_{a,2}}{T_{g,d} - T_g} & , T_{g,d} \neq T_g \\ 0 & , T_{g,d} = T_g \end{cases} \tag{6.74}$$

In order to avoid large peaks in the reconstruction of τ and τ_g in case the differences between the demanded and actual actuator signals are close to zero, the parameters $\hat{\Delta}a$ and $\hat{\Delta}a_g$ are only calculated using equations (6.73) and (6.74) if $|\beta_d - \beta| > \varepsilon_{\beta,\beta_d}$ and $|T_{g,d} - T_g| > \varepsilon_{T_g,T_{g,d}}$, with small positive constants $\varepsilon_{\beta,\beta_d}$ and $\varepsilon_{T_g,T_{g,d}}$. Otherwise, $\Delta\hat{a}$ and $\Delta\hat{a}_g$ are fixed at their respective values of the previous timestep.
If $\hat{f}_{a,1}$ assumes 0 for several cycles[4] (and not only when oscillating around it), there is no pitch dynamics fault present. In this case, the fault parameter $\Delta\hat{a}_1$ is set to zero. Equally, if $\hat{f}_{a,2}$ is 0 for several cycles, $\Delta\hat{a}_g$ is set to zero. In practice, these conditions are checked by defining small constants ε_1 and ε_g and by checking whether $\left|\hat{f}_{a,1}\right| < \varepsilon_1$ and $\left|\hat{f}_{a,g}\right| < \varepsilon_g$.

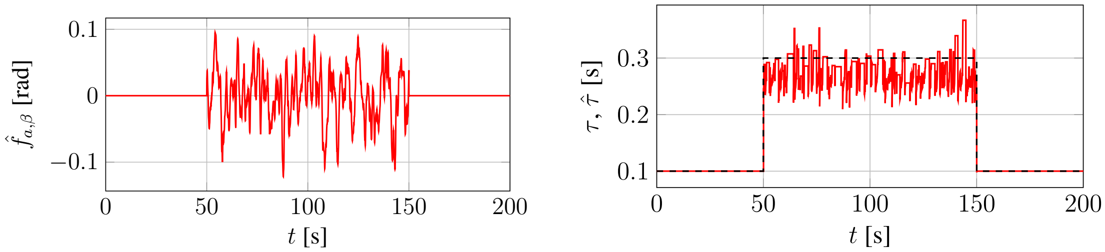
---

[4] the number of cycles depends on the timescale of the dynamics and on the stepsize of the numerical integration

Also, the faulty delay time constants can be calculated as $\tilde{\tau} = \frac{1}{\tilde{a}} = \frac{1}{a+\Delta\hat{a}}$ and $\tilde{\tau}_g = \frac{1}{\tilde{a}_g} = \frac{1}{a_g+\Delta\hat{a}_g}$.

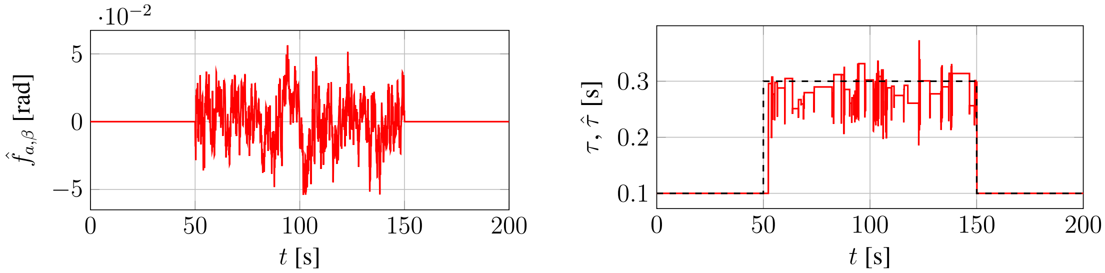
As an example for altered pitch actuator dynamics, simulations both of the nominal model and FAST were conducted with turbulent wind input (mean wind speed: 18 m/s) and a fault in the pitch actuator dynamics active between 50 s and 150 s (step of the pitch actuator delay time constant from 0.1 s to 0.3 s).

The results are depicted in Figure 6.13, where the upper two figures depict the results of the simulations using the nominal model and the lower two figures depict the FAST simulation results.



(a) Reconstructed fault component. Simulation with nominal model

(b) Actual and reconstructed delay time constant. Simulation with nominal model



(c) Reconstructed fault component. FAST simulation

(d) Actual and reconstructed delay time constant. FAST simulation

Figure 6.13: Pitch parameter fault. Turbulent wind input (mean wind speed: 18 m/s, turbulence intensity: 15 % (NTM, Kaimal model)).

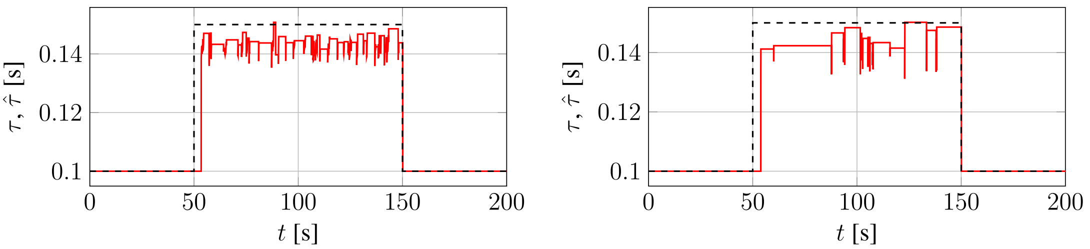
It can be seen from Figures 6.13b and 6.13d that the reconstruction quality of the faulty delay time constant is inferior to those achieved for the actuator input faults in Section 6.4.1. The quality of the reconstructed signal is not sufficient to be used in a fault-tolerant control scheme. However, the signal could be used for a fast fault detection scheme, where a critical signal height could be defined to decide whether or not a fault has occured. The time-series of $\hat{\tau}$ are interrupted with peaks, which occur whenever the difference $\beta_d - \beta$ is close to $\varepsilon_{\beta,\beta_d}$ which was set to 0.005 rad in both simulations. A possible remedy to obtain a smoother, but also slower, reconstruction of τ is to filter the reconstructed signal. The constant ε_1 was set to $\varepsilon_1 = 10^{-6}\,\frac{\text{rad}}{\text{s}}$.

In the FAST simulation, the reconstruction of τ is comparable to the case of using the nominal model, except that the reaction time is longer once the fault has occurred (see Figure 6.13d).

For comparison, two simulations were conducted with a lower fault amplitude. Between 50 s and 150 s, the faulty delay time constant was set to 0.15 s. The reconstructed constants are depicted in Figure 6.14.

It can be seen from Figure 6.14 that the reconstruction quality of the delay time constants is slightly better compared to the case in Figure 6.13, where a higher fault magnitude was set.

A torque actuator dynamics fault was also simulated using both the nominal model and FAST. A step fault of the delay time constant τ_g (step from 0.02 s to 0.1 s) was active between 40 s and 160 s. The reconstructed delay time constant is depicted in Figures 6.15b and 6.15d. The constant $\varepsilon_{T_g,T_{g,d}}$ was set to 10000 Nm in the case of the nominal model simulation and 5000 Nm in the FAST simulation. These values are still relatively small when set in relation to the order of magnitude of T_g, which is in the region

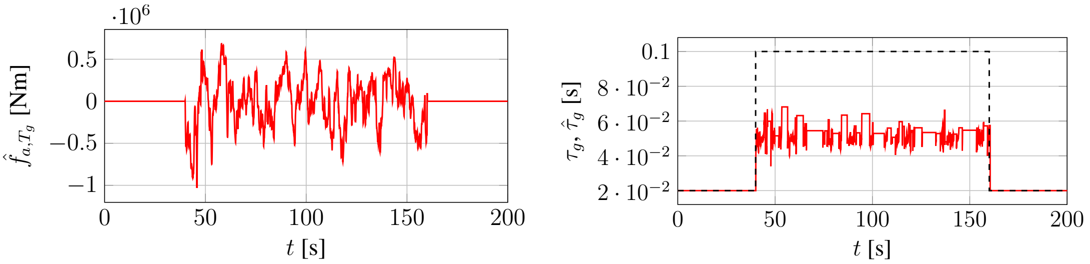


(a) Simulation with nominal model

(b) FAST simulation

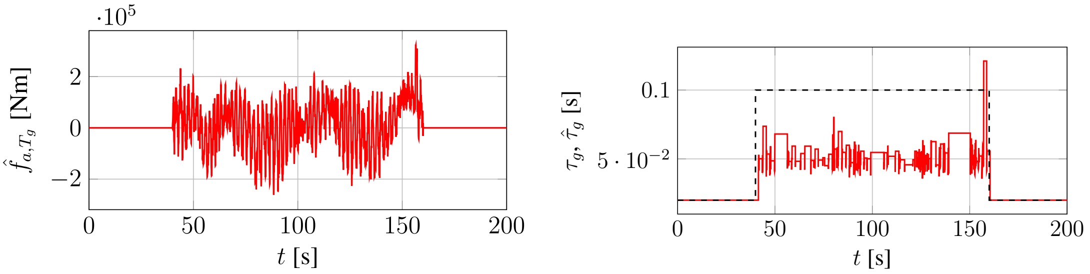
Figure 6.14: Pitch parameter fault (lower fault amplitude). Actual and reconstructed delay time constant. Turbulent wind input (mean wind speed: 18 m/s, turbulence intensity: 15 % (NTM, Kaimal model)).

of 10^6 Nm. The constant ε_g was set to $\varepsilon_g = 10^{-4}\,\frac{\text{Nm}}{\text{s}}$.



(a) Reconstructed fault component. Simulation with nominal model

(b) Actual and reconstructed delay time constant. Simulation with nominal model



(c) Reconstructed fault component. FAST simulation

(d) Actual and reconstructed delay time constant. FAST simulation

Figure 6.15: Torque parameter fault. Turbulent wind input (mean wind speed: 8 m/s, turbulence intensity: 20 % (NTM, Kaimal model)).

The reconstruction results are similar in both simulations, although one has to keep in mind that different values for the constant $\varepsilon_{T_g,T_{g,d}}$ were used. As in the example of the pitch parameter fault, the quality of the reconstruction if not sufficient for using the signal in fault-tolerant control applications but only for fault detection schemes.

Altered Actuator Dynamics for Second Order Pitch Model

In the previous section, altered pitch actuator dynamics were only considered for a first order delay model for the pitch system. In this section, it is investigated whether altered pitch dynamics can also be reconstructed for a second order delay model. In this case, the pitch dynamics is given by

$$\begin{pmatrix}\dot{\beta}\\ \ddot{\beta}\end{pmatrix} = \begin{pmatrix}0 & 1\\ -\omega_n^2 & -2\,\zeta\,\omega_n\end{pmatrix}\begin{pmatrix}\beta\\ \dot{\beta}\end{pmatrix} + \begin{pmatrix}0\\ \omega_n^2\end{pmatrix}\beta_d\,, \tag{6.75}$$

where ω_n are ζ are the characteristic frequency and the damping of the second order delay model, respectively. The (measurable) pitch rate $\dot{\beta}$ is then included in the state vector and in the output vector of both the simulation model and of the TS SMO.

Due to the nonlinear terms ω_n^2 and $-2\,\zeta\,\omega$, the modelling of the altered actuator dynamics for the second order pitch model is not as straightforward as for the first order delay model. As a first step, it is advantageous to substitute the nonlinear terms by linear parameters, as follows:

$$a_1 := \omega_n^2\,, \qquad a_2 := 2\,\zeta\,\omega_n\,. \tag{6.76}$$

An alteration in actuator dynamics (parameters ω_n and ζ) can then be formulated as an additive offset on the substituted dynamics parameters:

$$a_1 \;\rightarrow\; \tilde{a}_1 \;=\; a_1 + \Delta a_1\,, \tag{6.77}$$

$$a_2 \;\rightarrow\; \tilde{a}_2 \;=\; a_2 + \Delta a_2\,. \tag{6.78}$$

The altered pitch dynamics is then given by

$$\begin{pmatrix}\dot{\beta}\\ \ddot{\beta}\end{pmatrix} = \begin{pmatrix}0 & 1\\ -a_1 & -a_2\end{pmatrix}\begin{pmatrix}\beta\\ \dot{\beta}\end{pmatrix} + \begin{pmatrix}0\\ a_1\end{pmatrix}\beta_d + \begin{pmatrix}0 & 1\\ -\Delta a_1 & -\Delta a_2\end{pmatrix}\begin{pmatrix}\beta\\ \dot{\beta}\end{pmatrix} + \begin{pmatrix}0\\ \Delta a_1\end{pmatrix}\beta_d\,. \tag{6.79}$$

Analogously to the case of first order pitch dynamics, offset matrices $\Delta\mathbf{A}$ and $\Delta\mathbf{B}$ can be defined as

$$\Delta\mathbf{A} = \begin{pmatrix}0 & \cdots & 0 & 0 & 0\\ & \ddots & & 0 & 0\\ 0 & \cdots & 0 & 0 & 0\\ 0 & \cdots & -\Delta a_1 & -\Delta a_2 & 0\\ 0 & \cdots & 0 & 0 & -\Delta a_g\end{pmatrix} \qquad \Delta\mathbf{B} = \begin{pmatrix}0 & 0\\ \vdots & \vdots\\ 0 & 0\\ \Delta a_1 & 0\\ 0 & \Delta a_g\end{pmatrix}, \tag{6.80}$$

such that the dynamics of the observer system (without feedback terms) with altered actuator dynamics can be written as [5]

$$\dot{\mathbf{x}} = \sum_{i=1}^{N_r} h_i\,(\mathbf{z})\,(\mathbf{A}_i\mathbf{x} + \mathbf{D}_i\,\boldsymbol{\xi}) + \mathbf{B}\mathbf{u} + \Delta\mathbf{A}\,\mathbf{x} + \Delta\,\mathbf{B}\,\mathbf{u}\,. \tag{6.81}$$

The actuator dynamics fault term $(\Delta\mathbf{A}\,\mathbf{x} + \Delta\,\mathbf{B}\,\mathbf{u})$ is given by

---

[5] the torque actuator dynamics are again treated as a first order delay model

$$\Delta\mathbf{A}\,\mathbf{x} + \Delta\,\mathbf{B}\,\mathbf{u} \;=\; \begin{pmatrix} 0 \\ \vdots \\ 0 \\ \Delta a_1\,(\beta_d - \beta) - \Delta a_2\,\dot{\beta} \\ \Delta a_g\,(T_{g,d} - T_g) \end{pmatrix}, \tag{6.82}$$

and can be written as $\Delta\,\mathbf{A}\,\mathbf{x} + \Delta\,\mathbf{B}\,\mathbf{u} = \mathbf{F}\,\mathbf{f}_a$ by choosing

$$\mathbf{F} = \begin{pmatrix} 0 & 0 \\ \vdots & \vdots \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{pmatrix} \qquad \text{and} \qquad \mathbf{f}_a = \begin{pmatrix} \Delta a_1\,(\beta_d - \beta) \;-\; \Delta a_2\,\dot{\beta} \\ \Delta a_g\,(T_{g,d} - T_g) \end{pmatrix}. \tag{6.83}$$

The torque actuator fault Δa_g can be reconstructed as described in the previous section. For the pitch actuator faults Δa_1 and Δa_2, however, there is now only one equation available:
$f_{a,1} = \Delta a_1\,(\beta_d - \beta) - \Delta a_2\,\dot{\beta}$. This equation is under-determined, with the result that Δa_1 and Δa_2 cannot be directly reconstructed from it.

A possible way to proceed is to exploit the fact that the pitch rate $\dot{\beta}$ oscillates around zero in full-load operation of the wind turbine, so that there are occurrences when $\dot{\beta} = 0$ and thus $f_{a,1} = \Delta a_1\,(\beta_d - \beta)$. Consequently, at these single-point occurrences, $\Delta\hat{a}_1$ can be reconstructed as

$$\Delta\hat{a}_1 \;=\; \frac{\hat{f}_{a,1}}{\beta_d - \beta} \qquad \left(\dot{\beta} = 0\,, \quad \beta_d \neq \beta\right). \tag{6.84}$$

Then, keeping track of the reconstructed values of $\Delta\hat{a}_1$, for example by calculating a running average $\Delta\hat{a}_{1,\text{mean}}$, the second dynamics fault parameter Δa_2 can be reconstructed (when $\dot{\beta} \neq 0$) as

$$\Delta\hat{a}_2 \;=\; \frac{1}{\dot{\beta}}\left(\Delta\hat{a}_{1,\text{mean}}\,(\beta_d - \beta) \;-\; \hat{f}_{a,1}\right) \qquad \left(\dot{\beta} \neq 0\right). \tag{6.85}$$

In order to exclude large peaks in the reconstruction of $\Delta\hat{a}_2$ when $\dot{\beta}$ is close to zero, a threshold $\varepsilon_{\dot{\beta}} > 0$ can be defined and equation (6.85) is only calculated when $|\dot{\beta}| > \varepsilon_{\dot{\beta}}$. Otherwise, $\Delta\hat{a}_2$ is fixed to its value of the previous timestep.

Similarly to the fault detection algorithm for first order pitch dynamics, if $\hat{f}_{a,1}$ assumes 0 for several cycles (and not only when oscillating around it), there is no pitch dynamics fault present. In this case, the fault parameter $\Delta\hat{a}_1$ is set to zero.

To check whether $\hat{f}_{a,1} \approx 0$ and $\dot{\beta} \approx 0$ is best done by specifying small constants ε_1, ε_2 close to zero and testing whether $\left|\hat{f}_{a,1}\right| < \varepsilon_1$ and $|\dot{\beta}| < \varepsilon_2$.

From the definitions (6.76), (6.77) and (6.78), the reconstructed second order delay model parameters $\hat{\omega}_n$ and $\hat{\zeta}$ can be calculated from the reconstructed values of $\Delta\hat{a}_{1,\text{mean}}$ and $\Delta\hat{a}_2$ as

$$\hat{\omega}_n \;=\; \sqrt{\Delta\hat{a}_{1,\text{mean}} + \omega_n^2}\,, \qquad \hat{\zeta} \;=\; \frac{1}{\hat{\omega}_n}\left(\frac{\Delta\hat{a}_2}{2} + \zeta\,\omega_n\right). \tag{6.86}$$

Example for Second Order Pitch Dynamics Fault

As an example for a fault in the pitch actuator dynamics using a second order pitch model, one fault from the FDI benchmark model for FAST [Odgaard and Johnson, 2013] is used. The default parameters for the pitch model are: $\omega_n = 11.11 \, \frac{\text{rad}}{\text{s}}$ and $\zeta = 0.6$. One fault scenario described in [Odgaard et al., 2009] is a pressure drop in the hydraulic pitch system, with the following faulty parameters: $\tilde{\omega}_n = 5.73 \, \frac{\text{rad}}{\text{s}}$, $\tilde{\zeta} = 0.45$.

A fault in the torque actuator dynamics is not considered here, such that $\mathbf{F}$ in (6.83) consists only of one column and $\mathbf{f}_a$ is reduced to a scalar $f_a = f_{a,1} = \Delta a_1 \, (\beta_d - \beta) - \Delta a_2 \, \dot{\beta}$.

The same sliding mode observer as in Section 6.4 is used here for the fault reconstruction, with an enlarged system vector now including the pitch rate: $\mathbf{x} = \begin{pmatrix} \theta_s & \omega_r & \omega_g & \beta & \dot{\beta} & T_g \end{pmatrix}^T$. All states except for the torsional angle θ_s are measurable: $\mathbf{y} = \begin{pmatrix} \omega_r & \omega_g & \beta & \dot{\beta} & T_g \end{pmatrix}^T$.

The small threshold values were set to $\varepsilon_{\dot{\beta}} = 0.04 \, \frac{\text{rad}}{\text{s}}$, $\varepsilon_1 = 10^{-6} \, \frac{\text{rad}}{\text{s}^2}$ and $\varepsilon_2 = 10^{-4} \, \frac{\text{rad}}{\text{s}}$.

A simulation was conducted using the nominal model, with the pitch actuator fault active between 50 s and 150 s. Figure 6.16a shows the reconstructed fault signal $\hat{f}_{a,1} = \Delta \hat{a}_1 \, (\beta_d - \beta) - \Delta \hat{a}_2 \, \dot{\beta}$, which is calculated from the equivalent output injection signal $\boldsymbol{\nu}_{\text{eq}}$ using the relation $\hat{\mathbf{f}}_a = \boldsymbol{\mathcal{F}}_2 \, (\mathbf{z})^+ \, \boldsymbol{\nu}_{\text{eq}}$. While the fault is active, $\hat{f}_{a,1}$ oscillates around zero, whereas it is equal to zero when no fault is active. $\hat{f}_{a,1}$ implicitly contains both faulty second order parameters ω_n and ζ.

Figure 6.16b shows the pointwise reconstruction of the fault parameter $\Delta \hat{a}_1$, at occurrences when $\dot{\beta} \approx 0$. At these instances, sharp negative peaks are visible in the figure.

From the running average of the reconstructed values of $\Delta \hat{a}_1$, $\Delta \hat{a}_2$ is calculated at instances when $|\dot{\beta}| > \varepsilon_{\dot{\beta}}$. The time-series of $\Delta \hat{a}_2$ is displayed in Figure 6.16c.

Using the equations in (6.86), the reconstructed dynamics parameters $\hat{\omega}_n$ and $\hat{\zeta}$ are calculated and displayed in Figures 6.16d and 6.16e.

As can be seen in Figure 6.16d, a rough but still reasonable reconstruction of the frequency parameter ω_n can be achieved. The time-shift of > 15 s is due to some memory blocks that were included in the Simulink® model to avoid algebraic loops. When the memory blocks are removed, the time-shift can be significantly reduced (see Figure 6.17a).[6] The reconstruction of the damping parameter ζ, on the other hand, is of inferior quality and probably not even appropriate to be used in fault detection schemes.

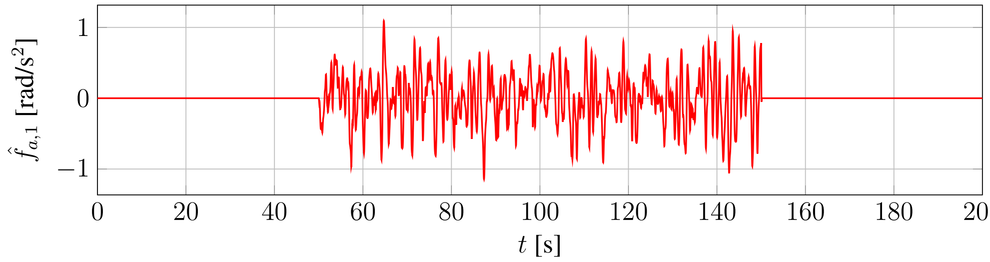
As a general rule, both for the first and for the second order delay models, the reconstruction of the dynamics parameters is strongly influenced by the choice of the ε-constants. Increasing the constants $\varepsilon_{\beta,\beta_d}$, $\varepsilon_{T_g \, T_{g,d}}$ and $\varepsilon_{\dot{\beta}}$ leads to a less noisy, but also more discretised and thus slower reconstruction. The fault reconstruction for parameter faults is thus not yet very robust, neither with respect to the ϵ-parameters nor with respect to model uncertainties. Still, the results presented in this section are interesting and demonstrate the breadth of possible applications of the TS sliding mode observer.

The main part of Section 6.4.2 was first published in [Georg and Schulte, 2014a] [7].

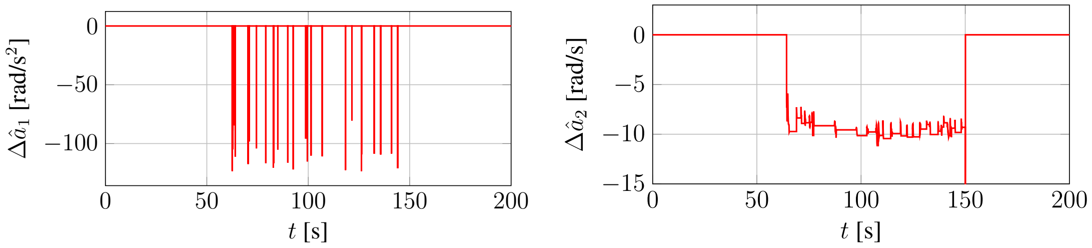
---

[6] The problem of algebraic loops is peculiar to Simulink®. When implementing the observers on a wind turbine control system one could thus expect to get a similar time-shift as in Figure 6.17a.

[7] Sören Georg and Horst Schulte. Diagnosis of Actuator Parameter Faults in Wind Turbines using a Takagi-Sugeno Sliding Mode Observer. In Józef Korbicz and Marek Kowal, editors, *Intelligent Systems in Technical and Medical Diagnostics*, volume 230 of *Advances in Intelligent Systems and Computing*, pages 29-40. © Springer-Verlag Berlin Heidelberg 2014, with kind permission from Springer Science and Business Media.

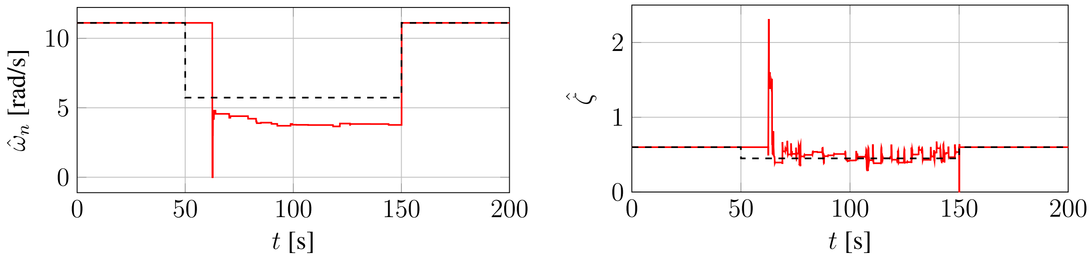


(a) Reconstructed fault signal $\hat{f}_{a,1}$



(b) Reconstructed fault parameter $\Delta\hat{a}_1$. Only at instances when $\dot{\beta} = 0$.

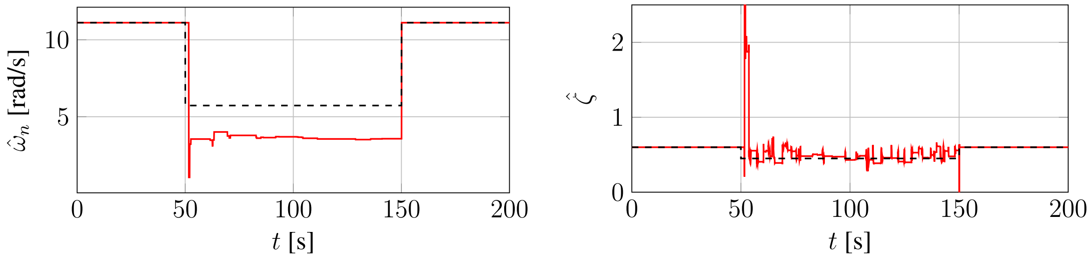
(c) Reconstructed fault parameter $\Delta\hat{a}_2$. Only when $|\dot{\beta}| > \varepsilon_{\dot{\beta}}$.



(d) Actual and reconstructed frequency parameter

(e) Actual and reconstructed damping parameter

Figure 6.16: Pitch parameter fault for second order delay model. Simulation with nominal model. Turbulent wind input (mean wind speed: 18 m/s, turbulence intensity: 15 % (NTM, Kaimal model)).



(a) Actual and reconstructed frequency parameter

(b) Actual and reconstructed damping parameter

Figure 6.17: Pitch parameter fault for second order delay model. Simulation with nominal model. Turbulent wind input (mean wind speed: 18 m/s, turbulence intensity: 15 % (NTM, Kaimal model)). Some memory blocks were removed in the Simulink model.

6.5 Sensor Fault Reconstruction

In this section, the dynamic sensor fault reconstruction method from [Tan and Edwards, 2003; Alwi et al., 2011] (see Section 6.2) is used to reconstruct some example sensor faults in simulations of both the nominal model and FAST. The same TS sliding mode observer as in Section 6.4 was used. The sensor fault selection matrix was chosen as $\mathbf{N} = \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$, which proved to be an appropriate choice. Other structures were tested, as well. In some cases, this led to problems in the simulation, sometimes not even a TS SMO design was feasible. The above choice of $\mathbf{N}$ means that the sensor fault vector is given by $\mathbf{f}_s = \begin{pmatrix} f_{s,\omega_g} & f_{s,\beta} & f_{s,T_g} \end{pmatrix}^T$, such that these three sensor faults can, in principle, be simultaneously reconstructed.

Notation Here and in the following, a subindex $_F$ denotes a faulty sensor value, for example, $\omega_{g,F}$ denotes a faulty generator speed sensor value.

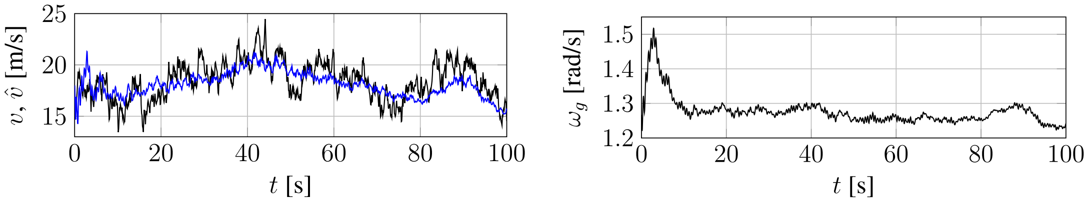
6.5.1 Simulation Examples

Additive Sensor Faults

As a first example, offset faults of the pitch angle and generator torque sensors were simulated. For the pitch angle, the offset fault was set to 0.1 rad. The generator torque offset fault was set to 97000 Nm. Since a gearbox ratio is not included in the observer design model, this fault corresponds to a torque offset of 1000 Nm in the FAST model, were a gearbox ratio of $n_g = 97$ is included.

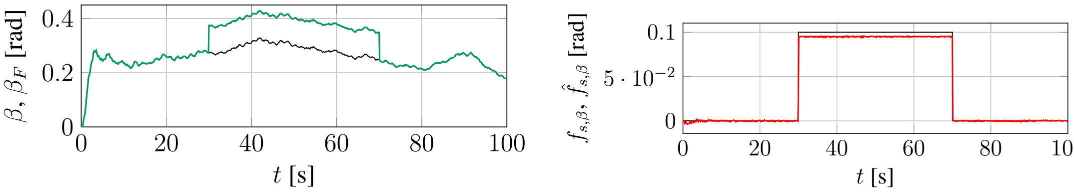
The simulation results are depicted in Figure 6.18.

It can be seen that both sensor faults are reconstructed fast. The absolute heights of the faults are reached to a degree of about 93 %. As discussed in Section 6.3.3, by choosing smaller values of δ and the time-step Δt, the correct fault heights could be reconstructed to any degree of accuracy. Even though no fault is active in the generator speed sensor, an erroneous fault signal is reconstructed in this channel (see Figure 6.18g). This effect, which is not limited to the FAST simulation but also occurs in the simulation of the nominal model, should be identified as an erroneous fault detection in an FDI decision algorithm.



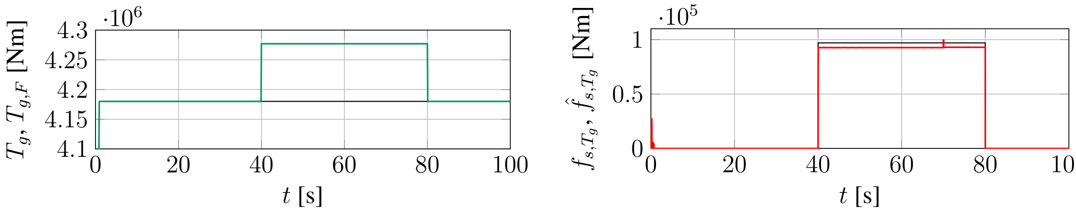
(a) Black line: FAST wind speed output; blue line: estimated wind speed with TS observer

(b) Generator Speed



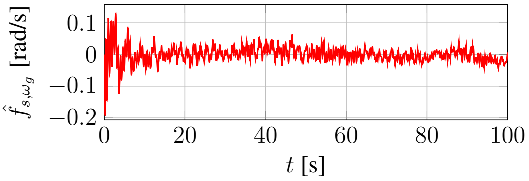
(c) Black line: physical pitch angle;
green line: (faulty) pitch angle sensor output

(d) Black line: actual pitch angle sensor fault; red line: reconstructed pitch angle sensor fault



(e) Black line: physical generator torque;
green line: (faulty) torque sensor output

(f) Black line: actual generator torque sensor fault; red line: reconstructed torque sensor fault



(g) (Erroneously) reconstructed generator speed sensor fault

Figure 6.18: FAST simulation with turbulent wind input (mean wind speed: 18 m/s, turbulence intensity: 15 % (NTM), Kaimal model) with offset faults of the pitch angle (0.1 rad) and generator torque (97000 Nm) sensors

Scaling Faults

For the detection of fault types other than additive offset faults, one can exploit the fact that several types of faults can be represented as additive faults [Chen and Patton, 1999; Sami and Patton, 2012c].

For a scaling fault with a scaling constant α, the faulty output signal y_F can be written as $y_F = \alpha\, y$. The sensor fault reconstruction with the TS sliding mode observer using relation (6.34) assumes an additive sensor fault component $f_s = y_F - y$. A scaling fault must therefore be represented in additive form to obtain an estimate for the scaling constant α using the reconstructed value $\hat{f}_s$:

$$\begin{aligned} y_F = \alpha\, y &\overset{!}{=} y + f_s \\ &= y + \alpha\, y - y \\ &= y + \underbrace{(\alpha - 1)\, y}_{= f_s} \end{aligned} \tag{6.87}$$

$$\begin{aligned} \Rightarrow \quad f_s = y_F - y &\overset{!}{=} (\alpha - 1)\, y \\ &= (\alpha - 1)\,(y_F - f_s) \end{aligned} \tag{6.88}$$

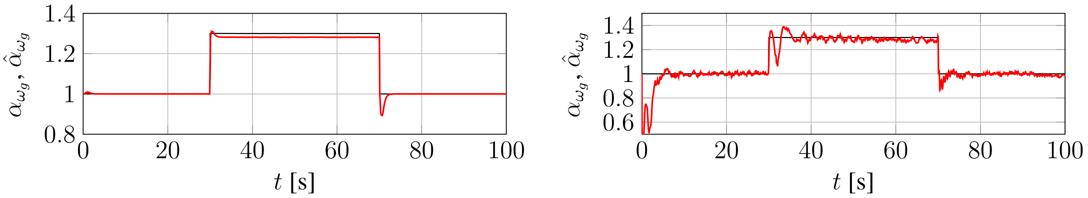
Using again the relation $y = y_F - f_s$, equation (6.88) can be rearranged to

$$\alpha = \frac{y_F}{y_F - f_s} = \frac{y + f_s}{y} \,. \tag{6.89}$$

From the reconstructed additive sensor fault component $\hat{f}_s$, an estimate $\hat{\alpha}$ for the scaling constant of the faulty measurement can be calculated using

$$\hat{\alpha} = \frac{y_F}{y_F - \hat{f}_s} = \frac{y + \hat{f}_s}{y} \,. \tag{6.90}$$

As an example, a scaling fault of the generator speed sensor (faulty scaling constant of 1.3 between 30 s and 70 s) was simulated using both the nominal model and FAST, with the same wind speed characteristics as in the examples for the offset faults on page 99. Here, the generator speed was used as controlled variable instead of the rotor speed, to get a realistic estimation of the effect of the sensor fault. The reconstructed generator speed sensor fault scaling constants are depicted in Figure 6.19.



(a) Simulation with nominal model

(b) FAST simulation

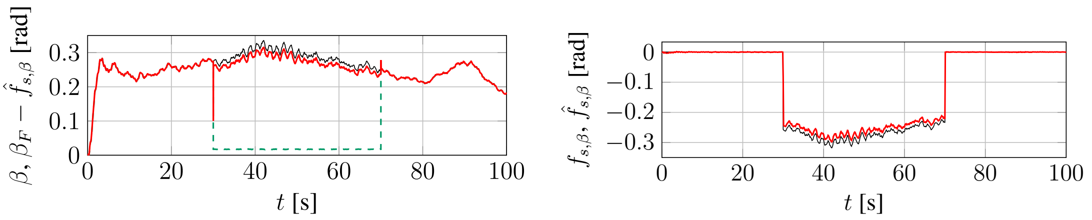
Figure 6.19: Actual and reconstructed generator speed sensor fault scaling constant for a turbulent wind simulation (mean wind speed: 18 m/s) with a generator speed sensor scaling fault.

The fault reconstruction in the nominal model simulation is comparably fast and precise. However, in the FAST simulation, the reconstruction quality is significantly impaired, in contrast to the results obtained in the simulations of the offsets faults. This indicates that the reconstruction quality can depend sensitively both on the type of fault and the respective fault channel.

Faults with Fixed Sensor Values

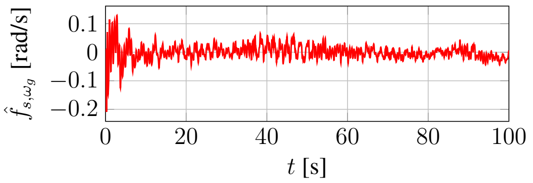
One of the sensor faults given in the FDI benchmark model [Odgaard and Johnson, 2013] is a fixed pitch sensor fault with a constant value of 1 deg of blade 1. In this work, only a collective pitch angle is included in the TS SM observer state vector. Thus, a pitch sensor fault can only be modelled as a collective fault of all three pitch angles. The fixed pitch angle sensor fault was simulated using FAST. The simulation results are depicted in Figure 6.20.

In general, for a fixed sensor fault with the faulty constant value $y_F = c$, an additive fault component f_s can still be reconstructed using the TS SMO, such that $y_F = y + f_s$ and in the case of perfect fault reconstruction, the relation $y = y_F - \hat{f}_s$ should hold.



(a) Black line: physical pitch angle; green dashed line: (faulty) pitch angle sensor value; red line: difference $\beta_F - \hat{f}_{s,\beta}$ between pitch angle sensor value and reconstructed sensor fault

(b) Actual (black) and reconstructed (red) additive pitch angle sensor fault component



(c) (Erroneously) reconstructed generator speed sensor fault component

Figure 6.20: FAST simulation with turbulent wind input (mean wind speed: 18 m/s) with fixed pitch angle sensor fault of 1 deg $\approx$ 0.0175 rad.

In Figure 6.20a, the physical and measured pitch angle are depicted, together with the faulty sensor output subtracted by the reconstructed sensor fault, i.e., $\beta_F - \hat{f}_{s,\beta}$, which should correspond to the physical pitch angle value in case of perfect fault reconstruction. It can be seen that the achieved reconstruction is reasonably good. However, even though no generator speed sensor fault is present, the reconstructed fault signal in this channel is not negligible (see Figure 6.20c), which is a similar effect as in the example of the additive sensor faults on page 99 and should be identified as erroneous fault detection in an FDI decision algorithm.

7

Fault-Tolerant Control

In this chapter, fault-tolerant control schemes for actuator and sensor fault compensation in wind turbines are investigated and tested in simulations. The faults are reconstructed by means of the modified TS sliding mode observer as described in section 6.3. Afterwards, the reconstructed faults are utilised to correct the faulty control input or output signals, respectively. In this way, virtual sensors or actuators are obtained which correspond to the fault-free cases in case of perfect fault reconstruction.

7.1 Actuator Fault-Tolerant Control

This section deals with actuator fault-tolerant control for wind turbines. Parts of this section were first published in [Georg and Schulte, 2013] [1].

7.1.1 Fault Compensation

If faults occur in the system, provided that these faults can be accurately reconstructed, the most straightforward strategy to achieve fault-tolerance is a simple fault compensation method, whereby the reconstructed (and possibly filtered) faults are subtracted from the demanded control input:

$$\mathbf{u}_{\text{corr}} = \mathbf{u} - \hat{\mathbf{f}}_{a,\text{filt}} \,, \tag{7.1}$$

where first order delay filters can be used to avoid feeding back erroneous peaks into the fault-tolerant control scheme. The dynamics of the filtered reconstructed fault signals is given by

$$\dot{\hat{\mathbf{f}}}_{a,\text{filt}} = -\text{diag}\,(1/\tau_1 \;\; 1/\tau_2)\;\hat{\mathbf{f}}_{a,\text{filt}} + \text{diag}\,(1/\tau_1 \;\; 1/\tau_2)\;\hat{\mathbf{f}}_a \,. \tag{7.2}$$

Here, the delay time constants are set to $\tau_1 = 0.3$ s and $\tau_2 = 2$ s. As an additional correction, the components of $\hat{\mathbf{f}}_a$ are set to zero during the first 20 seconds of the simulation, in order to avoid feeding back initial transients.

The schematics of the FTOL structure is depicted in Figure 7.1. After applying the correction (7.1), the

---

[1] Sören Georg, Horst Schulte. Actuator Fault Diagnosis and Fault-Tolerant Control of Wind Turbines using a Takagi-Sugeno Sliding Mode Observer. *International Conference on Control and Fault-Tolerant Systems (SysTol)*, pp. 516 – 522, Nice, France, 2013. © 2013 IEEE.

actual input signal that acts on the system is given by

$$\tilde{\mathbf{u}}_{\text{corr}} = \mathbf{u}_{\text{corr}} + \mathbf{f}_a = \mathbf{u} + \left(\mathbf{f}_a - \hat{\mathbf{f}}_{a,\text{filt}}\right) . \tag{7.3}$$

It is plain to see that the quality of the fault compensation depends on the quality of the fault reconstruction. If $\hat{\mathbf{f}}_{a,\text{filt}}$ were a perfect reconstruction of $\mathbf{f}_a$, $\tilde{\mathbf{u}}_{\text{corr}}$ would exactly correspond to the original faultless control signal $\mathbf{u}$.

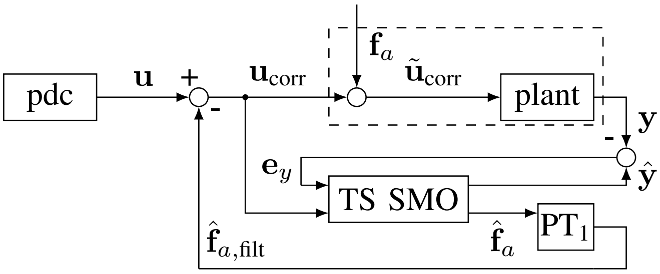
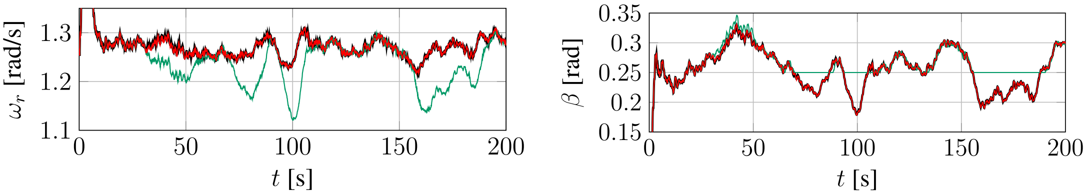


Figure 7.1: Schematics of the actuator fault-tolerant control structure. Reprinted from [Georg and Schulte, 2013] © 2013 IEEE.

7.1.2 Example Actuator Faults

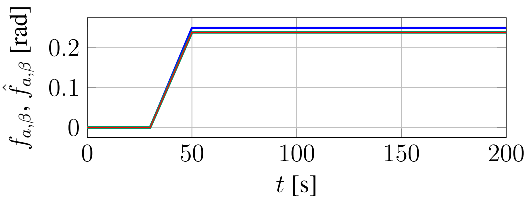
In this section, two example actuator faults are simulated in FAST with the applied fault compensation (7.1) and compared to the fault-free case and the faulty case without active fault compensation.

The first example is an incipient offset fault on the demanded pitch angle, which is increased with a linear ramp between 30 s and 50 s. The simulation results are depicted in Figures 7.2 and 7.3.



(a) Rotor speed

(b) Physical pitch angle



(c) Pitch angle faults. blue line: actual fault; green line: reconstructed fault in the case without fault compensation; red line: reconstructed fault in the case with active fault compensation

Figure 7.2: FAST simulations with turbulent wind input (mean wind speed: 18 m/s, turbulence intensity: 15 % (NTM), Kaimal model); Subfigures 7.2a, 7.2b: black lines: fault-free case; green lines: incipient pitch actuator offset fault (0.25 rad) active from 30 s, no fault compensation; red: with active fault compensation

It can be seen from Figure 7.2 that in the faulty case without fault compensation (green lines), the rotor speed is in parts significantly lower compared to the fault-free case. This is due to the pitch angle, which in these time-ranges is fixed at 0.25 rad although it should normally be lower. Since in the fault-free case the maximum generator torque has already been reached, the significantly reduced rotor speed in the faulty case will then inevitably lead to a decrease in produced power and thus a loss of energy. In the case with active fault compensation (red lines), the fault-free behaviour is almost recovered (see also Figure 7.3). This demonstrates the effectiveness of the fault compensation scheme based on direct fault reconstruction.

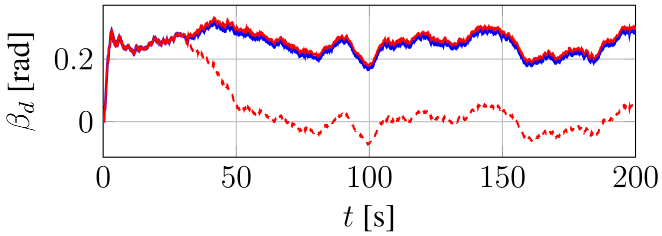


Figure 7.3: FAST simulation with the same pitch angle fault as in Figure 7.2 with active fault compensation; blue line: demanded pitch angle β_d; red dashed line: corrected pitch angle $\beta_{d,\mathrm{corr}}$; red solid line: effective demanded pitch angle $\tilde{\beta}_{d,\mathrm{corr}}$

As a second example, an incipient generator torque scaling fault was simulated, where the faulty scaling constant was set to 1.4. The simulation results are depicted in Figure 7.4.

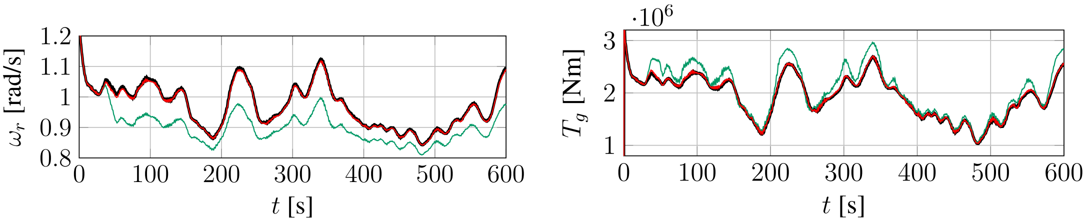
It can be seen from Figure 7.4a that when the fault is active, the rotor speed is reduced due to the increased generator torque. Since the generator torque-speed curve for the partial load region is selected such that the turbine is operating at the optimal tip-speed ratio, one would expect a reduction in produced power compared to the fault-free case. In fact, this is the case most of the time as can be seen in Figure 7.4c. This leads to a reduced energy production, which can be seen from the time-series of the energy difference in Figure 7.4d. Again, with active fault compensation, the fault-free behaviour is almost recovered, avoiding the loss of energy that occurs without fault compensation. Surprisingly, there is even a slight gain in energy compared to the fault-free case (see Figure 7.4d).

Regarding the reduction in produced energy in the faulty case without active fault compensation, one might expect a still more significant energy reduction compared to the fault-free case. However, such a conjecture would be based on the assumption that in the fault-free case the turbine is always running at the optimal tip-speed ratio. However, as can be seen in Figure 7.5, this is not the case. In fact, there are quite large deviations from the optimal ratio at times. This is due to the fast changes in the wind speed, which are not leveled off equally fast by the changes in the rotor speed.
One must keep in mind that the generator torque-speed curve for the partial load region is deduced based on static calculations, where no acceleration effects of the rotor are taken into account. The above finding supports the demand raised in many recent publications for closed-loop control in the partial load region (see for example [Johnson et al., 2006]).

7.1.3 Global Stability Analysis - Full Load Region

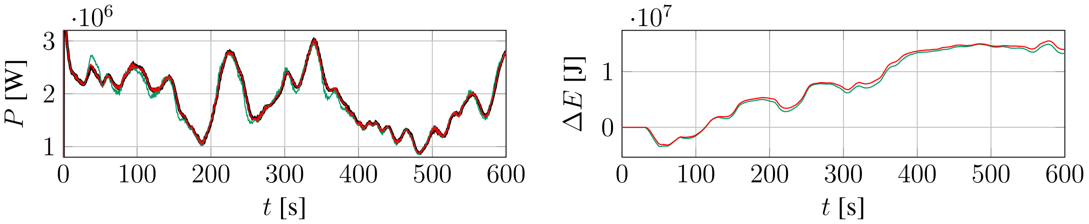
In the full load region, where only the pitch controller is active, the stability of the fault-tolerant control scheme can be shown similarly as for the standard control structure in the fault-free scenario. Analogously to the procedure in section 4.1.4, the derivative $\dot{V}$ of the Lyapunov function was analysed in 3 turbulent wind simulations with the mean wind speeds 14, 18 and 24 m/s to achieve a large coverage of the full load region. This does not constitute a formal proof of stability, but, arguing as in section 4.1.4, it gives good theoretical support for the stability.

Including faults and the active fault compensation, the approximated model (4.44) without the external bounded disturbance $\sum_i^{N_r} h_i(\beta)\,\tilde{\mathbf{E}}_i\,v$ is modified by the error term $\mathbf{F}\,\mathbf{f}_a = \mathbf{B}\,\mathbf{f}_a$ and the correction term $-\mathbf{B}\,\hat{\mathbf{f}}_{a,\mathrm{filt}}$:



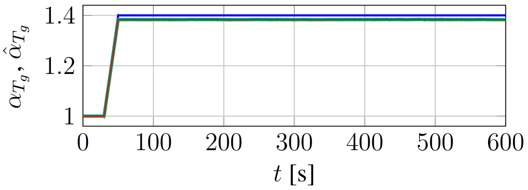
(a) Rotor speed

(b) Generator torque



(c) Power

(d) Green line: Energy difference between fault-free case and faulty case without fault compensation; red line: Energy difference between faulty case with fault compensation and faulty case without fault compensation.



(e) Generator torque scaling fault. blue line: actual (faulty) scaling constant; green line: reconstructed scaling constant in the case without fault compensation; red line: reconstructed scaling constant in the case with active fault compensation

Figure 7.4: FAST simulations with turbulent wind input (mean wind speed: 8 m/s, turbulence intensity: 20 % (NTM), Kaimal model); Subfigures 7.4a - 7.4c: black lines: fault-free case; green lines: incipient torque actuator scaling fault (scaling constant: 1.4) active from 30 s, no fault compensation; red: with active fault compensation

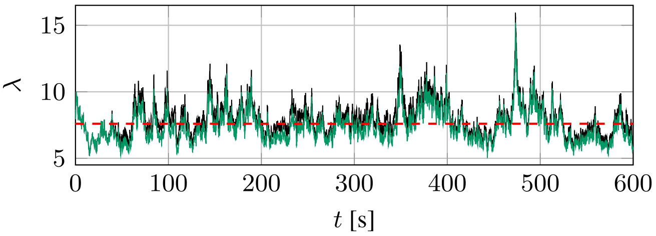


Figure 7.5: Tip speed ratio for turbulent FAST simulation in partial load (mean wind speed 8 m/s). Black line: fault-free case; green line: with generator torque scaling fault ($\alpha = 1.4$) without fault compensation. red dashed line: optimal tip-speed ratio.

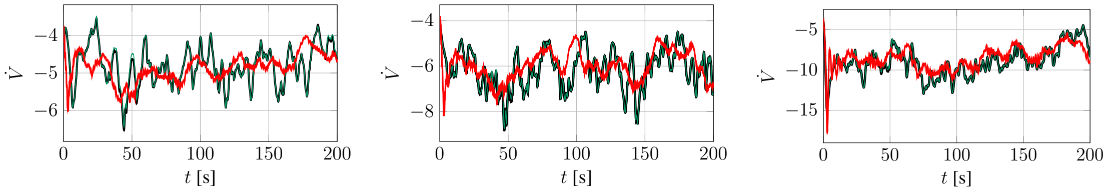
$$\dot{\mathbf{x}} = \sum_{i}^{N_r} h_i(\beta)\left(\mathbf{A}_i\,\mathbf{x} + \mathbf{B}\,u + \mathbf{a}_i\right) + \mathbf{B}\left(\mathbf{f}_a - \hat{\mathbf{f}}_{a,\text{filt}}\right) \tag{7.4}$$

$$= \sum_{i}^{N_r} h_i(\beta)\left(\mathbf{A}_{i,\text{CL}}\,\mathbf{x} + \mathbf{g}_i\right) - \mathbf{B}\,k_I\,x_I + \mathbf{B}\underbrace{\left(\mathbf{f}_a - \hat{\mathbf{f}}_{a,\text{filt}}\right)}_{=:\,\boldsymbol{\Delta}_f} \tag{7.5}$$

Then, the derivative of the Lyapunov function is given by

$$\dot{V} = \sum_{i}^{N_r} h_i\,(\beta)\;\mathbf{x}^T\left(\mathbf{P}\,\mathbf{A}_{i,\text{CL}} + \mathbf{A}_{i,\text{CL}}^T\,\mathbf{P}\right)\mathbf{x} + 2\sum_{i}^{N_r} h_i\,(\beta)\;\mathbf{x}^T\,\mathbf{P}\,\mathbf{g}_i - 2\,x^T\,\mathbf{P}\,\mathbf{B}\,k_I\,x_I + 2\,\mathbf{x}^T\,\mathbf{P}\,\mathbf{B}\,\boldsymbol{\Delta}_f\,. \tag{7.6}$$

Since the controller was designed using the reduced-order state-space model with only ω_r and β as states (see section 4.1.4), the simulations to calculate $\dot{V}$ were also carried out in the reduced-order model, in order to be consistent. The results are depicted in Figure 7.6, both for the fault-free case (black lines) and for the faulty case (incipient pitch angle offset fault of 0.25 rad) with active fault compensation, including the additional term $2\,\tilde{\mathbf{x}}^T\,\mathbf{P}\,\tilde{\mathbf{B}}\,\boldsymbol{\Delta}_f$ (green lines). For the sake of comparison, one simulation was also carried out in FAST (faulty case with active fault compensation), where the corresponding three states of the reduced state vector were extracted (red lines in Figure 7.6).



(a) 14 m/s (b) 18 m/s (c) 24 m/s

Figure 7.6: Derivatives of Lyapunov function for three different turbulent simulation runs in the full-load region. Black lines: simulation with reduced-order model (4.34), fault-free case; green lines: simulation with reduced-order model, incipient pitch angle offset fault of 0.25 rad, active fault compensation; red lines: FAST simulation with pitch angle fault and active fault compensation.

One can see from Figure 7.6 that in all three cases, $\dot{V}$ is below zero, such that the stability is ensured within the reduced-order model. As a general tendency, the negativity of $\dot{V}$ increases for higher wind speed, which means that the stability is increased. This corresponds to the findings in section 4.1.4.

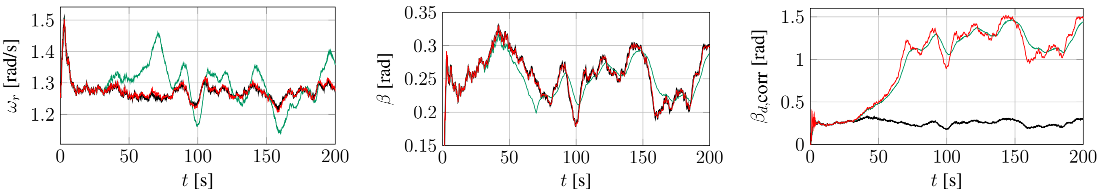
For the FAST simulations, the negativity of $\dot{V}$ is also ensured. However, especially in the case with mean wind speeds of 14 m/s, there are discrepancies between the FAST simulation results and the results from the reduced-order simulations. Still, the stability analysis using the reduced-order model seems to provide a reasonable estimate for the stability even in FAST simulations.

7.1.4 Actuator Scaling Faults - Limits of Fault Compensation

The pitch actuator offset fault simulated and analysed in section 7.1.2 represents a rather moderate fault in terms of fault magnitude and rate of increase. It could be shown in section 7.1.3 that the stability is still maintained in the presence of this fault with active fault compensation.

To investigate the limits of the fault-tolerant control strategy presented in this work, more severe pitch angle (scaling) faults shall be analysed in this section, combined with a stability analysis. Faulty scaling constants of $\alpha = 0.2$, 0.05, 0.02, and 0 were simulated by gradually introducing them with a linear ramp between 30 s and 70 s of a total simulation time of 200 s. In Figures 7.7 - 7.10, the results for the states ω_r and β, as well as for the demanded (corrected) pitch angle $\beta_{d,\text{corr}}$ are depicted for the fault-free case, the faulty case without fault compensation, and the faulty case with active fault compensation.

For the scaling faults in this section, the reconstructed fault signal $\hat{\mathbf{f}}_a$ was not filtered before feeding it back via the fault compensation (7.1), since the additional time-delay induced by the first-order filter led to unsatisfying fault compensation results in the cases of $\alpha \leq 0.05$.



(a) Rotor speed (b) Physical pitch angle (c) (Corrected) demanded pitch angle

Figure 7.7: FAST simulations with turbulent wind input (mean wind speed: 18 m/s, turbulence intensity: 15 % (NTM), Kaimal model) and incipient pitch angle scaling fault (final faulty scaling value: $\alpha = 0.2$. Black lines: fault-free case; green lines: faulty case without fault compensation; red: with active fault compensation

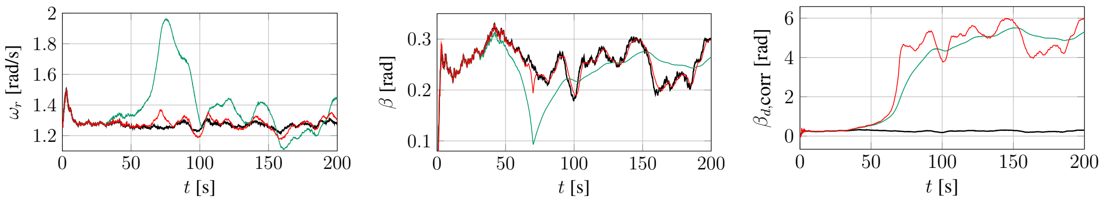
It can be seen from Figure 7.7 that with active fault compensation, the scaling fault with $\alpha = 0.2$ is fairly well compensated for, such that the system behaviour comes close to the behaviour in the fault-free case. In this case, a pitch rate and pitch angle limitation on the control system side (just behind the first node in Figure 7.1 where $\hat{\mathbf{f}}_{a,\text{filt}}$ is subtracted) can be included without leading to perceptible changes in the simulation results including fault compensation.

By contrast, for faulty scaling constants of $\alpha = 0.05$ and 0.01, no pitch rate and angle limitation was included in the control system side, as this would hamper the performance of the fault compensation. This can be observed from Figures 7.8c - 7.10c, where the pitch rates and the absolute values of the (corrected) demanded pitch angles significantly exceed the usual limits of 10 deg/s = 0.1745 rad/s and 1.57 rad.
On the actuator side (behind the second node where $\mathbf{f}_a$ is added), a pitch rate and angle limitation were included to avoid obtaining unphysical results.

While a fault compensation can still be achieved for faulty scaling constants near zero, the fault compensation scheme comes to technical limit in the case of a faulty scaling constant of $\alpha = 0$, since in this case, any value of the corrected demanded pitch angle will be scaled to zero, such that a compensation is impossible. This effect can be clearly seen in Figure 7.10, where the physical pitch angle falls rapidly to zero around a simulation time of 70 s (in the case with active fault compensation) and remains there (although the demanded pitch angle keeps increasing). Consequently, the rotor speed remains at values far beyond the rated speed.[2]

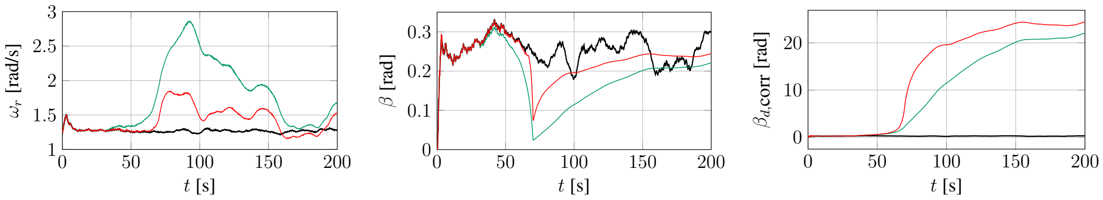
---

[2] Obviously, this consideration can only be done in a simulation. A real wind turbine is shut down once a critical rotor speed is exceeded, which for a turbine like the NREL 5 MW reference turbine would be no higher than 1.6 rad/s. This means that in the two cases of $\alpha = 0.2$ and 0.05, a fault correction would be possible without crossing the overspeed mark.



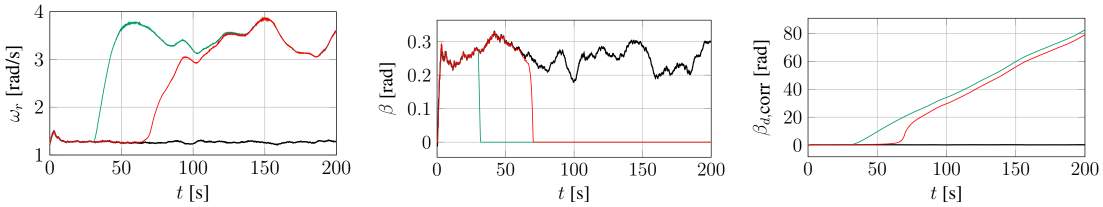
(a) Rotor speed (b) Physical pitch angle (c) (Corrected) demanded pitch angle

Figure 7.8: FAST simulations with turbulent wind input (mean wind speed: 18 m/s, turbulence intensity: 15 % (NTM), Kaimal model) and incipient pitch angle scaling fault (final faulty scaling value: $\alpha = 0.05$. Black lines: fault-free case; green lines: faulty case without fault compensation; red: with active fault compensation



(a) Rotor speed (b) Physical pitch angle (c) (Corrected) demanded pitch angle

Figure 7.9: FAST simulations with turbulent wind input (mean wind speed: 18 m/s, turbulence intensity: 15 % (NTM), Kaimal model) and incipient pitch angle scaling fault (final faulty scaling value: $\alpha = 0.01$. Black lines: fault-free case; green lines: faulty case without fault compensation; red: with active fault compensation



(a) Rotor speed (b) Physical pitch angle (c) (Corrected) demanded pitch angle

Figure 7.10: FAST simulations with turbulent wind input (mean wind speed: 18 m/s, turbulence intensity: 15 % (NTM), Kaimal model) and incipient pitch angle scaling fault (final faulty scaling value: $\alpha = 0$. Black lines: fault-free case; green lines: faulty case without fault compensation; red: with active fault compensation

Stability Analysis

A stability analysis, which is conducted for the four cases with different scaling constants, yields negative derivatives of the Lyapunov function for the three cases of $\alpha = 0.2$, 0.05 and 0.01, which means a stable closed-loop control. By contrast, in the case of $\alpha = 0$, $\dot{V}$ exceeds zero at a simulation time of approximately 70 s, which means that no stable closed-loop control can be achieved. This result corresponds to the intuitive observation that no fault compensation can be achieved for the case of $\alpha = 0$.

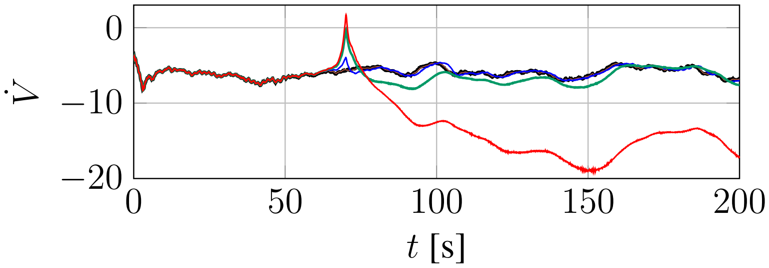


Figure 7.11: Derivatives of Lyapunov functions for turbulent FAST simulations (mean wind speed: 18 m/s) with incipient scaling faults of the pitch angle for different scaling constants and active fault compensation. Black line: fault-free case; orange line: $\alpha = 0.2$; blue line: $\alpha = 0.05$; green line: $\alpha = 0.01$; red line: $\alpha = 0$.

7.2 Sensor Fault-Tolerant Control

Using the dynamic sensor fault reconstruction method (6.34) introduced in Section 6.2, a sensor fault-tolerant control scheme can be implemented by substracting the reconstructed sensor faults from the (faulty) outputs:

$$\begin{aligned} \mathbf{y}_{\text{corr}} &= \mathbf{C}\,\mathbf{x} + \mathbf{N}\,\mathbf{f}_s - \mathbf{N}\,\hat{\mathbf{f}}_s \\ &= \mathbf{C}\,\mathbf{x} + \mathbf{N}\left(\mathbf{f}_s - \hat{\mathbf{f}}_s\right) , \end{aligned} \tag{7.7}$$

where the quality of the fault compensation is determined by the quality of the fault reconstruction, i.e., whether the difference $\mathbf{f}_s - \hat{\mathbf{f}}_s$ is close to zero.

7.2.1 Example Sensor Faults

Additive Fault

As an example, an additive fault in the generator speed sensor shall be reconstructed using a TS SMO with observer state vector $\hat{\mathbf{x}} = \begin{pmatrix} \hat{\theta}_s & \hat{\omega}_r & \hat{\omega}_g & \hat{\beta} & \hat{T}_g \end{pmatrix}^T$. The output vector is given by $\hat{\mathbf{x}} = \begin{pmatrix} \hat{\omega}_r & \hat{\omega}_g & \hat{\beta} & \hat{T}_g \end{pmatrix}^T$. As in section 6.5, the sensor fault selection matrix was chosen as $\mathbf{N} = \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$, such that the sensor fault vector is given by $\mathbf{f}_s = \begin{pmatrix} f_{s,\omega_g} & f_{s,\beta} & f_{s,T_g} \end{pmatrix}^T$.

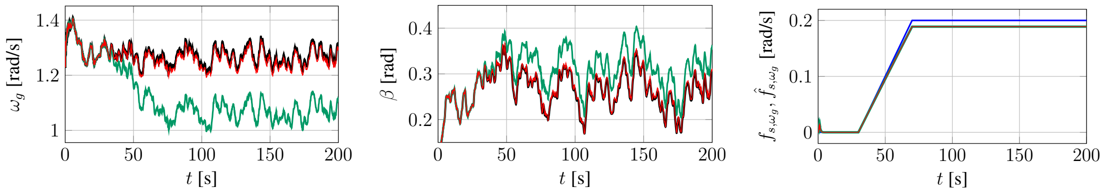
Simulations were conducted using both the nominal model and FAST, with an additive sensor fault of the generator speed ω_g active from 30 s. In the first case, the reference wind speed was used as input for the TS SMO, whereas for the FAST simulation, the reconstructed wind speed obtained from the TS Luenberg wind observer (see chapter 5) was used. In both cases, the generator speed ω_g was used here as controlled variable instead of the rotor speed ω_r [3]. The fault magnitude was increased linearly between 30 s and 70 s up to its full height of 0.2 rad/s. In Figure 7.12, one can see that the sensor fault is pretty well reconstructed (to a degree of about 93 % [4]) with the nominal model. Due to the precise sensor fault reconstruction, very good results are achieved for the fault-tolerant control, too. The simulation behaviour is close to the behaviour in the fault-free case.

By contrast, in the FAST simulation (see Figure 7.13), the quality of the sensor fault reconstruction is reduced. The mean fault value is still reasonably well reconstructed, but the signal is superimposed with noise. As a consequence, the fault compensation within the FAST simulation is of reduced quality compared to the nominal model simulation. Still, compared to the case without any fault compensation, the simple sensor fault-tolerant control scheme achieves a reasonable fault compensation even in FAST (see Figure 7.13a).

---

[3] and a notch filter was included in FAST simulations to obtain a filtered signal of ω_g to be used in the controller (see section 4.3)

[4] with $\delta \to 0$ and $\Delta t \to 0$, the fault magnitude could theoretically be reconstructed to a degree of 100 %, as argued in section 6.3.3

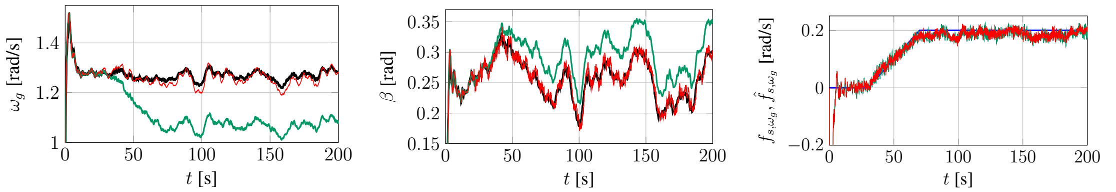


(a) Generator speed

(b) Physical pitch angle

(c) Actual (blue) and reconstructed (green, red) sensor fault

Figure 7.12: Turbulent simulation of the nominal model (mean wind speed: 18 m/s, turbulence intensity: 15 % (NTM), Kaimal model) with incipient offset fault of the generator speed sensor. Black lines: fault-free case; green lines: faulty case without fault compensation; red lines: faulty case with active fault compensation.



(a) Generator speed

(b) Physical pitch angle

(c) Actual (blue) and reconstructed (green, red) sensor fault

Figure 7.13: Turbulent FAST simulation (mean wind speed: 18 m/s, turbulence intensity: 15 % (NTM), Kaimal model) with incipient offset fault of the generator speed sensor. Black lines: fault-free case; green lines: faulty case without fault compensation; red lines: faulty case with active fault compensation.

Stuck Sensor Fault

As a second example, a stuck generator speed sensor fault was simulated. The results are depicted in Figure 7.14 for the nominal model simulation and in Figure 7.15 for the FAST simulation.

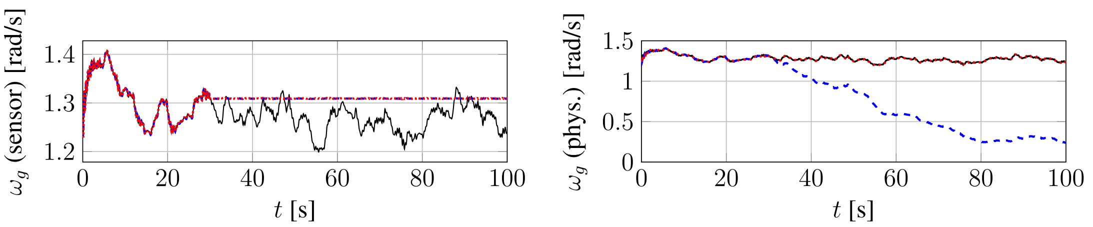
As with the additive sensor fault example, the results with the stuck sensor fault show a similar pattern. In the nominal model simulation, a precise fault reconstruction can be achieved, which can be seen in Subfigure 7.14d, where the difference $(\omega_{g,F} - \hat{f}_{s,\omega_g})$ between the faulty generator speed and the reconstructed fault signal is almost equal to the generator speed in the fault-free case. Consequently, a very good fault-tolerant behaviour is achieved (see Subfigure 7.14b).

Within the FAST simulation, the reconstruction quality is reduced. Still, a reasonable fault-tolerant behaviour is achieved when the fault compensation is active (see Subfigure 7.15b).

7.2.2 Global Stability Analysis - Full Load Region

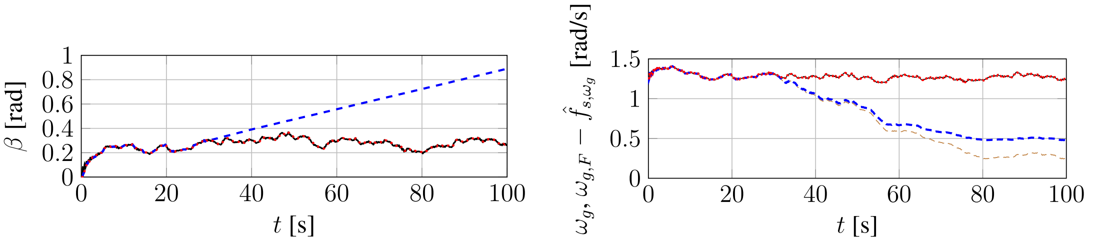
In this section, the global stability in the full load region in the presence of generator speed sensor faults with active fault compensation is examined analogously to sections 4.1.4 and 7.1.3, that is by analysing the derivative of an appropriate Lyapunov function for several example turbulent simulation runs. Since the PDC pitch controller (4.37) was designed based on a 2-state reduced-order model with the state vector $\mathbf{x} = (\omega_r \quad \beta)^T$, the stability analyses in sections 4.1.4 and 7.1.3 were also based on this 2-state model.

When including generator speed sensor faults, however, the generator speed must obviously be included in the stability analysis. In the following analysis, the rotor speed ω_r is simply replaced by the generator speed ω_g. Strictly speaking, this is an inconsistency and not fully theoretically correct. However, when evaluating the derivatives of the Lyapunov function with either the rotor speed or the generator speed,



(a) Generator speed (sensor values)

(b) Generator speed (physical values)



(c) Pitch angle

(d) Black line: generator speed in the fault-free case; red dotted line: difference between faulty generator speed and reconstructed sensor fault ($\omega_{g,F} - \hat{f}_{s,\omega_g}$) in the case with active fault compensation; brown dashed line: physical generator speed in the faulty case without fault compensation; blue dashed line: difference ($\omega_{g,F} - \hat{f}_{s,\omega_g}$) in the case without active fault compensation.

Figure 7.14: Turbulent simulation of the nominal model (mean wind speed: 18 m/s, turbulence intensity: 15 % (NTM), Kaimal model) with stuck generator speed sensor fault. Black lines: fault-free case; blue dashed lines: faulty case without fault compensation; red dotted lines: faulty case with active fault compensation.

there are only marginal differences except for initial transients. Despite the inconsistency, the following procedure therefore still yields a reasonable estimation for the stability.

The system to be analysed is the approximated model (4.40) without the external bounded disturbance $\sum_i^{N_r} h_i(\beta)\,\mathbf{E}_i\,v$, i.e. the autonomous model

$$\dot{\mathbf{x}} = \sum_i^{N_r} h_i(\beta)(\mathbf{A}_i\,\mathbf{x} + \mathbf{B}\,u + \mathbf{a}_i)\,, \tag{7.8}$$

with the affine term $\mathbf{a}_i = -\mathbf{A}_i\,\mathbf{x}_i - \mathbf{B}\,u_i - \mathbf{E}_i\,v_i$ as defined in section 4.1.4.

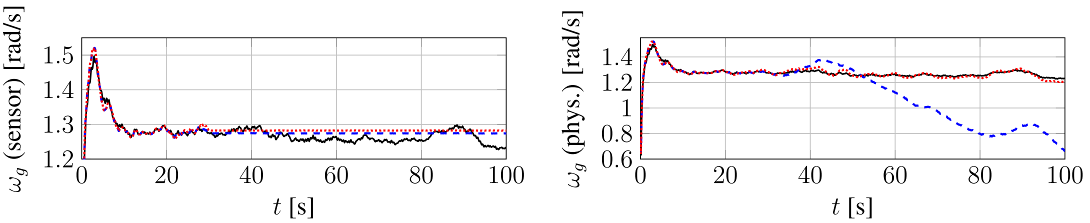
In the fault-free case, the controller (4.37) can either be written as a state control-law

$$u = -\sum_i^{N_r} h_i\,(\beta)\;\mathbf{k}_i^T\,\Delta\mathbf{x}_i - k_I\,x_I \tag{7.9}$$

or, alternatively, as an output control-law:

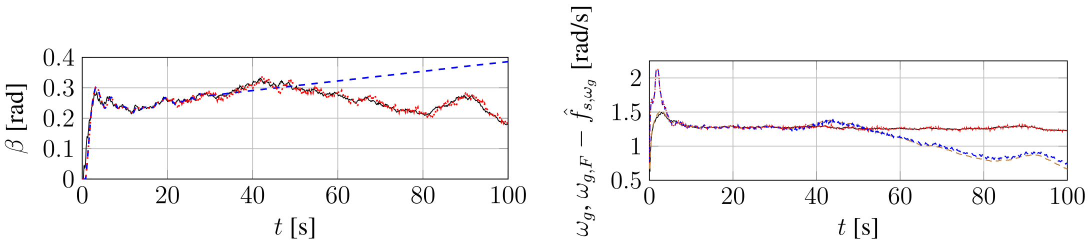
$$u = -\sum_i^{N_r} h_i\,(\beta)\;\mathbf{k}_i^T\,\Delta\mathbf{y}_i - k_I\,x_I\,, \tag{7.10}$$

because both states ω_g and β are measurable.



(a) Generator speed (sensor values)

(b) Generator speed (physical values)



(c) Pitch angle

(d) Black line: generator speed in the fault-free case; red dotted line: difference between faulty generator speed and reconstructed sensor fault ($\omega_{g,F} - \hat{f}_{s,\omega_g}$) in the case with active fault compensation; brown dashed line: physical generator speed in the faulty case without fault compensation; blue dashed line: difference ($\omega_{g,F} - \hat{f}_{s,\omega_g}$) in the case without active fault compensation.

Figure 7.15: Turbulent FAST simulation (mean wind speed: 18 m/s, turbulence intensity: 15 % (NTM), Kaimal model) with stuck generator speed sensor fault. Black lines: fault-free case; blue dashed lines: faulty case without fault compensation; red dotted lines: faulty case with active fault compensation.

In the presence of a generator speed sensor fault and with the active fault-compensation (7.7), the output vector is modified to $\mathbf{y}_{\text{corr}} = \mathbf{y} + \mathbf{N}\left(f_s - \hat{f}_s\right) = \mathbf{x} + \begin{pmatrix}\Delta f_{s,\omega_g}\\ 0\end{pmatrix}$, where $\Delta f_{s,\omega_g} := \left(f_{s,\omega_g} - \hat{f}_{s,\omega_g}\right)$. Furthermore, the integrator state is modified: $x_{I,\text{corr}} = \int_0^t d\tau\, \left(\omega_g - \omega_{g,\text{SP}} + \Delta f_{s,\omega_g}\right)$.

Thus, the control law is modified to

$$u_{\text{corr}} = -\sum_i^{N_r} h_i(\beta)\, \mathbf{k}_i^T\, \Delta\mathbf{y}_i - k_I\, x_{I,\text{corr}} - \sum_i^{N_r} h_i(\beta)\, \mathbf{k}_i^T \begin{pmatrix}\Delta f_{s,\omega_g}\\ 0\end{pmatrix} . \tag{7.11}$$

Defining $l_i := -\mathbf{k}_i^T\, \mathbf{x}_i = -(k_{\omega_r,i}\, \omega_{g,\text{SP}} + k_{\beta,i}\, \beta_i)$, the control law can be written as

$$u_{\text{corr}} = -\sum_i^{N_r} h_i(\beta)\, \left(\mathbf{k}_i^T\, \mathbf{x} + l_i\right) - \sum_i^{N_r} h_i(\beta)\, k_{\omega_g,i}\, \Delta f_{s,\omega_g} - k_I\, x_{I,\text{corr}} . \tag{7.12}$$

Introducing the closed-loop system matrix $\mathbf{A}_{\text{CL},i} := \mathbf{A}_i - \mathbf{B}\, \mathbf{k}_i^T$ and the affine term $\mathbf{g}_i := \mathbf{a}_i - \mathbf{B}\, l_i$, the closed-loop system can be written as

$$\dot{\mathbf{x}} = \sum_i^{N_r} h_i(\beta)\, \left(\mathbf{A}_{\text{CL},i}\, \mathbf{x} + \mathbf{g}_i\right) - \sum_i^{N_r} h_i(\beta)\, \mathbf{B}\, k_{\omega_g,i}\, \Delta f_{s,\omega_g} - \mathbf{B}\, k_I\, x_{I,\text{corr}} . \tag{7.13}$$

Let $V(\mathbf{x}) = \mathbf{x}^T \mathbf{P} \mathbf{x}$ be a candidate Lyapunov function for the closed-loop system (7.13). Then it is obvious that $V > 0$ holds. It must now be shown that $\dot{V} < 0$ is fulfilled.

$$\dot{V} = \sum_{i}^{N_r} h_i\,(\beta)\;\mathbf{x}^T \left(\mathbf{P}\,\mathbf{A}_{\mathrm{CL},i} + \mathbf{A}_{\mathrm{CL},i}^T\,\mathbf{P}\right)\mathbf{x} + 2\sum_{i}^{N_r} h_i\,(\beta)\;\mathbf{x}^T\,\mathbf{P}\,\mathbf{g}_i + \dot{V}_I + \dot{V}_{f_s}\,, \tag{7.14}$$

where the additional terms $\dot{V}_I$ and $\dot{V}_{f_s}$ are given by

$$\begin{aligned} \dot{V}_{f_s} &= -2\sum_{i}^{N_r} h_i\,(\beta)\;\mathbf{x}^T\,\mathbf{P}\,\mathbf{B}\,k_{\omega_g,i}\,\Delta f_{s,\omega_g}\,, && (7.15)\\ \dot{V}_I &= -2\,\mathbf{x}^T\,\mathbf{P}\,\mathbf{B}\,k_I\,x_{I,\mathrm{corr}}\,. && (7.16) \end{aligned}$$

The first term in (7.14) is negative if a common, symmetric, positive definite matrix $\mathbf{P}$ exists for all linear submodels, such that the Lyapunov condition

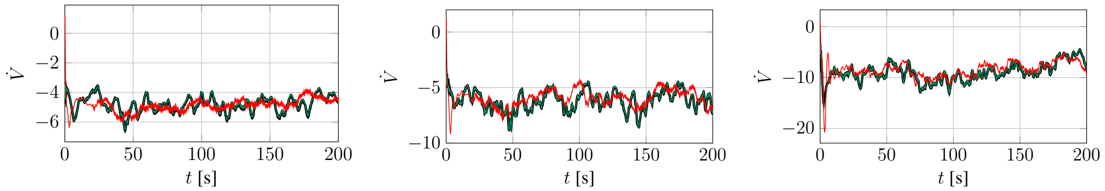
$$\mathbf{P}\,\left(\mathbf{A}_i - \mathbf{B}\mathbf{k}_i^T\right) + \left(\mathbf{A}_i - \mathbf{B}\mathbf{k}_i^T\right)^T\,\mathbf{P} < 0 \tag{7.17}$$

is fulfilled for all submodels. As in section 4.1.4, the following valid solution was obtained:

$$\mathbf{P} = \begin{pmatrix} 5.8697 & -0.2465 \\ -0.2465 & 0.3080 \end{pmatrix}. \tag{7.18}$$

Analogously to sections 4.1.4 and 7.1.3, the time-series of $\dot{V}$ was calculated for several turbulent simulation runs so as to obtain a reasonable estimate of $\dot{V}$ for the complete full load region. In Figure 7.16, results of $\dot{V}$ are depicted for three turbulent simulations of the nominal model with mean wind speeds of 14, 18, and 24 m/s in the case of the generator speed offset fault from section 7.2.1 (final fault value: 0.2 rad/s). In each case, the results are compared to the results of the fault-free case. Also, FAST simulations of the faulty case with fault compensation were conducted for the sake of comparison.

It can bee seen that even when including the effect of the fault and fault compensation (both in the nominal model and in FAST), the derivatives of the Lyapunov function are well below zero in all three cases. Again, this it not a formal proof for the global stability, but it gives very good theoretical support that the stability of the controller in the full load region is preserved even in the presence of generator speed sensor faults and with active fault compensation.



(a) 14 m/s (b) 18 m/s (c) 24 m/s

Figure 7.16: Derivatives of Lyapunov function obtained from (7.14) for turbulent simulations with incipient offset fault of the generator speed sensor (final value: 0.2 rad/s). Black lines: simulation of nominal model, fault-free case; green lines: simulation of nominal model, faulty case with active fault compensation; red lines: FAST simulation, faulty case with active fault compensation

7.3 Remarks on the Applicability of the FTC Scheme

The presented FTC scheme for compensating faults has the advantage that no decision algorithm is needed to decide upon the type of fault, since the fault compensation equations (7.1) and (7.7) make use of the reconstructed additive fault components $\hat{f}_a$ respectively $\hat{f}_s$ of the reconstructed fault, regardless of whether an additive, scaling or other type of fault has occured.

However, the presented FTC strategies only work in this straightforward way under the assumption that either actuator and sensor faults are not present at the same time, which is, however, a common assumption in FDI problems Chen and Patton [1999]. A general structural weakness of the fault reconstruction methods (6.19) and (6.34) is that actuator and sensor faults are not exactly distinguishable, which may lead to undesired results when both actuator and sensor fault compensation is active. For example, if two separate observers were used for actuator and sensor fault reconstruction and either a pitch or torque actuator fault occured, both observers would reconstruct a fault, such that an erroneous sensor fault correction would be applied even though no sensor fault was present in the first place.

For a practical application on a wind turbine, an appropriate decision block would thus first have to be added in order to appropriately differentiate between sensor and actuator faults, which has not yet been considered in this work.

For sensor faults, the situation would be different if two sensors were provided per output, in which case one observer could be used for each output. Upon the occurrence of a sensor fault (on only one output), only the corresponding observer would reconstruct a fault, whereas an actuator fault would affect both observers. Consequently, distinguishing between actuator and sensor faults would be possible [Pöschke, Georg, and Schulte, 2014].

8

Conclusion and Outlook

8.1 Conclusion

In this work, fault diagnosis and fault-tolerant control schemes for wind turbines were investigated based on direct fault reconstruction by means of Takagi-Sugeno sliding mode observers. As important milestones, the formulation of a reduced-order dynamical wind turbine model in TS structure, the PDC controller design for the full load region, and the estimation of an effective wind speed by means of a TS Luenberger observer were achieved.

In simulations of the nominal reduced-order model, fast and precise reconstructions of both actuator (pitch angle and generator torque) and sensor faults (pitch angle, generator torque, generator speed) could be achieved. In simulations of the aero-elastic code FAST, some faults (pitch angle and generator torque) could be reconstructed with virtually no loss of accuracy, whereas in the case of generator speed sensor faults, the reconstruction quality within FAST simulations was reduced, though still useful for fault-tolerant control.

Though the evaluation of the equivalent output injection signal of the TS sliding mode observers leads to a reconstruction of the additive fault components, other types of faults like scaling or stuck faults could be equally well reconstructed in simulations by tracing them back to the additive fault structure. Even for actuator parameter faults, a scheme was successfully implemented to obtain estimates of these types of faults, too.

The precise fault reconstructions allowed for using the signals in fault compensation schemes to achieve fault-tolerant control in the presence of both actuator and sensor faults with a behaviour similar to the fault-free case. Moreover, it could be shown that the stability (in the full load region) of the controller is maintained in the presence of faults when the fault compensation is active. To this end, the derivatives of appropriate Lyapunov functions were analysed for several turbulent simulation runs.

8.2 New Contributions

In this section, an overview of some new contributions of this work (respectively previously published works by the author) to the research fields of wind turbine modelling and control, as well as fault diagnosis and fault-tolerant control is given.

Wind Turbine Modelling and Control

Analytical Approximations for Aero Maps The analytical approximations for the rotor torque coefficient C_Q given in [Wasynczuk et al., 1981; Amlang et al., 1992; Heier, 2006] were extended in [Georg et al., 2012] with more design freedom for tuning the parameters in order to achieve a better agreement between the tabulated aero maps (extracted from FAST/AeroDyn or calculated by means of a BEM method) and the analytical functions.

The new expression

$$\begin{aligned} C_Q\left(\lambda,\beta\right) = &\; c_1\left(1 + c_2\,\left(\beta + c_3\right)^{\frac{1}{2}}\right) \\ &+ \frac{c_4}{\lambda}\,\left(c_5\,\lambda_i\left(\lambda,\beta\right)\,-\,c_6\beta\,-\,c_7\beta^{c_8}\,-c_9\right)\,\exp\left(-c_{10}\,\lambda_i\left(\lambda,\beta\right)\right)\quad\left(\lambda>0\right), \end{aligned} \tag{8.1}$$

with $\lambda_i\left(\lambda,\beta\right) \,=\, \frac{1}{\lambda + 0.08\,\beta} \,-\, \frac{0.035}{c_{11} + c_{12}\beta^3} \quad (\lambda > 0)$

is by no means simple, yet the achieved agreement between the tabulated and approximated aero maps is considerably good for pitch angles between 0 and 20°, which corresponds to the main operating range (see Figure 2.5).

Furthermore, since expression (8.1) is still analytically differentiable, it allows to obtain linearised system matrices at stationary points which contain comparably precise information about the aerodynamic behaviour at the corresponding local point, reflected in the derivatives of C_Q with respect to the rotor speed, the pitch angle, and the wind speed (see Section 4.1.1).

For the rotor thrust coefficient, a new analytical approximation was introduced [Georg et al., 2012]:

$$C_T\left(\lambda,\beta\right) \,=\, a_1 \,+ a_2\,\left(\lambda - a_3\beta\right)\,e^{-a_4\beta} \,+\, a_5\,\lambda^2\,e^{-a_6\beta} \,+\, a_7\,\lambda^3\,e^{-a_8\beta}\,. \tag{8.2}$$

With this new expression, a reasonably good agreement between the analytical and the tabulated C_T map can be achieved for pitch angles between 0 and 20° (see Figure 2.6). By calculating analytical derivatives of (8.2), it is then possible to include the variations in rotor thrust into the control design, which is useful if load-reducing multi-variable control structures shall be designed (see the discussion in Section 4.1.3).

Derivation of Parameters for Reduced-Order Model In this work, the reduced-order wind turbine model (2.29) was taken as a basis for control design as well as for some simulation studies, where the NREL 5 MW turbine [Jonkman et al., 2009] was chosen as the underlying turbine reference. Whereas some parameters from [Jonkman et al., 2009] could be directly taken for the nominal model (2.29), other parameters had to be specifically adapted for the reduced-order model, like the stiffness parameters for the tower and blade degrees of freedom (see Section 2.3.2).

A direct stiffness method and beam bending theory were applied in [Georg, Müller, and Schulte, 2014] to obtain a theoretically well founded value for the effective translational stiffness parameter for the tower-top motion, which comes close to the value estimated from direct comparison in simulation studies.

Together with the inclusion of a centrifugal blade stiffening into the reduced-order model (see Section 2.2.5), the two aforementioned points thus contribute to the general goal of closing the gap between precise high-order dynamic models used in load simulation packages like FAST and reduced-order models needed for control design purposes, which nevertheless need to be tuned as precisely as possible.

TS Observer for Direct Wind Speed Reconstruction In [Georg, Müller, and Schulte, 2014] and Chapter 5 of this work, a TS Luenberger observer was presented which achieves a reasonably good reconstruction of the rotor effective wind speed. Obviously, there are numerous techniques that could be applied for an effective wind speed estimation. A recent comparison of different approaches can be found in [Soltani et al., 2013]. However, these techniques are either comparably specialised (Extended Kalman Filter, Immersion and Invariance Estimator), require an implicit numerical calculation of the wind speed from the reconstructed rotor torque (Power Balance Estimator) or are based on linear models and thus require several designs at different operating points (Unknown Input Observer, see also [Odgaard et al., 2008]).

The advantage of the TS observer presented in Chapter 5 is that it can be used for the whole nonlinear operating range of the wind turbine[1], since the underlying observer model was obtained using the sector-nonlinearity approach. Furthermore, the TS structure guarantues a certain level of uniformity with the other controller and observer designs presented in this work and the optimal LMI design procedure allows a fairly simple tuning of the observer.

Fault Diagnosis and Fault-Tolerant Control

Weighted Switching Term for Sliding Mode Observer In [Georg and Schulte, 2014c], a weighted switching term for sliding mode observers was introduced:

$$\boldsymbol{\nu} = \begin{cases} -\boldsymbol{\rho}\,\frac{\mathbf{W}\,\mathbf{P}_2\,\tilde{\mathbf{e}}_y}{\|\mathbf{W}\,\mathbf{P}_2\,\tilde{\mathbf{e}}_y\|} & \text{if } \tilde{\mathbf{e}}_y \neq \mathbf{0} \\ \mathbf{0} & \text{otherwise} \end{cases}, \tag{8.3}$$

where $\mathbf{W}$ denotes a weighting matrix for the individual system outputs and $\boldsymbol{\rho}$ denotes the sliding mode gain matrix. With these two modifications, it is possible to simultaneously reconstruct faults of significantly differing orders of magnitude, like generator torque and pitch angle actuator faults in wind turbines (see Section 6.3).

Reconstruction/Detection of Actuator Parameter Faults The reconstruction of actuator parameter faults for wind turbines presented in [Georg and Schulte, 2014a] and Section 6.4.2 was implemented for the first time by evaluating the equivalent output injection signal of a TS sliding mode observer. More commonly, parameter estimation schemes are applied for this type of faults (see for example [Rotondo et al., 2012]). Although the results in section 6.4.2 may not yet allow for using the method for fault-tolerant control of parameter faults, a fast fault detection scheme may be possible. The presented method thus provides a further contribution to studying the application fields of sliding mode techniques in fault diagnosis.

---

[1] although the performance decreases slightly in the partial load region

8.3 Outlook

In this final section, an outlook will be given on possible future extensions and improvements of the work. From all the chapters in this thesis, many possible paths may be taken for future work, which would all make for interesting and sizeable research packages.
The outlook is divided in the two main parts **Modelling and Control** and **Fault Diagnosis and Fault-Tolerant Control**.

Wind Turbine Modelling and Control

Extension of Nominal Wind Turbine Model The nominal reduced-order wind turbine model presented in Chapter 2 features four degrees of freedom. For the controller and observer designs in this work, this model order proved to be well-suited, indeed, sometimes even a further reduction to only one or two degrees of freedom was advantageous.

However, some applications or specialised control designs would require further extensions of the model order. For instance, for individual pitch control [Bossanyi et al., 2010b,a], three individual blade degrees of freedom need to be represented in the control design model. Such an extension would also be necessary for introducing the concept of the multi-blade coordinate transformation, which can be exploited for the detection of rotor asymmetries [Henriksen et al., 2011, 2012]. In the two latter works, an Extended Kalman Filter was used for the state estimation and residual generation. It would be interesting to explore the possibilities of TS sliding mode observers for this category of rotor asymmetry faults.

Control Design based on Sector-Nonlinearities The pitch controller design presented in section 4.1 was based on Taylor-linearised models at several stationary points. A strong case can be made for using this approach, as the derivatives of the aero maps, which represent the dynamics at each local point, are then included in the control design, resulting in control gains well tuned to the aerodynamical behaviour.

Still, it would be well worth attempting a pitch control design based on a TS model obtained by means of the sector-nonlinearity approach. For one thing, fewer TS submodels would be needed for the controller, resulting in a reduced computation time. Moreover, it might facilitate stability analyses in connection with the TS observer for wind speed estimation and the TS sliding mode observers for fault reconstruction, if the same underlying models and thus the same TS membership functions were used.

Control for Load Reduction Apart from individual pitch control, there is much room for investigating other aspects of load reducing control structures (like reducing drivetrain or tower vibrations) within multivariable control design approaches, as was already embarked on in section 4.1.3. Particularly, it would be interesting to check whether a damping of oscillations can also be achieved within FAST simulations.

Wind Observer The TS observer for the effective wind speed estimation has not yet been compared to other established methods like Kalman Filtering or Unknown Input Observers. Such a comparison would be helpful to establish the TS wind observer as a possible standard method. Furthermore, the observer design as yet has been conducted based on models containing only drivetrain degrees of freedom. The question remains whether an inclusion of blade or tower degrees of freedom can either improve estimation results or lead to a reduction of the wind speed weights in the optimal LMI design procedure.

Fault Diagnosis and Fault-Tolerant Control

As for the fault diagnosis and fault-tolerant control part, further extensions in both theoretical and practical aspects would be desirable.

Coordinate Transformations for TS Sliding Mode Observer The fault reconstruction schemes in Chapter 6 using TS sliding modes observers require a transformation of the original system into a special system structure. For sliding mode observers based on linear systems, one canonical transformation is obtained, such that the stability can be shown in both the original and the transformed system [Edwards and Spurgeon, 1998]. The same would be the case for nonlinear TS systems which nevertheless yield one common $\mathbf{T}$-matrix for all TS subsystems, which for example is the case for the TS sliding mode observer used for actuator fault reconstruction in this work. For the case of distinct $\mathbf{T}$-matrices for the TS subsystems, it is a question of current research to establish a formal framework for the transformations [Schulte and Georg, 2014].

Reconstruction of Actuator Parameter Faults The use of a TS sliding mode observer for the reconstruction of actuator parameter faults in [Georg and Schulte, 2014a] and Section 6.4.2 revealed the broad range of different fault types that can in principle be detected using sliding mode observer techniques. However, the results in Section 6.4.2 also showed that both the reconstruction quality and the robustness would need to be improved before this technique for reconstructing parameter faults could be applied in fault-tolerant control concepts.

Rotor and Generator Speed Sensor Faults In terms of possible fault-tolerant control loops in wind turbines, generator speed sensor faults would be the most important or at least one of the most important faults to reconstruct at a high quality. The examples in Section 7.2.1 showed that the reconstruction quality for generator speed sensor faults is very high in simulations of the nominal model, yet reduced in FAST simulations. Further improvements here would be desirable, as well as a successful reconstruction of rotor speed sensor faults, which has not yet been achieved using the methods presented in this work.

Distinction between Actuator and Sensor Faults One structural weakness of the sliding mode fault reconstruction techniques is that actuator and sensor faults are not directly distinguishable. As already explained in Section 7.3, one exception would be a situation with two redundant sensors and two separate observers. Apart from this, an appropriate decision block to differentiate between actuator and sensor faults would have to be included before applying the fault-tolerant schemes presented in Chapter 7 on a real wind turbine. This decision block would then have to make use of different strategies, for example evaluating residual signals.

Tests on Real Wind Turbine or Test Rig The results in this work have all been obtained in simulation studies. Though FAST is a reasonably detailed wind turbine model, fault diagnosis and fault-tolerant control schemes ultimately have to be tested on a real wind turbine or at least on a test rig representing one system part of a wind turbine like a pitch system or a miniature drivetrain. First promising tests on a drivetrain test rig have been conducted in [Georg et al., 2014] and are envisaged to be extended in future work.

A

Appendix

A.1 Model Parameters

ρ	1.225 $\frac{\text{kg}}{\text{m}^3}$	N	3
R	63 m	r_B	21.975 m
m_{Blade}	17740 kg	m_{Tower}	347640 kg
m_{Rotor}	110000 kg	m_{Nacelle}	240000 kg
m_{Hub}	56780 kg	m_T	436865 kg
m_B	4435 kg	n_g	97 (gearbox ratio)
J_r	38759227 $\text{kg}\,\text{m}^2$	J_g	534 $\text{kg}\,\text{m}^2$ $\cdot\, n_g^2 = 5025347\,\text{kg}\,\text{m}^2$
k_S	867637000 Nm	d_S	6215000 Nm s
d_T	$7 \cdot 10^4 \frac{\text{Ns}}{\text{m}}$	d_B	$2 \cdot 10^4 \frac{\text{Ns}}{\text{m}}$
k_T	1981900 $\frac{\text{N}}{\text{m}}$	k_B	40000 $\frac{\text{N}}{\text{m}}$
α	0.02 m^{-1}	$\dot{\beta}_{\text{max}}$	$10 \cdot \frac{\pi}{180}\,\frac{\text{rad}}{\text{s}} = 0.1745\,\frac{\text{rad}}{\text{s}}$
τ	0.1 s	τ_{PT2}	0.09 s
ζ	0.6	τ_g	0.02 s
τ_v	4 s	$\omega_{1,\text{notch}}$	10.364 s^{-1}
$\omega_{2,\text{notch}}$	8 s^{-1}	$\zeta_{1,\text{notch}}$	0.6
$\zeta_{2,\text{notch}}$	1.2	$\omega_{r,\text{SP}}$	1.2671 $\frac{\text{rad}}{\text{s}}$

Table A.1: Model parameters for NREL 5 MW turbine adapted to the reduced-order wind turbine model. The generator inertia from FAST has to be multiplied by n_g^2, as a gearbox ratio is considered in the FAST simulation model.

A.2 Extraction of the Aero Maps from FAST Simulations

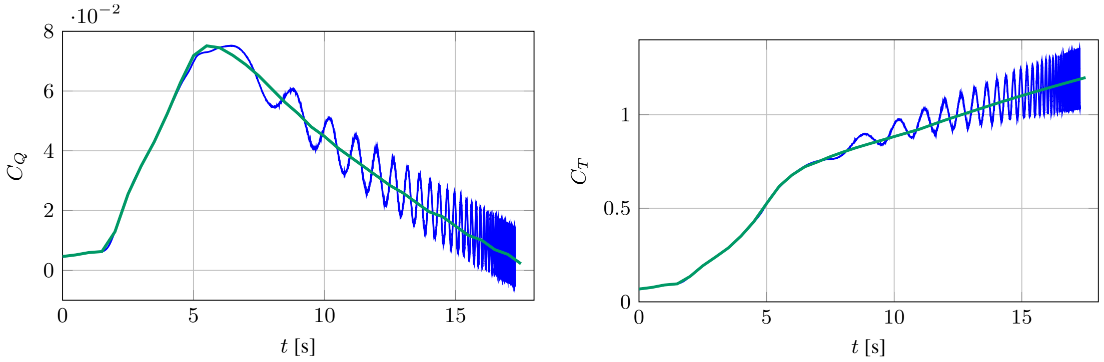
The extraction of the aero maps for the rotor torque and thrust coefficient for the NREL 5 MW reference turbine from FAST/AeroDyn simulations was done in three steps. First, load-free start-up procedures were simulated for different fixed pitch angles. For each simulation run, the thrust force $dS(r_i)$ and the circumferential force $dU(r,i)$ for each blade element were extracted from AeroDyn [Laino and Hansen, 2002] (where r_i denotes the distance from the hub cenre to the ith blade element). From this, the coefficients C_T and C_Q were calculated using the following relations [Gasch and Twele, 2012]:

$$C_T\,F_{\text{st}} = N\sum_i dS(r_i) \tag{A.1}$$

$$C_Q\,F_{\text{st}}\,R = N\sum_i dU(r_i)\,r_i\,, \tag{A.2}$$

where N denotes the number of rotor blades and F_{dyn} the reference force from equation (2.26).

This procedure is not as straighforward as extracting the thrust and torque coefficients directly from FAST, which is possible. However, wrong results are obtained this way (especially for C_Q), as other terms except for the pure aerodynamic ones are contained in the FAST outputs for C_T and C_Q.
On the other hand, the aero maps obtained from AeroDyn via summation of the forces at each blade element are superimposed with dynamical oscillatory effects (see figure A.1) starting from certain tip speed ratios. Therefore, in a second step, the obtained curves were smoothed using a running average starting from those tip speed ratios were the superposition effects were visible. Third, the curves for each pitch angle were assembled to the 2-dimensional aero maps in dependence on the tip speed ratio and the pitch angle (depicted in figures 2.6 and 2.5).



(a) Rotor torque coefficient

(b) Rotor thrust coefficient

Figure A.1: Rotor torque and thrust coefficients for a fixed pitch angle of $\beta = 0°$; blue lines: curves obtained with Equations (A.2) and (A.1) using the circumferential and thrust forces on each blade element extracted from AeroDyn; green lines: filtered curves

A.3 Derivation of Affine I-Augmented TS System Structure

In this section, it is shown that the I-augmented approximated TS system

$$\dot{\tilde{\mathbf{x}}} = \sum_{i=1}^{N_r} h_i\left(\beta\right)\left(\tilde{\mathbf{A}}_i\,\Delta\tilde{\mathbf{x}}_i\,+\,\tilde{\mathbf{B}}\,\Delta u_i\,+\,\tilde{\mathbf{E}}_i\,\Delta v_i\right) \tag{A.3}$$

can also be written in the following affine TS system structure

$$\dot{\tilde{\mathbf{x}}} = \sum_{i=1}^{N_r} h_i\left(\beta\right)\left(\tilde{\mathbf{A}}_i\,\tilde{\mathbf{x}}\,+\,\tilde{\mathbf{B}}\,u\,+\mathbf{a}_i\right)\,+\,\sum_{i=1}^{N_r} h_i\left(\beta\right)\tilde{\mathbf{E}}_i\,v\,, \tag{A.4}$$

if the PDC control law $u\,=\,-\sum_{i=1}^{N_r} h_i\left(\beta\right)\left(\mathbf{k}_i^T\,\Delta\mathbf{x}_i\,+\,k_{I,i}\,x_I\right)$ is used.

Proof The natural choice for the local subcontrollers is $\Delta u_i\,=\,-\tilde{\mathbf{k}}_i^T\,\Delta\tilde{\mathbf{x}}_i$, with $\tilde{\mathbf{k}}_i^T\,=\,\begin{pmatrix}k_i^T & k_{I,i}\end{pmatrix}$ and $\Delta\tilde{\mathbf{x}}_i = \begin{pmatrix}\Delta\mathbf{x}_i \\ \Delta x_{I,i}\end{pmatrix}$.
Thus, Equation (A.3) can be written as

$$\begin{aligned}
\dot{\tilde{\mathbf{x}}} \;&=\; \sum_{i=1}^{N_r} h_i\left(\beta\right)\left(\tilde{\mathbf{A}}_i\,\Delta\tilde{\mathbf{x}}_i\,+\,\tilde{\mathbf{B}}\,\Delta u_i\,+\,\tilde{\mathbf{E}}_i\,\Delta v_i\right)\\
&=\; \sum_{i=1}^{N_r} h_i\left(\beta\right)\left(\tilde{\mathbf{A}}_i\,\Delta\tilde{\mathbf{x}}_i\,-\,\tilde{\mathbf{B}}\,\tilde{\mathbf{k}}_i^T\,\Delta\tilde{\mathbf{x}}_i\,+\,\tilde{\mathbf{E}}_i\,\Delta v_i\right)\\
&=\; \sum_{i=1}^{N_r} h_i\left(\beta\right)\left(\tilde{\mathbf{A}}_i\,\Delta\tilde{\mathbf{x}}_i\,+\,\tilde{\mathbf{E}}_i\,\Delta v_i\right)\,-\,\tilde{\mathbf{B}}\,\sum_{i=1}^{N_r} h_i\left(\beta\right)\;\underbrace{\left(\mathbf{k}_i^T\,\Delta\mathbf{x}_i\,+\,\mathbf{k}_{I,i}\,\Delta x_{I,i}\right)}_{=\,\mathbf{k}_i^T\,\Delta\mathbf{x}_i\,+\,\mathbf{k}_{I,i}\,x_I\,-\,\mathbf{k}_{I,i}\,x_{I,i}}\\
&=\; \sum_{i=1}^{N_r} h_i\left(\beta\right)\left(\tilde{\mathbf{A}}_i\,\Delta\tilde{\mathbf{x}}_i\,+\,\tilde{\mathbf{E}}_i\,\Delta v_i\right)\,+\,\tilde{\mathbf{B}}\,\underbrace{\left(-\sum_{i=1}^{N_r} h_i\left(\beta\right)\left(\mathbf{k}_i^T\,\Delta\mathbf{x}_i\,+\,\mathbf{k}_{I,i}\,x_I\right)\right)}_{=\,u}\\
&\qquad -\,\tilde{\mathbf{B}}\sum_{i=1}^{N_r} h_i\left(\beta\right)\underbrace{\left(-\mathbf{k}_{I,i}\,x_{I,i}\right)}_{=\,u_i}\\
&=\; \sum_{i=1}^{N_r} h_i\left(\beta\right)\left(\tilde{\mathbf{A}}_i\,\tilde{\mathbf{x}}\,+\,\tilde{\mathbf{B}}\,u\,\underbrace{-\left(\tilde{\mathbf{A}}_i\,\tilde{\mathbf{x}}_i\,+\,\tilde{\mathbf{B}}\,u_i\,+\tilde{\mathbf{E}}_i\,v_i\right)}_{=\,\mathbf{a}_i}\right)\,+\,\sum_{i=1}^{N_r} h_i\left(\beta\right)\tilde{\mathbf{E}}_i\,v
\end{aligned}$$

□

A.4 Coordinate Transformations for the TS Sliding Mode Observer

In this section, the four transformations applied to obtain the transformed TS system as introduced on page 72 are described. For further details and proofs, see [Edwards and Spurgeon, 1998; Gerland et al., 2010b,a].

Consider the following TS system including disturbances $\boldsymbol{\xi}$, actuator faults $\mathbf{f}_a$ and sensor faults $\mathbf{f}_s$:

$$\begin{aligned}\dot{\mathbf{x}} &= \sum_{i=1}^{N_r} h_i(\mathbf{z})\left(\mathbf{A}_i\,\mathbf{x} + \mathbf{B}_i\,\mathbf{u} + \mathbf{D}_i\,\boldsymbol{\xi} + \mathbf{F}_i\,\mathbf{f}_a\right)\,, \\ \tilde{\mathbf{y}} &= \mathbf{y} + \mathbf{f}_s = \mathbf{C}\mathbf{x} + \mathbf{f}_s\,,\end{aligned} \tag{A.5}$$

where $\mathbf{A}_i \in \mathbb{R}^{n\times n}$, $\mathbf{B}_i \in \mathbb{R}^{n\times m}$, $\mathbf{D}_i \in \mathbb{R}^{n\times d}$, $\mathbf{F}_i \in \mathbb{R}^{n\times a}$, and $\mathbf{C} \in \mathbb{R}^{p\times n}$.

The TS sliding mode observer is designed in the transformed TS system

$$\dot{\mathbf{x}}_1 = \sum_{i=1}^{N_r} h_i(\mathbf{z})\left(\boldsymbol{\mathcal{A}}_{11,i}\,\mathbf{x}_1 + \boldsymbol{\mathcal{A}}_{12,i}\,\mathbf{y} + \boldsymbol{\mathcal{B}}_{1,i}\,\mathbf{u}\right)\,, \tag{A.6}$$

$$\dot{\tilde{\mathbf{y}}} = \sum_{i=1}^{N_r} h_i(\mathbf{z})\left(\boldsymbol{\mathcal{A}}_{21,i}\,\mathbf{x}_1 + \boldsymbol{\mathcal{A}}_{22,i}\,\mathbf{y} + \boldsymbol{\mathcal{B}}_{2,i}\,\mathbf{u} + \boldsymbol{\mathcal{D}}_{2,i}\,\boldsymbol{\xi} + \boldsymbol{\mathcal{F}}_{2,i}\,\mathbf{f}_a\right) + \dot{\mathbf{f}}_s\,, \tag{A.7}$$

where $\mathbf{x}_1$ denotes the vector of the unmeasurable states and the transformed system matrices have the following structure:

$$\boldsymbol{\mathcal{A}}_i = \mathbf{T}_i\mathbf{A}_i\mathbf{T}_i^{-1} = \begin{bmatrix}\boldsymbol{\mathcal{A}}_{11,i} & \boldsymbol{\mathcal{A}}_{12,i}\\ \boldsymbol{\mathcal{A}}_{21,i} & \boldsymbol{\mathcal{A}}_{22,i}\end{bmatrix},$$

$$\boldsymbol{\mathcal{B}}_i = \mathbf{T}_i\,\mathbf{B}_i = \begin{bmatrix}\boldsymbol{\mathcal{B}}_{1,i}\\ \boldsymbol{\mathcal{B}}_{2,i}\end{bmatrix}, \boldsymbol{\mathcal{D}}_i = \mathbf{T}_i\,\mathbf{D}_i = \begin{bmatrix}\mathbf{0}\\ \boldsymbol{\mathcal{D}}_{2,i}\end{bmatrix}, \boldsymbol{\mathcal{F}}_i = \mathbf{T}_i\,\mathbf{F}_i = \begin{bmatrix}\mathbf{0}\\ \boldsymbol{\mathcal{F}}_{2,i}\end{bmatrix}.$$

A.4.1 Existence Conditions

For a stable observer to exist, the following three existence conditions have to be fulfilled [Edwards and Spurgeon, 1998; Gerland et al., 2010b,a]:

Condition 1. The uncertainties and actuator faults are unknown but bounded: $\|\boldsymbol{\xi}^T(t)\ \mathbf{f}_a^T(t)\|^T \le \Xi$. Furthermore, individual bounds exist: $\|\boldsymbol{\xi}(t)\|^T \le \Xi_\xi$ and $\mathbf{f}_a(t) \le \Xi_{f_a}$. The sensor faults and their derivatives are assumed to be bounded, too: $\|\mathbf{f}_s\| \le \Psi$, $\|\dot{\mathbf{f}}_s\| \le \Psi_d$. Moreover, the system states and inputs are assumed to be bounded.

Condition 2. Let q be defined as the number of columns of $[\mathbf{D}_i\ \mathbf{F}_i]$. Then, the condition $q_i = \operatorname{rank}(\mathbf{C}\,[\mathbf{D}_i\ \mathbf{F}_i]) = \operatorname{rank}[\mathbf{D}_i\ \mathbf{F}_i]$ must be fulfilled. Furthermore, it must hold that $p > q_i$, where p is the number of measurable system states.

Condition 3. All invariant zeros of $(\mathbf{A}_i,\ [\mathbf{D}_i\ \mathbf{F}_i]\,,\ \mathbf{C})$ must lie in $\mathbb{C}_-$.

A further assumption is that the error vector $\mathbf{e}_1 = \hat{\mathbf{x}}_1 - \mathbf{x}_1$ of the transformed unmeasurable states is bounded: $\|\mathbf{e}_1\| < \Gamma$. When investigating the stability of the error system $\mathbf{e}_1$, an LMI design condition is obtained (see Section A.4.3 below).

A.4.2 Description of the Transformation Matrices

The transformation matrices $\mathbf{T}_i$ are composed of four sub-transformations: $\mathbf{T}_i = \mathbf{T}_{L,i}\,\mathbf{T}_{a,i}\,\mathbf{T}_{b,i}\,\mathbf{T}_c$, where each transformation fulfills a special purpose:

---

1. $\mathbf{T}_c$: Separates the system into measurable and unmeasurable system states
2. $\mathbf{T}_{b,i}$: Generates a special structure in the disturbance, fault and output matrices, such that the unknown disturbances and faults only act on the measurable states
3. $\mathbf{T}_{a,i}$: Transforms the system matrix $\mathbf{A}_i$ into a special structure such that a submatrix is always observable
4. $\mathbf{T}_{L,i}$: Introduces a design matrix $\mathbf{L}_i$ in order to stabilise the observer

---

1. $\mathbf{T}_c$

The first transformation is defined by $\mathbf{T}_c := \begin{bmatrix}\mathbf{N}_c^T \\ \mathbf{C}\end{bmatrix}$, where the columns of $\mathbf{N}_c \in \mathbb{R}^{n\times(n-p)}$ span the null space of $\mathbf{C}$, i.e. $\mathbf{C}\,\mathbf{N}_c = \mathbf{0}$. In the new coordinate system, the output matrix is then given by $\mathbf{C}_c = \mathbf{C}\,\mathbf{T}_c^{-1} = \mathbf{C}\begin{bmatrix}\mathbf{N}_c & \mathbf{C}^T\end{bmatrix} = [\mathbf{0}\ \mathbf{I}_{p\times p}]$, which shows that $\mathbf{T}_c$ in fact achieves a separation of the measurable states $\mathbf{y}$ and the non-measurable states $\mathbf{x}_1$, such that the state vector in the new coordinate system is given by $\mathbf{x}_c = \begin{pmatrix}\mathbf{x}_1 \\ \mathbf{y}\end{pmatrix}$.

The other transformed matrices in the new coordinate system are given by

$$\mathbf{A}_{c,i} = \mathbf{T}_c\,\mathbf{A}_i\,\mathbf{T}_c^{-1}, \quad \mathbf{B}_{c,i} = \mathbf{T}_c\,\mathbf{B}_i, \quad \mathbf{D}_{c,i} = \mathbf{T}_c\,\mathbf{D}_i, \quad \mathbf{F}_{c,i} = \mathbf{T}_c\,\mathbf{F}_i$$

2. $\mathbf{T}_{b,i}$

To apply the second transformation, the combined disturbance and actuator fault matrix is first partitioned conformably to the structure introduced by $\mathbf{T}_c$, i.e.

$[\mathbf{D}_{c,i}\quad \mathbf{F}_{c,i}] = \begin{bmatrix}\mathbf{D}_{c,1,i} & \mathbf{F}_{c,1,i} \\ \mathbf{D}_{c,2,i} & \mathbf{F}_{c,2,i}\end{bmatrix}$, where $\mathbf{D}_{c,2,i} \in \mathbb{R}^{p\times d}$ and $\mathbf{F}_{c,2,i} \in \mathbb{R}^{p\times a}$.

The transformation is defined by

$$\mathbf{T}_{b,i} = \begin{bmatrix}\mathbf{I}_{n-p} & -[\mathbf{D}_{c,1,i}\ \mathbf{F}_{c,1,i}]\left([\mathbf{D}_{c,2,i}\ \mathbf{F}_{c,2,i}]^T\,[\mathbf{D}_{c,2,i}\ \mathbf{F}_{c,2,i}]\right)^{-1}[\mathbf{D}_{c,2,i}\ \mathbf{F}_{c,2,i}]^T \\ \mathbf{0}_{p\times n-p} & \mathbf{T}^T\end{bmatrix},$$

where $\mathbf{T}$ is obtained from a QR decomposition of $[\mathbf{D}_{c,2,i}\ \mathbf{F}_{c,2,i}]$ and subsequent flipping of the rows of the transposed matrix: $\mathbf{T}_{\text{temp}} = \text{qr}([\mathbf{D}_{c,2,i}\ \mathbf{F}_{c,2,i}])$, $\mathbf{T} = \text{flipud}(\mathbf{T}_{\text{temp}}^T)$ (MATLAB commands).

One can easily verify that by applying $\mathbf{T}_{b,i}$ to $[\mathbf{D}_{c,i}\ \mathbf{F}_{c,i}]$, only zeros are obtained in the top rows of the transformed matrix:

$$[\mathbf{D}_{b,i}\ \mathbf{F}_{b,i}] := \mathbf{T}_{b,i}\,[\mathbf{D}_{c,i}\ \mathbf{F}_{c,i}] = \begin{bmatrix}\mathbf{0}_{n-p\times d+a} \\ \mathbf{T}^T\,[\mathbf{D}_{c,2,i}\ \mathbf{F}_{c,2,i}]\end{bmatrix}.$$

The other transformed matrices in the new coordinate system are given by

$$\mathbf{A}_{b,i} = \mathbf{T}_{b,i}\,\mathbf{A}_{c,i}\,\mathbf{T}_{b,i}^{-1}, \quad \mathbf{B}_{b,i} = \mathbf{T}_{b,i}\,\mathbf{B}_{c,i}, \quad \mathbf{C}_b = \mathbf{C}_c\,\mathbf{T}_{b,i}^{-1} = [\mathbf{0}\ \mathbf{T}].$$

3. $\mathbf{T}_{a,i}$

The matrix $\mathbf{A}_{b,i}$ can be written as

$$\mathbf{A}_{b,i} = \left[\begin{array}{c|c} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \hline \mathbf{A}_{211} & \\ \mathbf{A}_{212} & \mathbf{A}_{22} \end{array}\right].$$

If the pair $(\mathbf{A}_{b,i},\, \mathbf{C}_b)$ is not completely observable, the pair $(\mathbf{A}_{11},\, \mathbf{A}_{211})$ can be brought into the observability canonical form using the MATLAB command *obsvf*, which yields the following structure of the two submatrices:

$\bar{\mathbf{A}}_{11} = \begin{bmatrix} \mathbf{A}_{11}^0 & \mathbf{A}_{12}^0 \\ \mathbf{0} & \mathbf{A}_{22}^0 \end{bmatrix}$, $\bar{\mathbf{A}}_{211} = \begin{bmatrix} \mathbf{0} & \mathbf{A}_{21}^0 \end{bmatrix}$ by means of a sub-transformation matrix $\mathbf{T}_{\text{obs}}$.

The complete transformation is defined by $\mathbf{T}_{a,i} = \begin{bmatrix} \mathbf{T}_{\text{obs}} & \mathbf{0}_{n-p\times p} \\ \mathbf{0}_{p\times n-p} & \mathbf{I}_p \end{bmatrix}$ and the transformed system matrices in the new coordinate system are given by

$$\mathbf{A}_{a,i} = \mathbf{T}_{a,i}\,\mathbf{A}_{b,i}\,\mathbf{T}_{a,i}^{-1}, \quad \mathbf{B}_{a,i} = \mathbf{T}_{a,i}\,\mathbf{B}_{b,i}, \quad \mathbf{C}_a = \mathbf{C}_b\,\mathbf{T}_{a,i}^{-1}, \quad [\mathbf{D}_{a,i}\;\; \mathbf{F}_{a,i}] := \mathbf{T}_{a,i}\,[\mathbf{D}_{b,i}\;\; \mathbf{F}_{b,i}]$$

4. $\mathbf{T}_{L,i}$

The last transformation introduces a gain matrix $\mathbf{L}_i$ to stabilise the error dynamics of the unmeasurable states and is given by

$\mathbf{T}_{L,i} = \begin{bmatrix} \mathbf{I}_{n-p} & \bar{\mathbf{L}}_i \\ \mathbf{0}_{p\times n-p} & \mathbf{T} \end{bmatrix}$, where $\mathbf{T}$ is the same matrix as obtained from the QR decomposition for the second transformation $\mathbf{T}_{b,i}$ and $\bar{\mathbf{L}}_i = [\mathbf{L}_i \quad \mathbf{0}_{n-p\times q_i}]$.

The final transformed system matrices in the new coordinates are defined by

$$\boldsymbol{\mathcal{A}}_i = \mathbf{T}_{L,i}\,\mathbf{A}_{a,i}\,\mathbf{T}_{L,i}^{-1}, \quad \boldsymbol{\mathcal{B}}_i = \mathbf{T}_{L,i}\,\mathbf{B}_{a,i}, \quad \boldsymbol{\mathcal{C}} = \mathbf{C}_a\,\mathbf{T}_{L,i}^{-1}, \quad [\boldsymbol{\mathcal{D}}_i\;\; \boldsymbol{\mathcal{F}}_i] := \mathbf{T}_{L,i}\,[\mathbf{D}_{a,i}\;\; \mathbf{F}_{a,i}].$$

$\mathbf{L}_i$ is designed such as to yield a stable sub-matrix $\boldsymbol{\mathcal{A}}_{11,i} = \bar{\mathbf{A}}_{11} + \mathbf{L}_i\,\bar{\mathbf{A}}_{211,i}$ by means of an LMI condition (see the following section).

A.4.3 LMI Design Condition for $\mathbf{L}_i$

From equations (6.3) and (6.5), the dynamics of the error $\mathbf{e}_1$ without sensor faults can be calculated as

$$\dot{\mathbf{e}}_1 = \sum_{i=1}^{N_r} h_i(\mathbf{z})\, \boldsymbol{\mathcal{A}}_{11,i}\, \mathbf{e}_1 \,. \tag{A.8}$$

A Lyapunov function candidate for the error $\mathbf{e}_1$ is $V(\mathbf{e}_1) = \mathbf{e}_1^T\,\mathbf{P}_1\mathbf{e}_1$, with a symmetric positive definite matrix $\mathbf{P}_1$.

For V to be a Lyapunov function, both $V > 0$ and $\dot{V} < 0$ have to be fulfilled.

Differentiating V and using the relation $\boldsymbol{\mathcal{A}}_{11,i} = \bar{\mathbf{A}}_{11,i} + \mathbf{L}_i\,\bar{\mathbf{A}}_{211,i}$, one obtains

$$\begin{aligned} \dot{V}(\mathbf{e}_1) &= \dot{\mathbf{e}}_1^T\,\mathbf{P}_1\,\mathbf{e}_1 + \mathbf{e}_1^T\,\mathbf{P}_1\,\dot{\mathbf{e}}_1 \\ &= \sum_{i=1}^{N_r} h_i(\mathbf{z})\; \mathbf{e}_1^T \left(\boldsymbol{\mathcal{A}}_{11,i}^T\,\mathbf{P}_1 + \mathbf{P}_1\,\boldsymbol{\mathcal{A}}_{11,i}\right) \mathbf{e}_1 \\ &= \sum_{i=1}^{N_r} h_i(\mathbf{z})\; \mathbf{e}_1^T \left(\left(\bar{\mathbf{A}}_{11,i}^T + \bar{\mathbf{A}}_{211,i}^T\,\mathbf{L}_i^T\right)\mathbf{P}_1 + \mathbf{P}_1\left(\bar{\mathbf{A}}_{11,i} + \mathbf{L}_i\,\bar{\mathbf{A}}_{211,i}\right)\right) \mathbf{e}_1 \end{aligned}$$

(A.9)

Defining $\mathbf{N}_i := \mathbf{P}_1\mathbf{L}_i$, the condition $V < 0$ leads to the following LMI conditions:

$$\begin{aligned} \mathbf{P}_1 &> \mathbf{0}\,, && \text{(A.10)} \\ \bar{\mathbf{A}}_{11,i}^T\mathbf{P}_1 + \mathbf{P}_1\bar{\mathbf{A}}_{11,i} + \bar{\mathbf{A}}_{211,i}^T\,\mathbf{N}_i^T + \mathbf{N}_i\,\bar{\mathbf{A}}_{211,i} &< \mathbf{0}\,. && \text{(A.11)} \end{aligned}$$

After solving the above LMIs with a suitable solver (for example LMI Lab [Gahinet et al., 1994]), the desired gain matrix can be simply calculated as $\mathbf{L}_i = \mathbf{P}_1^{-1}\,\mathbf{N}_i$.

In order to achieve a faster convergence of the error towards zero, one can optionally include a decay rate $\alpha > 0$ into the design condition:

$$\begin{aligned} \mathbf{P}_1 &> \mathbf{0}\,, && \text{(A.12)} \\ \bar{\mathbf{A}}_{11,i}^T\mathbf{P}_1 + \mathbf{P}_1\bar{\mathbf{A}}_{11,i} + \bar{\mathbf{A}}_{211,i}^T\,\mathbf{N}_i^T + \mathbf{N}_i\,\bar{\mathbf{A}}_{211,i} &< -2\,\alpha\,\mathbf{P}_1\,. && \text{(A.13)} \end{aligned}$$

Bibliography

Akhenak, A., M. Chadli, J. Ragot, and D. Maquin (2008). Fault detection and isolation using sliding mode observer for uncertain Takagi-Sugeno fuzzy model. In *Mediterranean Conference on Control and Automation*, Ajaccio, France, pp. 286–291.

Akhenak, A., Chadli, M., Ragot, J, and D. Maquin (2007). Design of Sliding Mode Unknown Input Observer for Uncertain Takagi-Sugeno Model. In *Mediterranean Conference on Control and Automation*, Athens, Greece.

Alwi, H., C. Edwards, and C. P. Tan (2011). *Fault Detection and Fault-Tolerant Control Using Sliding Modes*. Advances in Industrial Control. Springer-Verlag London Limited.

Amlang, B., D. Arsudis, W. Leonhard, W. Vollstedt, and K. Wefelmeier (1992). Elektrische Energieversorgung mit Windkraftanlagen. Technical report, BMFT 032-8265-B, Technische Universität Braunschweig.

Badihi, H. and Y. Zhang (2013). Fault-Tolerant Control Design for a Large Off-Shore Wind Turbine Using Fuzzy Gain-Scheduling and Signal Correction. In *American Control Conference*, Washington, D.C., USA, pp. 1448–1453.

Bergsten, P., R. Palm, and D. Driankov (2001). Fuzzy Observers. In *IEEE International Conference on Fuzzy Systems*, Melbourne, Australia, pp. 700–703.

Bergsten, P., R. Palm, and D. Driankov (2002). Observers for Takagi-Sugeno Fuzzy Systems. *IEEE Transactions on Systems, Man, and Cybernetics 32*(1), 114–121.

Bianchi, F. D., H. De Battista, and R. J. Mantz (2007). *Wind Turbine Control Systems - Principles, Modelling and Gain Scheduling Design*. Advances in Industrial Control. Springer-Verlag, London Limited.

Bindner, H. (1999). Active Control: Wind Turbine Model. Technical report, Risø-R-920(EN), Risø National Laboratory, Roskilde, Denmark.

Bir, G. (2012). NWTC Design Codes (BModes by Gunjit Bir). http://wind.nrel.gov/designcodes/preprocessors/bmodes.

Bir, G. and J. Jonkman (2007). Aeroelastic Instabilities of Large Offshore and Onshore Wind Turbines. *Journal of Physics: Conference Series 75*(012069), 1 – 19.

Blanke, M., M. Kinnaert, J. Lunze, and M. Staroswiecki (2006). *Diagnosis and Fault-Tolerant Control* (2nd ed.). Springer-Verlag Berlin Heidelberg.

Blesa, J., V. Puig, J. Romera, and J. Saludes (2011). Fault Diagnosis of Wind Turbines using a Set-membership Approach. In *IFAC World Congress*, Milano, Italy, pp. 8316–8321.

Bossanyi, E. A. (2000). The Design of Closed Loop Controllers for Wind Turbines. *Wind Energy 3*(3), 149–163.

Bossanyi, E. A., A. Wright, and P. Fleming (2010a). Further progress with field testing of individual pitch control. In *European Wind Energy Conference*, Warsaw, Poland.

Bossanyi, E. A., A. Wright, and P. Fleming (2010b). Progress with field testing of individual pitch control. In *The Science of Making Torque from Wind*, Heraklion, Crete, Greece.

Boukhezzar, B. and H. Siguerdidjane (2005). Nonlinear Control of Variable Speed Wind Turbines without wind speed measurement. In *IEEE Conference on Decision and Control and the European Control Conference*, Sevilla, Spain, pp. 3456–3461.

Boukhezzar, B. and H. Siguerdidjane (2010). Comparison between linear and nonlinear control strategies for variable speed wind turbines. *Control Engineering Practice 18*(12), 1357–1368.

Burton, T., N. Jenkins, D. Sharpe, and E. Bossanyi (2011). *Wind Energy Handbook* (2nd ed.). John Wiley & Sons, Ltd.

Casau, P., P. Rosa, S. M. Tabatabaeipour, and C. Silvestre (2012). Fault Detection and Isolation and Fault Tolerant Control of Wind Turbines using Set-Valued Observers. In *IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*, Mexico City, Mexico, pp. 120–125.

Chadli, M. and A. El Hajjaji (2010). Wind energy conversion systems control using T-S fuzzy modeling. In *Mediterranean Conference on Control and Automation*, Marrakech, Morocco.

Chen, J. and R. J. Patton (1999). *Robust Model-Based Fault Diagnosis for Dynamic Systems*. Kluwer Academic Publishers.

Chen, J., R. J. Patton, and H. Y. Zhang (1996). Design of Unknown Input Observers and Robust Fault-Detection Filters. *International Journal of Control 63*(1), 85–105.

Chen, J. and H. Y. Zhang (1991). Robust Detection of Faulty Actuators via Unknown Input Observers. *International Journal of Systems Science 22*(10), 1829–1839.

Chen, W., S. X. Ding, A. Haghani, A. Naik, A. Q. Khan, and S. Yin (2011). Observer-based FDI Schemes for Wind Turbine Benchmark. In *IFAC World Congress*, Milano, Italy, pp. 7073–7078.

Chow, E. Y. and A. S. Willsky (1984). Analytical Redundancy and the Design of Robust Failure Detection Systems. *IEEE Transactions on Automatic Control 29*(7), 603–614.

Corradini, M. L., G. Ippoliti, and G. Orlando (2013). Robust Control of Variable-Speed Wind Turbines Based on an Aerodynamic Torque Observer. *IEEE Transactions on Control Systems Technology 21*(4), 1199–1206.

Dolan, B. and H. Aschemann (2013). Nonlinear Control for Load Reduction on a Variable Speed Pitch Regulated Wind Turbine. In *International Conference on Methods and Models in Automation and Robotics (MMAR)*, Międzyzdroje, Poland, pp. 753 – 758.

Dong, J. and M. Verhaegen (2011). Data Driven Fault Detection and Isolation of a Wind Turbine Benchmark. In *IFAC World Congress*, Milano, Italy, pp. 7086–7091.

Echavarria, E., van Bussel, G. J. W., and T. Tomiyama (2012). Finding functional redundancies in offshore wind turbine design. *Wind Energy 15*(4), 609–626.

Edwards, C. (2004). A comparison of sliding mode and unknown input observers for fault reconstruction. In *IEEE Conference on Decision and Control*, Atlantis, Paradise Island, Bahamas, pp. 5279–5284.

Edwards, C. and S. K. Spurgeon (1994). On the development of discontinuous observers. *International Journal of Control 59*(5), 1211–1229.

Edwards, C. and S. K. Spurgeon (1998). *Sliding Mode Control: Theory and Applications*. Taylor & Francis, Boca Raton.

Edwards, C., S. K. Spurgeon, and R. J. Patton (2000). Sliding mode observers for fault detection and isolation. *Automatica 36*, 541–553.

Edwards, C. and C. P. Tan (2006a). A Comparison of Sliding Mode and Unknown Input Observers for Fault Reconstruction. *European Journal of Control 12*, 245–260.

Edwards, C. and C. P. Tan (2006b). Sensor fault tolerant control using sliding mode observers. *Control Engineering Practice 14*, 897–908.

Ekelund, T. (1994). Speed Control of Wind Turbines in the Stall Region. In *IEEE Conference on Control Applications*, Glasgow, UK, pp. 227–232.

Ekelund, T. (1997). *Modeling and Linear Quadratic Optimal Control of Wind Turbines*. Ph. D. thesis, Chalmers University of Technology.

Franklin, G. F., J. D. Powell, and A. Emami-Naeini (1994). *Feedback Control of Dynamic Systems*. Addison-Wesley Publishing Company, Inc.

Gahinet, P., A. Nemirovskii, A. J. Laub, and M. Chilali (1994). The LMI Control Toolbox. In *IEEE Conference on Decision and Control*, Lake Buena Vista, USA, pp. 2038–2041.

Galdi, V., A. Piccolo, and P. Siano (2008). Designing an Adaptive Fuzzy Controller for Maximum Wind Energy Extraction. *IEEE Transactions on Energy Conversion 23*(2), 559 – 569.

Galdi, V., A. Piccolo, and P. Siano (2009). Exploiting maximum energy from variable speed wind power generation systems by using an adaptive Takagi-Sugeno-Kang Fuzzy Model. *Energy Conversion and Management 50*(2), 413 – 421.

Gasch, R. and J. Twele (Eds.) (2012). *Wind Power Plants* (2nd ed.). Springer-Verlag, Berlin, Heidelberg.

Georg, S., S. Heyde, and H. Schulte (2014). Sensor Fault-Tolerant Control of a Drivetrain Test Rig via an Observer-Based Approach within a Wind Turbine Simulation Model. *Journal of Physics: Conference Series 570*(082004), 1 – 11.

Georg, S., M. Müller, and H. Schulte (2014). Wind Turbine Model and Observer in Takagi-Sugeno Model Structure. *Journal of Physics: Conference Series 555*(012042), 1 – 11.

Georg, S. and H. Schulte (2013). Actuator Fault Diagnosis and Fault-Tolerant Control of Wind Turbines using a Takagi-Sugeno Sliding Mode Observer. In *International Conference on Control and Fault-Tolerant Systems*, Nice, France, pp. 516–522.

Georg, S. and H. Schulte (2014a). Diagnosis of Actuator Parameter Faults in Wind Turbines Using a Takagi-Sugeno Sliding Mode Observer. In J. Korbicz and M. Kowal (Eds.), *Intelligent Systems in Technical and Medical Diagnostics*, Volume 230 of *Advances in Intelligent Systems and Computing*, pp. 29–41. Springer-Verlag Berlin Heidelberg.

Georg, S. and H. Schulte (2014b). Fault-Tolerant Control of Wind Turbines using a Takagi-Sugeno Sliding Mode Observer. *Journal of Physics: Conference Series 524*(012053), 1 – 10.

Georg, S. and H. Schulte (2014c). Takagi-Sugeno Sliding Mode Observer with a Weighted Switching Action and Application to Fault Diagnosis for Wind Turbines. In J. Korbicz and M. Kowal (Eds.), *Intelligent Systems in Technical and Medical Diagnostics*, Volume 230 of *Advances in Intelligent Systems and Computing*, pp. 41–52. Springer-Verlag Berlin Heidelberg.

Georg, S., H. Schulte, and H. Aschemann (2012). Control-Oriented Modelling of Wind Turbines Using a Takagi-Sugeno Model Structure. In *IEEE International Conference on Fuzzy Systems*, Brisbane, Australia, pp. 1737–1744.

Gerland, P. (2011). *Klassifikationsgestützte on-line Adaption eines robusten beobachterbasierten Fehlerdiagnoseansatzes für nichtlineare Systeme*. Ph. D. thesis, Universität Kassel.

Gerland, P., D. Groß, H. Schulte, and A. Kroll (2010a). Design of Sliding Mode Observers for TS Fuzzy Systems with Application to Disturbance and Actuator Fault Estimation. In *IEEE Conference on Decision and Control*, Atlanta, USA, pp. 4373–4378.

Gerland, P., D. Groß, H. Schulte, and A. Kroll (2010b). Robust Adaptive Fault Detection Using Global State Information and Application to Mobile Working Machines. In *International Conference on Control and Fault-Tolerant Systems*, Nice, France, pp. 813–818.

GL Renewables Certification (2010). Rules and Guidelines, IV Industrial Services, 1 Guideline for the Certification of Wind Turbines (Edition 2010). Technical report, Germanischer Lloyd, Hamburg and Germany.

Groß, D. (2010). Fault Detection for a Class of Uncertain Nonlinear Systems Using Global State Information. Master's thesis, University of Kassel, Department of Measurement and Control.

Guan, Y. and M. Saif (1991). A Novel Approach to the Design of Unknown Input Observers. *IEEE Transactions on Automatic Control 36*(5), 632–635.

Hameed, Z., Y. S. Yong, Y. M. Cho, S. H. Ahn, and C. K. Song (2009). Condition monitoring and fault detection of wind turbines and related algorithms: A review. *Renewable & Sustainable Energy Reviews 13*, 1–39.

Hansen, Martin O. L. (2008). *Aerodynamics of Wind Turbines* (2nd ed.). Earthscan.

Heier, S. (2006). *Grid Integration of Wind Energy Conversion Systems*. John Wiley & Sons Ltd.

Henriksen, L. C., H. H. Niemann, and N. K. Poulsen (2011). Detecting Asymmetries in the Rotor of a Wind Turbine using the Multi-Blade Coordinate Transformation. In *International Conference on Diagnostics of Processes and Systems*, Zamość, Poland, pp. 57–66.

Henriksen, L. C., H. H. Niemann, and N. K. Poulsen (2012). Fault diagnosis of a Wind Turbine Rotor using a Multi-blade Coordinate Framework. In *IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*, Mexico City, Mexico, pp. 37–42.

Isermann, R. (1984). Process Fault Detection Based on Modeling and Estimation Methods - A Survey. *Automatica 20*(4), 387–404.

Isermann, R. (2006). *Fault-Diagnosis Systems: An Introduction from Fault Detection to Fault Tolerance*. Springer-Verlag Berlin Heidelberg.

Isermann, R. and P. Ballé (1997). Trends in the Application of Model-Based Fault Detection and Diagnosis of Technical Processes. *Control Engineering Practice 5*(5), 709–719.

Jain, T., J. J. Yamé, and D. Sauter (2013). A real-time projection-based approach for fault accomodation in NREL's 5MW Wind Turbine System. In *American Control Conference*, Washington, D.C., USA, pp. 4471–4476.

Johnson, K. E. and P. A. Fleming (2011). Development, implementation, and testing of fault detection strategies on the National Wind Technology Center's controls advanced research turbines. *Mechatronics 21*, 728–736.

Johnson, P. A., L. Y. Pao, M. J. Balas, and L. J. Fingersh (2006). Control of Variable-Speed Wind Turbines - Standard and Adaptive Techniques for Maximizing Energy Capture. *IEEE Control Systems Magazine 26*(3).

Jonkman, J., S. Butterfield, W. Musial, and G. Scott (2009). Definition of a 5-MW Reference Wind Turbine for Offshore System Development. Technical report, NREL/TP-500-38060, National Renewable Energy Laboratory, Golden, Colorado.

Jonkman, J. M. and M. L. Buhl Jr. (2005). FAST User's Guide. Technical report, NREL/EL-500-38230, National Renewable Energy Laboratory, Golden, Colorado.

Kaiser, K. (2000). *Luftkraftverursachte Steifigkeits- und Dämpfungsmatrizen von Windturbinen und ihr Einfluß auf das Stabilitätsverhalten*. VDI-Fortschritt-Berichte, Nr. 294, VDI-Verlag DÃ¼sseldorf.

Kaiser, K. and R. Gasch (1996). The Influence of Aerodynamic Damping and Stiffness on the Vibrational Behaviour of Wind Turbines. In *European Wind Energy Conference*, Gothenburg, Sweden.

Kelley, N. NWTC Computer-Aided Engineering Tools (TurbSim by Neil Kelley, Bonnie Jonkman). http://wind.nrel.gov/designcodes/preprocessors/turbsim.

Khalil, H. K. (2002). *Nonlinear Systems*. Prentice Hall.

Kiasi, F., J. Prakash, S. L. Shah, and J. M. Lee (2011). Fault Detection and Isolation of a Benchmark Wind Turbine using the Likelihood Ratio Test. In *IFAC World Congress*, Milano, Italy, pp. 7079–7085.

Kim, J., I. Yang, and D. Lee (2012). Control Allocation based Compensation for Faulty Blade Actuator of Wind Turbine. In *IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*, Mexico City, Mexico, pp. 355–360.

Laino, D. J. NWTC Computer-Aided Engineering Tools (IECWind by Dr. David J. Laino). http://wind.nrel.gov/designcodes/preprocessors/iecwind.

Laino, D. J. and A. C. Hansen (2002). User's Guide to the Wind Turbine Aerodynamics Computer Software AeroDyn. Technical report, Windward Engineering LC, Prepared for the National Renewable Energy Laboratory under Subcontract No. TCX-9-29209-01.

Landau, L. D. and E. M. Lifshitz (1976). *Course of Theoretical Physics, Volume 1 - Mechanics*. Elsevier Butterworth-Heinemann.

Laouti, N., N. Sheibat-Othman, and S. Othman (2011). Support Vector Machines for Fault Detection in Wind Turbines. In *IFAC World Congress*, Milano, Italy, pp. 7067–7072.

Leithead, W. E. and B. Connor (2000). Control of variable speed wind turbines: Design task. *International Journal of Control 73*(13), 1189–1212.

Lendek, Z., T. M. Guerra, R. Babuška, and B. De Schutter (2010). *Stability Analysis and Nonlinear Observer Design Using Takagi-Sugeno Fuzzy Models*. Springer-Verlag Berlin Heidelberg.

Luenberger, D. G. (1964). Observing the State of a Linear System. *IEEE Transactions on Military Electronics 8*(2), 74–80.

Luenberger, D. G. (1971). An Introduction to Observers. *IEEE Transactions on Automatic Control AC-16*(6), 596–602.

Luo, H., S. X. Ding, A. Haghani, H. Hao, S. Yin, and T. Jeinsch (2013). Data-Driven Design of KPI-related Fault-Tolerant Control System for Wind Turbines. In *American Control Conference*, Washington, D.C., USA, pp. 4465–4470.

Lyapunov, A. M. (1992). The general problem of the stability of motion. *International Journal of Control 55*(3), 531–773.

Ma, X. (1997). *Adaptive Extremum Control and Wind Turbine Control*. Ph. D. thesis, Informatics and Mathematical Modelling, Technical University of Denmark, DTU.

Ma, X., N. K. Poulsen, and H. Bindner (1995). Estimation of Wind Speed in Connection to a Wind Turbine. Technical report, The Technical University of Denmark.

Mikkelsen, T., N. Angelou, K. Hansen, M. Sjöholm, M. Harris, C. Slinger, and P. Hadley (2013). A spinner-integrated wind lidar for enhanced wind turbine control. *Wind Energy 16*, 625–643.

Munteanu, I., A. I. Bratcu, N.-A. Cututulis, and E. Ceangă (2008). *Optimal Control of Wind Energy Systems - Towards a Global Approach*. Springer-Verlag, London.

Nourdine, S., H. Camblong, I. Vechiu, and G. Tapia (2010). Comparison of Wind Turbine LQG Controllers Using Individual Pitch Control to Alleviate Fatigue Loads. In *Mediterranean Conference on Control and Automation*, Marrakech, Morocco, pp. 1591–1596.

Odgaard, P. F., C. Damgaard, and R. Nielsen (2008). On-Line Estimation of Wind Turbine Power Coefficients Using Unknown Input Observers. In *IFAC World Congress*, Seoul, Korea, pp. 10646–10651.

Odgaard, P. F. and K. E. Johnson (2013). Wind turbine fault detection and fault tolerant control - an enhanced benchmark challenge. In *American Control Conference*, Washington, D.C., USA, pp. 4447–4452.

Odgaard, P. F. and J. Stoustrup (2012a). Fault Tolerant Control of Wind Turbines Using Unknown Input Observers. In *IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*, Mexico City, Mexico, pp. 313–318.

Odgaard, P. F. and J. Stoustrup (2012b). Results of a Wind Turbine FDI Competition. In *IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*, Mexico City, Mexico, pp. 102–107.

Odgaard, P. F., J. Stoustrup, and M. Kinnaert (2009). Fault Tolerant Control of Wind Turbines - a benchmark model. In *IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*, Barcelona, Spain, pp. 155–160.

Odgaard, P. F., J. Stoustrup, and M. Kinnaert (2013). Fault-Tolerant Control of Wind Turbines: A Benchmark Model. *IEEE Transactions on Control Systems Technology 21*(4), 1168–1182.

Ohtake, H., K. Tanaka, and H. O. Wang (2001). Fuzzy Modeling via Sector Nonlinearity Concept. In *Joint 9th IFSA World Congress and 20th NAFIPS International Conference*, Vancouver, Canada, pp. 127–132.

Østergaard, K. Z., P. Brath, and J. Stoustrup (2007). Estimation of effective wind speed. *Journal of Physics: Conference Series 75*(012082).

Ozdemir, A. A., P. Seiler, and G. J. Balas (2011). Wind Turbine Fault Detection Using Counter-Based Residual Thresholding. In *IFAC World Congress*, Milano, Italy, pp. 8289–8294.

Patton, R. J. (1997). Fault-Tolerant Control: The 1997 Situation. In *IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*, Hull, UK, pp. 1033–1055.

Patton, R. J. and J. Chen (1992). Robust Fault Detection of Jet Engine Sensor Systems using Eigenstructure Assignment. *Journal of Guidance, Control, and Dynamics 15*(6), 1491–1497.

Patton, R. J. and J. Chen (1997). Observer-Based Fault Detection and Isolation: Robustness and Applications. *Control Engineering Practice 5*(5), 671–682.

Patton, R. J., S. W. Willcox, and S. J. Winter (1987). Parameter-Insensitive Technique for Aircraft Sensor Fault Analysis. *Journal of Guidance, Control, and Dynamics 10*(4), 359–367.

Pisu, P. and B. Ayalew (2011). Robust Fault Diagnosis for a Horizontal Axis Wind Turbine. In *IFAC World Congress*, Milano, Italy, pp. 7055–7060.

Pöschke, F., S. Georg, and H. Schulte (2014). Fault Reconstruction using a Takagi-Sugeno Sliding Mode Observer for the Wind Turbine Benchmark. In *UKACC International Conference on Control*, Loughborough, UK, pp. 464–469.

Rotondo, D., F. Nejjari, V. Puig, and J. Blesa (2012). Fault Tolerant Control of the Wind Turbine Benchmark using Virtual Sensors/Actuators. In *IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*, Mexico City, Mexico, pp. 114–119.

Saif, M. and Y. Guan (1993). A New Approach to Robust Fault Detection and Identification. *IEEE Transactions on Aerospace and Electronic Systems 29*(3), 685–695.

Sami, M. and R. J. Patton (2012a). An FTC Approach to Wind Turbine Power Maximisation via T-S Fuzzy Modelling and Control. In *IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*, Mexico City, Mexico, pp. 349–354.

Sami, M. and R. J. Patton (2012b). Fault tolerant adaptive sliding mode controller for wind turbine power maximisation. In *IFAC Symposium on Robust Control Design*, Aalborg, Denmark, pp. 499–504.

Sami, M. and R. J. Patton (2012c). Fault Tolerant Output Feedback Tracking Control for Nonlinear Systems via T-S Fuzzy Modelling. In *IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*, Mexico City, Mexico, pp. 999–1004.

Sami, M. and R. J. Patton (2012d). Global wind turbine FTC via T-S fuzzy modelling and control. In *IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*, Mexico City, Mexico, pp. 325–330.

Schulte, H. (2006). *Approximative Modellierung, Systemidentifikation und Reglerentwurf mittels gewichteter Kombination lokaler Zustandsraummodelle am Beispiel fluidischer Antriebe*. Kassel University Press.

Schulte, H. and S. Georg (2014). Coordinate Transformations of Takagi-Sugeno Systems: Stability Condition and Observer Canonical Forms. In *IEEE International Conference on Fuzzy Systems*, Beijing, China, pp. 2472–2476.

Schulte, H., M. Zając, and P. Gerland (2012). Takagi-Sugeno Sliding Mode Observer Design for Fault Diagnosis in Pitch Control Systems of Wind Turbines. In *IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*, Mexico City, Mexico, pp. 546–551.

Sheibat-Othman, N., S. Othman, M. Benlahrache, and P. F. Odgaard (2013). Fault detection and isolation in wind turbines using support vector machines and observers. In *American Control Conference*, Washington, D.C., USA, pp. 4459–4464.

Simani, S. and P. Castaldi (2012a). Adaptive Fault-Tolerant Control Design Approach for a Wind Turbine Benchmark. In *IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*, Mexico City, Mexico, pp. 319–324.

Simani, S. and P. Castaldi (2012b). Data-Driven Design of Fuzzy Logic Fault Tolerant Control for a Wind Turbine Benchmark. In *IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*, Mexico City, Mexico, pp. 108–113.

Simani, S., P. Castaldi, and M. Bonfè (2011). Hybrid Model-Based Fault Detection of Wind Turbine Sensors. In *IFAC World Congress*, Milano, Italy, pp. 7061–7066.

Simani, S., P. Castaldi, and A. Tilli (2011). Data-Driven Approach for Wind Turbine Actuator and Sensor Fault Detection and Isolation. In *IFAC World Congress*, Milano, Italy, pp. 8301–8306.

Sloth, C., T. Esbensen, and J. Stoustrup (2011). Robust and fault-tolerant linear parameter-varying control of wind turbines. *Mechatronics 21*, 645–659.

Soltani, M. N., T. Knudsen, M. Svenstrup, R. Wisniewski, P. Brath, and R. Ortega (2013). Estimation of Rotor Effective Wind Speed: A Comparison. *IEEE Transactions on Control Systems Technology 21*(4), 1155–1167.

Sontag, E. D. (1989). Smooth Stabilization Implies Coprime Factorization. *IEEE Transactions on Automatic Control 34*(4), 435–443.

Sontag, E. D. (1990). Further Facts about Input to State Stabilization. *IEEE Transactions on Automatic Control 35*(4), 473–476.

Sontag, E. D. and Y. Wang (1995). On characterizations of the input-to-state stability property. *Systems and Control Letters 24*, 351–359.

Stoican, F., C. F. Raduinea, and S. Olaru (2011). Adaptation of set theoretic methods to the fault detection of a wind turbine benchmark. In *IFAC World Congress*, Milano, Italy, pp. 8322–8327.

Sugeno, M. and G. T. Kang (1988). Structure Identification of Fuzzy Model. *Fuzzy Sets and Systems 28*, 15–33.

Svärd, C. and M. Nyberg (2011). Automated Design of an FDI-System for the Wind Turbine Benchmark. In *IFAC World Congress*, Milano, Italy, pp. 8307–8315.

Taher, S. A., M. Farshadnia, and M. R. Mozdianfard (2013). Optimal gain scheduling controller design of a pitch-controlled VS-WECS using DE optimization algorithm. *Applied Soft Computing 13*, 2215–2223.

Takagi, T. and M. Sugeno (1985). Fuzzy Identification of Systems and Its Application to Modeling and Control. *IEEE Transactions on Systems, Man, and Cybernetics 15*(1), 116–132.

Tan, C. P. and C. Edwards (2003). Sliding mode observers for robust detection and reconstruction of actuator and sensor faults. *International Journal of Robust and Nonlinear Control 13*, 443–463.

Tanaka, K., T. Ikeda, and H. O. Wang (1998). Fuzzy Regulators and Fuzzy Observers: Relaxed Stability Conditions and LMI-Based Design. *IEEE Transactions on Fuzzy Systems 6*, 250–265.

Tanaka, K. and M. Sano (1994a). A Robust Stabilization Problem of Fuzzy Control Systems and Its Application to Backing up Control of a Truck-Trailer. *IEEE Transactions on Fuzzy Systems 2*(2), 119–134.

Tanaka, K. and M. Sano (1994b). On the Concepts of Regulator and Observer of Fuzzy Control Systems. In *IEEE International Conference on Fuzzy Systems*, Orlando, USA, pp. 767–772.

Tanaka, K. and M. Sugeno (1992). Stability analysis and design of fuzzy control systems. *Fuzzy Sets and Systems 45*(2), 135–156.

Tanaka, K., T. Taniguchi, and H. O. Wang (1998). Fuzzy Control Based on Quadratic Performance Function - A Linear Matrix Inequality Approach. In *IEEE Conference on Decision and Control*, Tampa, USA, pp. 2914–2919.

Tanaka, K. and H. O. Wang (1997). Fuzzy Regulators and Fuzzy Observers: A Linear Matrix-Inequality Approach. In *IEEE Conference on Decision and Control*, San Diego, USA, pp. 1315–1320.

Tanaka, K. and H. O. Wang (2001). *Fuzzy Control Systems Design and Analysis: A Linear Matrix Inequality Approach*. John Wiley & Sons, Inc.

Utkin, V. I. (1977). Variable Structure Systems with Sliding Modes. *IEEE Transactions on Automatic Control 22*(2), 212–222.

Utkin, V. I. (1992). *Sliding Modes in Control Optimization*. Springer-Verlag Berlin Heidelberg.

Van der Hooft, E. L., P. Schaak, and van Engelen, T. G. (2003). Wind Turbine Control Algorithms. Technical report, DOWEC-F1W1-EH-03-094/0, ECN-C-03-111.

Wang, H. O., K. Tanaka, and M. F. Griffin (1995a). An Analytical Framework of Fuzzy Modeling and Control of Nonlinear Systems: Stability and Design Issues. In *American Control Conference*, Seattle, USA, pp. 2272–2276.

Wang, H. O., K. Tanaka, and M. F. Griffin (1995b). Parallel Distributed Compensation of Nonlinear Systems by Takagi-Sugeno Fuzzy Model. In *International Conference on Fuzzy Systems (FUZZ-IEEE/IFES)*, Yokohama, Japan, pp. 531–538.

Wang, H. O., K. Tanaka, and M. F. Griffin (1996). An Approach to Fuzzy Control of Nonlinear Systems: Stability and Design Issues. *IEEE Transactions on Fuzzy Systems 4*(1), 14–23.

Wasynczuk, O., D. T. Man, and J. P. Sullivan (1981). Dynamic Behavior of a Class of Wind Turbine Generators During Random Wind Fluctuations. *IEEE Transactions on Power Apparatus and Systems PAS-100*(6), 2837–2845.

Watanabe, K. and D. M. Himmelblau (1982). Instrument Fault Detection in Systems with Uncertainties. *International Journal of Systems Science 13*(2), 137–158.

Wei, X. and M. Verhaegen (2011). Sensor and actuator fault diagnosis for wind turbine systems by using robust observer and filter. *Wind Energy 14*(4), 491–516.

Yang, X. and J. M. Maciejowski (2012). Fault-tolerant model predictive control of a wind turbine benchmark. In *IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*, Mexico City, Mexico, pp. 337–342.

Zeng, J., D. Lu, Y. Zhao, Z. Zhang, W. Qiao, and X. Gong (2013). Wind Turbine Fault Detection and Isolation Using Support Vector Machine and a Residual-Based Method. In *American Control Conference*, Washington, D.C., USA, pp. 3661–3666.

Zhang, X., Q. Zhang, S. Zhao, R. Ferrari, M. M. Polycarpou, and T. Parisini (2011). Fault Detection and Isolation of the Wind Turbine Benchmark: an Estimation-based Approach. In *IFAC World Congress*, Milano, Italy, pp. 8295–8300.

Zhang, Y. and J. Jiang (2003). Bibliographical Review on Reconfigurable Fault-Tolerant Control Systems. In *IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*, Washingto, D. C., USA, pp. 265–276.