Robust LPV Control for Wind Turbines

A THESIS
SUBMITTED TO THE FACULTY OF THE GRADUATE SCHOOL
OF THE UNIVERSITY OF MINNESOTA
BY

Shu Wang

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

Peter J. Seiler, Advisor

July 2016

Acknowledgments

I would like to thank all the people who supported and helped me through the years. It is a long way with joys and pains to get the PhD degree and I would never have been able to make it alone. First of all, I want to thank my advisor Peter Seiler. I am grateful for the time you spent advising me on research, on scientific writing and presentations, etc. None of my work would have been done without your guidance. It is your advising that shaped my thoughts on how to solve problems rigorously and efficiently as a Doctor of Philosophy.

I would like to thank my prelim and final oral exams' committee members, Zongxuan Sun, Rajesh Rajamani, Andrew Lamperski and Mihailo Jovanovic. I want to express my gratitude for your comments and suggestions on my research and thesis. I would also like to thank my collaborators Harald Pfifer and Daniel Ossmann, for the time we worked together on IQC theories and wind turbine control problems. I also want to thank all my collegues and friends in the society of systems and controls, especially Bin Hu, Yongxin Chen and Sei Zhen Khong. I could never forget the discussions we had on research and all other interesting topics in the past few years.

Finally, I want to thank my parents for supporting and understanding me all these years. For sure I will find a girlfriend sooner or later (I can not stop laughing at this moment). I would also like to acknowledge my gratitude to the Solie's family for hosting me in Minnesota. There were so many good memories with you and all other friends that made me feel the warm of home in a country of different language and culture.

Dedication

To My Family

Abstract

This thesis proposes a uniform multi-input, multi-output (MIMO) control framework for wind turbines using the robust linear parameter varying (LPV) design method. This framework is built on an LPV model of the wind turbine, which has a parametric dependence on the trim wind speed. It takes multiple objectives in different wind conditions into a systematic consideration. Therefore, existing results based on single-input, single-output (SISO) linear control design can be integrated together with stability and performance guarantees. The proposed design has a uniform structure that covers turbine operations in all wind conditions and provides better load reduction performance than the baseline controller. This MIMO control architecture can also be extended for active power control (APC) purposes. Therefore, the wind turbine is capable of providing ancillary services to maintain reliability of power grids.

The control design in this thesis takes a robust LPV approach. Specifically, this thesis proposes a robust synthesis algorithm for LPV systems using the theory on integral quadratic constraints (IQCs). This algorithm is a coordinate-wise descent similar to the well-known DK iteration for μ synthesis. It alternates between an LPV synthesis step and an IQC analysis step. Both steps can be efficiently solved as semidefinite programs. It is shown that the proposed algorithm ensures that the robust performance is non-increasing at each iteration step. Therefore, this algorithm is used in this thesis to synthesize a robust LPV controller for wind turbines to provide APC. Robust performance of this controller has been verified using high fidelity simulations. Applications of this method will be various and not limited to wind turbines.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Wind power is a promising renewable green energy for its zero fuel cost, zero emissions and abundant sources. With advancements in technology, wind power is increasing very fast all over the world. Figure 1.1 shows world installed capacity of wind from 1997 to 2012 [1]. Though it only met $3\,\%$ of the electricity demand globally in 2012, the penetration of wind power is very high in some European countries [4]. In the United States, the amount of wind power is expected to increase to about $20\,\%$ of the electricity supply by 2030 [5,6]. Correspondingly, most states of U.S. have renewable portfolio standards, with wind an important part of it. According to National Renewable Energy Laboratory's (NERL) research [7], estimated U.S. electricity demand in 2050, under some assumptions and constraints, could be met with $80\,\%$ generation from renewable energy (wind, solar and hydroelectric, etc.). In that scenario, wind will be able to provide about 40% of overall generation.
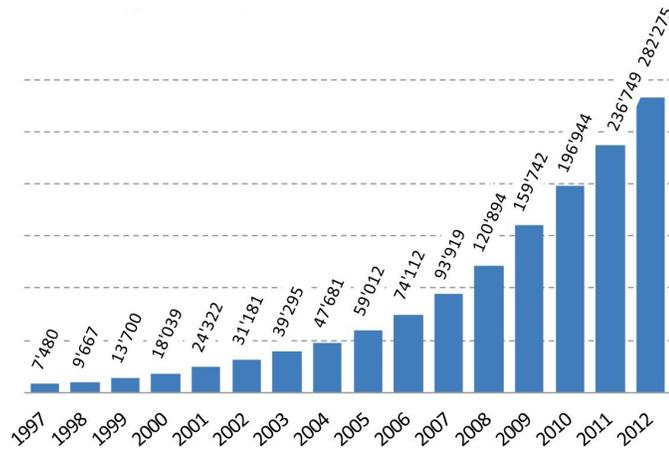
Figure 1.1: World total installed capacity of wind [1].

The rapid growth of wind power requires the wind industry to further minimize costs and maximize profits in energy markets. These objectives bring several challenges to the control system of a single utility scale wind turbine, as it has a significant effect on the turbine performance. Therefore, this thesis is focused on the control design for wind turbines.

Traditionally, the wind turbine control system is designed to maximize the power generation from available energy in the wind and minimize structural loads of turbine components. The trade-off between these two objectives is typically achieved by using a mode-dependent controller which contains two independent control loops to achieve distinct objectives in different wind speed regions. This baseline controller has been widely accepted by the industry. However, as the size of wind turbines grows and the structure becomes more flexible, considerations on load reduction are more critical for larger wind turbines. Therefore, extra control loops were proposed to improve the load reduction performance, such as individual pitch control [8, 9], tower and drive train dampers [10–13]. Though these methods significantly decreased the turbine loads, the control system structure became more complicated. There are also other concerns, such as potential dynamic couplings between different control loops that might affect system performance and stability.

The second challenge for turbine control systems comes from the dynamics variation and uncertainties in the turbine model. Traditional control system design for wind turbines usually takes a linear time invariant (LTI) approach since LTI control theory has been well developed and design tools are simple and reliable for applications [14, 15]. Though some other controllers such as individual pitch control are implemented in the form of linear time varying systems, they are still designed in the framework of LTI systems [9, 16]. However, the performance with an LTI design is difficult to guarantee as the capacity of utility scaled turbines increase. Larger capacity of turbine usually indicates larger size and more flexibility, which lead to larger variation of system dynamics in different wind conditions. At the same time, there are also concerns on robustness in the control design. Though high fidelity simulation environments like FAST [17] provide relatively accurate model of the turbine for control design, the real dynamics of the turbine might be subject to different uncertainties that will significantly degrade the control performance or even lead instability of the system.

Another challenge for turbine control systems comes from the updated requirements of power grids. Traditional turbine control systems attempt to maximize the power generation from available energy in the wind. The power output of wind turbines is therefore variable due to time-varying wind speeds and it may cause unreliable operation of the power grid. This is not a significant issue when wind power is only a small portion of the total electricity generated on the grid. However, to integrate higher levels of variable wind power into the grid it is important for wind turbines to provide ancillary services to help maintain the

reliability of power grids. Ancillary services require wind turbines to provide active power control (APC) [4]. Therefore, it is important to improve the traditional turbine control system for APC purposes.

1.1 Thesis Overview

To address all these issues as mentioned above, this thesis proposes a uniform multi-input, multi-output (MIMO) control framework for wind turbines using the robust linear parameter varying (LPV) design method. The contents in each chapter of the thesis are briefly introduced as follows.

Chapter 2 gives an overview of wind turbine modeling and controls. It covers the basics on the Clipper Liberty C96 2.5 MW wind turbine, which is the research focus of this thesis. The operations of wind turbine in different wind speeds and objectives for its control system are also introduced here. The modeling of wind turbines starts from a brief review of a simple nonlinear one state model that captures the aerodynamics and rotor dynamics of the turbine. To cover the structural dynamics of the turbine, a high fidelity model is also presented. This high fidelity model is integrated in the FAST simulation package [17] and can be linearized for advanced control design purposes. This chapter also briefly reviews the actuators and sensors used for control of wind turbines. In the end, a baseline controller which has been widely accepted in industry is presented.

In Chapter 3, a uniform MIMO control architecture that covers all wind conditions will be first proposed. This LPV controller is able to maximize the power generation in Region 2 and track the rated generator speed in Region 3. Considerations on load reduction will also be parts of the design. The recently developed LPV toolbox in Matlab [18] will be used to synthesize the LPV controller. This uniformly designed controller allows the turbine to meet objectives in different wind conditions and ensures a smooth transition between different wind speed regions. Performance of this LPV controller will be further verified by high fidelity simulations in FAST and post analysis.

In Chapter 4, the theory of integral quadratic constraints (IQCs) [19] will be applied to develop a robust synthesis algorithm for a class of uncertain LPV systems. The uncertain system is described as an interconnection of a nominal (not-uncertain) gridding based LPV system and a block structured uncertainty. The input/output behavior of the uncertainty is described by IQCs. The robust synthesis problem leads to a non-convex optimization and the proposed algorithm is a coordinate-wise descent which is similar to the well-known DK iteration for μ synthesis and can be efficiently solved as semidefinite programs. The effectiveness of the proposed method is demonstrated on a simple numerical example.

In Chapter 5, the robust synthesis algorithm as proposed in Chapter 4 will be used to design an LPV controller to provide APC. The proposed control system architecture can be considered as an extension of the LPV controller in Chapter 3. The design procedure is therefore significantly simplified since some of the tuning results in Chapter 3 can be directly inherited here. In addition, a multiplicative uncertainty is considered in the blade pitch input channel of the turbine model. The synthesized robust LPV controller shows similar performance on APC as a nominal LPV controller designed without considerations of uncertainty. However, the robust controller has much better performance when the worst case uncertainty is added to the system dynamics.

Chapter 6 is the end of this thesis. Brief conclusions will be made to summarize all works and contributions in this thesis and provide possible directions for future works.

1.2 Thesis Contributions

This thesis extends theories on robust control for LPV systems and provides a uniform MIMO control architecture for wind turbines using the robust LPV approach. The contributions of this thesis are listed as follows.

1. **LPV Control Framework:** This framework as proposed in Chapter 3 is built on an LPV model of the wind turbine, which has a parametric dependence on the trim wind speed. It takes multiple objectives in different wind conditions into a systematic consideration, such that existing results based on single-input, single-output (SISO) linear control design can be integrated together with stability and performance guarantees. Detailed simulations and post analysis show that this LPV controller achieves similar performance of power capturing as the baseline controller in below rated wind speed and much better performance of load reduction and generator speed tracking in above rated wind speed. More importantly, the proposed framework is an open structure and can be extended in the future to allow more feedback loops for further load reductions and/or other operations of wind turbines, such as APC.

2. **Robust LPV Synthesis:** The nominal LPV design, as shown Chapter 3 does not consider possible uncertainties in the plant. Therefore, the control performance might be significantly degraded if the system dynamics is perturbed. However, robust synthesis algorithms like μ-synthesis [20] are not directly available for LPV systems and existing theories for robust LPV control are still incomplete. Chapter 4 proposes a robust synthesis algorithm for LPV systems using the theories on IQCs. It is a coordinate-wise descent similar to the well-known DK iteration for μ synthesis. Specifically, the proposed algorithm alternates between an LPV synthesis step and an IQC

analysis step. Both steps can be efficiently solved as semidefinite programs. It is shown that the proposed algorithm ensures that the robust performance is non-increasing at each iteration step. The effectiveness of the proposed method is demonstrated on a simple numerical example. Applications of this method are various and not limited to wind turbines.

3. **Robust LPV Design for APC:** Chapter 5 proposes an LPV controller to provide APC using the robust LPV synthesis method presented in Chapter 4. This LPV controller is developed from the design in Chapter 3 by adding a feedback loop for the power reference tracking. To ensure robustness of the controller, a multiplicative uncertainty in the blade pitch input channel is considered in the turbine model. Simulation results show that the robust LPV controller provides fast responses to the power reference commands and maintains performance even with existence of the worst case uncertainty. Therefore, it is expected that wind turbines will be able to participate in ancillary services of power grids by providing APC. This will greatly improve the competitiveness of wind in the energy market.

Chapter 2

Wind Turbine Modeling and Control

2.1 Overview of Turbine Operation and Control

The history of extracting energy from wind dates back to the first windmills about 3000 years ago [8]. Windmills were used to take energy from wind for mechanical operations, such as pumping water or grinding grains. As electricity became a major source of power for modern industry, wind turbines were built by James Blyth in 1887 [21] and Charles Brush in 1888 [22] respectively. These turbines consisted of a windmill and a generator to convert the wind power into electricity. Another milestone in the history of wind power development was the Smith Putnam wind turbine built in 1941 [23]. As the first wind turbine in history with a capacity of 1.25 MW, it showed a potential for large scale turbines. Many different turbine designs have been considered over the past 100 years [8, 24]. The capacity of a single turbine has also increased from 12 KW to 10 MW [25]. In this thesis, the focus is on 3-blades, variable speed, horizontal axis, on shore, upwind turbine. This is the most popular one in wind industry. Commercialized utility scale turbines with this design have been installed with capacity up to 7.5 MW (E-126 wind turbine designed by Enercon in Germany [25], which has a hub height of 135 m and a rotor diameter of 126 m).

The specific model that will be studied in this thesis is the Clipper Liberty C96 2.5 MW wind turbine which is located in UMore Park, Rosemount, MN, as shown in Figure 2.1. This turbine is owned by EOLOS Wind Energy Research Consortium at University of Minnesota [2] for research purposes. The author of this thesis therefore has access to the high fidelity model, the baseline controller and real time operational data of this turbine. Basics of the C96 turbine are shown in Table 2.1.

Figure 2.1: Clipper Liberty C96 2.5 MW wind turbine in University of Minnesota [2].

Table 2.1: Basics of the C96 2.5 MW wind turbine.

Hub height	80.4 m
Rotor radius	48 m
Wind speed range for operation	$4 \sim 25$ m/s
Rated rotor speed	15.49 RPM
Rated generator torque	23473 N · m
Rated power	2.5 MW

Figure 2.2 shows the common structure and components of the above described type of wind turbine [3]. As the wind flow passes through the rotor plane of the turbine, lift is generated on blades and leads to the rotation of the low speed rotor shaft. The aerodynamic torque generated on the low speed rotor shaft transmits to the high speed generator shaft through the gearbox and the generator converts mechanical energy of the rotation into electrical energy. The low speed shaft, gearbox, high speed shaft and generator are all installed inside the nacelle, which is mounted on top of the turbine tower.

The control of wind turbines relies on some actuators. As shown in Figure 2.2, the yaw motor is used in the upwind turbine to rotate the nacelle such that the rotor plane faces the direction of the incoming wind flow for maximizing the power capturing. Other actuators that are commonly used for turbine control are the blade pitch and generator torque. The servo motor installed at the root of each blade is used to pitch the blade and change the lift force generated on the blade. The generator torque is used to change the power output and generator speed. Details on turbine actuators will be provided in Section 2.4.1.

Sensors are necessary for the turbine control and monitoring. For example, sensors on the drive train (which includes the low speed rotor shaft, high speed generator shaft and gearbox) are used to measure the rotor speed and/or generator speed for the closed loop

Figure 2.2: Wind turbine structure and components [3].

feedback control of wind turbines. Another sensor that is widely used on wind turbines is the anemometer (as shown in Figure 2.2), which usually sits on the top of the nacelle. However, the wind speed measurement from the anemometer is corrupted by the rotation of blades and it is therefore limited for control purposes. Modern wind turbines also have some other sensors for monitoring states and loads of the turbine and improving the performance of turbine control systems, which will be further discussed in Section 2.4.2.

The captured power of a wind turbine is given by

$$P_c = \frac{1}{2}C_p(\beta, \lambda)\rho A_r v^3 \tag{2.1}$$

where ρ is the air density [kg/m^3], $A_r := \pi R^2$ is the swept area of rotor blades perpendicular to the wind flow [m^2], R is the radius of the rotor area [m], and v is the wind speed [m/s] [8,26]. The non-dimensional power coefficient C_p is the fraction of the available wind power captured by the turbine. The power coefficient is a function of the (collective) blade pitch angle β [deg] and the tip speed ratio λ [unitless]. The tip speed ratio is defined as the ratio $\lambda := \frac{R\omega}{v}$ where ω is the rotor speed [rad/s]. In words, λ is defined as the blade tip tangential velocity divided by the wind speed. Figure 2.3 shows the contour of C_p as a function of β and λ (contours with values less than or equal to 0 have been removed from the plot). It is generated for the above mentioned C96 2.5 MW wind turbine and C_p achieves its maximal value at $\beta_* = 1.6 \deg$ and $\lambda_* = 8.4$. This maximum value C_{p*} might

be varying from one turbine to another. However, there is a theoretical upper bound for C_{p*} that is equal to $16/27$. This upper bound is known as Betz limit after the German aerodynamicist Albert Betz [8]. Here, the C_{p*} for the C96 turbine is also below the Betz limit but the exact value is not provided for proprietary reasons.
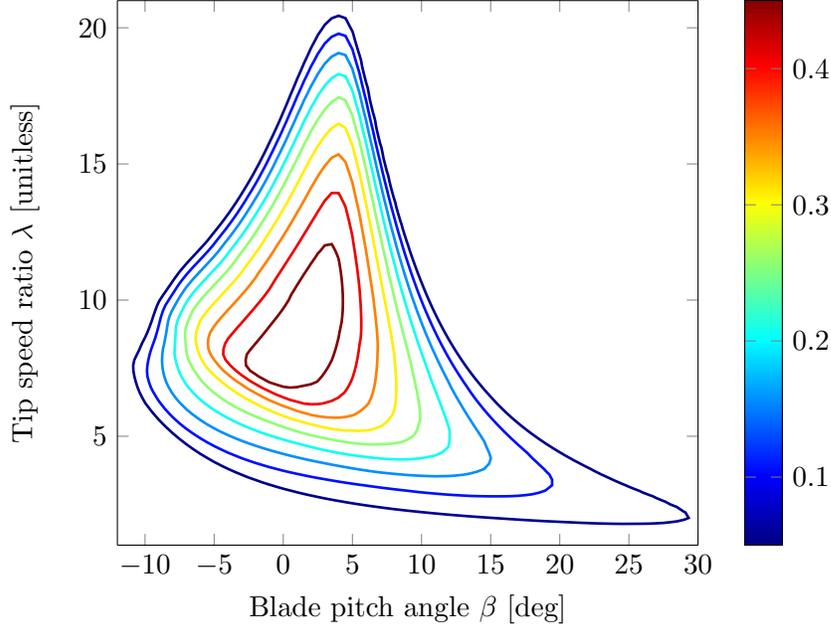
Figure 2.3: The contour of C_p for the C96 2.5 MW wind turbine.

The objectives of the wind turbine control are to maximize the power generation from available energy in the wind and minimize structural loads of turbine components. The trade-off between these two objectives is typically achieved using a mode-dependent controller with distinct objectives in different wind speed regions [8, 27, 28]. As shown in Figure 2.4, there are essentially four operating regions for the turbine in the power versus wind speed curve. Below the cut-in speed (Region 1), the turbine is kept in a parked, non-rotating state as there is insufficient energy available from the wind. Above the cut-out speed (Region 4), the turbine is shut down to prevent structural damages. Between the cut-in and rated wind speeds (Region 2), the main objective is to maximize the captured power. Between the rated and cut-out wind speeds (Region 3), the main objective is to maintain the rated power while minimizing structural loads on the turbine. The transition between Regions 2 and 3 is referred to as Region 2.5. Region 2.5 is introduced because the rated rotor speed is usually reached before the Region 2 control law reaches the rated torque.

The design of wind turbine controllers requires dynamic models of the wind turbine plant, actuators and sensors. Details on the modeling and baseline controller design are provided in following sections. In Section 2.2, a low fidelity, one-state, nonlinear turbine model will be
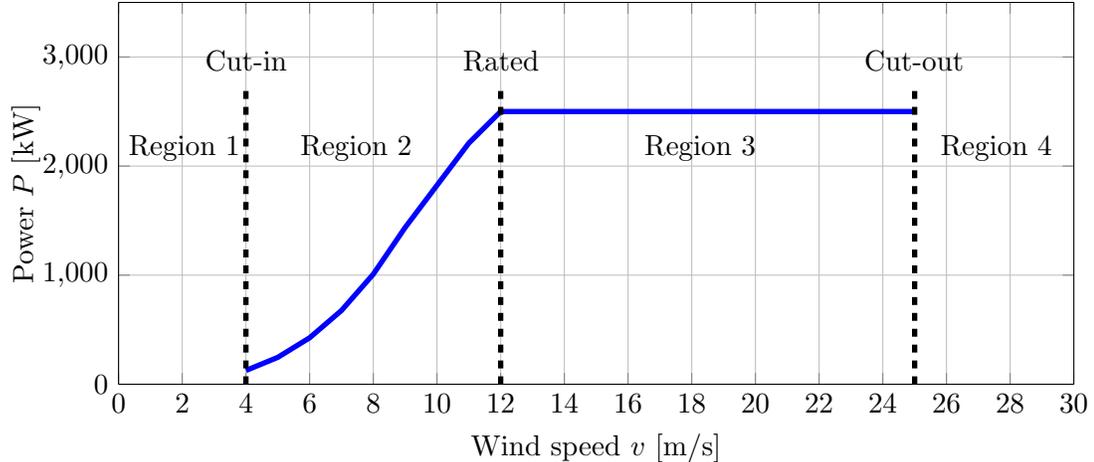
Figure 2.4: Operation regions for the C96 2.5 MW wind turbine.

introduced that captures the rotor dynamics. In Section 2.3, a high fidelity nonlinear wind turbine model from FAST [17] will be explained. This high fidelity model includes more detailed structural dynamics of the turbine and it is therefore suitable for advanced control design and simulation purposes. Linearization of the FAST model and post-processing of the linearized model will also be explained in this section. Section 2.4 will discuss actuator and sensor dynamics. In Section 2.5, a baseline controller for wind turbines will be presented.

2.2 Low Fidelity Model

This section provides a one-state, nonlinear model of a wind turbine that captures the steady state aerodynamics and the rigid body rotor dynamics [29]. This low fidelity model does not contain structural dynamics of the turbine, such as vibrations of tower and blades. Therefore, it is usually used when loads on the tower and blades are not considered. However, this simplified model is helpful for understanding basic principles of the turbine operation.

Figure 2.5 shows a diagram of this one-state model. The lift force generated on blades leads to the aerodynamic torque τ_a on the low speed rotor shaft. It is balanced by the generator torque τ_g on the other side of the drive train. Considering all rotating parts of the turbine as a rigid body, which includes the blades, hub and drive train, the rotor dynamics can be expressed as comprehensive effects of τ_a and τ_g:

$$J\dot{\omega} = \tau_a - N\tau_g \tag{2.2}$$

where τ_a and τ_g [N $\cdot$ m] are aerodynamic and generator torques on the drive train. J [kg $\cdot$ m^2] is the inertia of all rotating parts, including the blades, hub and drive train. ω [rad/s] is the

10

speed of the rotor shaft. N [unitless] is the gearbox ratio.
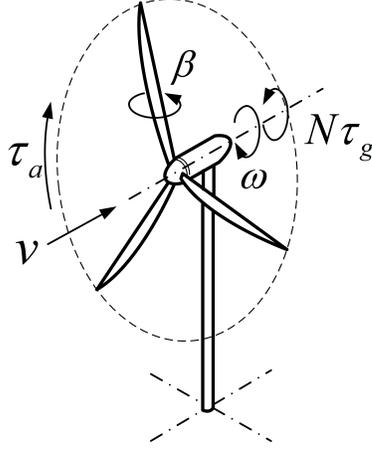
Figure 2.5: One state model of wind turbine.

The aerodynamic torque can be expressed in terms of the captured power and rotor speed as

$$\tau_a = \frac{P_c}{\omega} = \frac{C_p(\beta, \lambda)\rho\pi R^2 v^3}{2\omega} \tag{2.3}$$

Plugging Equation 2.3 into Equation 2.2 leads to the one-state model given by:

$$\dot{\omega} = \frac{C_p(\beta, \lambda)\rho\pi R^2 v^3}{2J\omega} - \frac{N}{J}\tau_g \tag{2.4}$$

In this one-state model, rotor speed ω is the state, wind speed v is the input, collective blade pitch angle β and generator torque τ_g are two control inputs to the system. As shown in Section 2.5, this model can be used to design a baseline controller for the turbine.

2.3 High Fidelity Model

2.3.1 Model Description

In this thesis, the FAST wind turbine simulation package [17] will be used to model turbine dynamics and interact with Matlab/Simulink for turbine control system simulations. FAST, which stands for Fatigue, Aerodynamics, Structures and Turbulence, is a publicly available nonlinear aeroelastic turbine simulation tool developed by the National Renewable Energy Laboratory (NREL). It captures structural dynamics of the turbine drive train, tower and blades that were ignored by the one-state nonlinear rotor model in Section 2.2. FAST has been evaluated against Germanischer Lloyd turbine simulation codes [30]. It has been certified by Germanischer Lloyd that it is suitable for turbine manufacturers to use FAST

simulations for wind turbine performance certifications.

In FAST, the drive train, tower and blades are treated as flexible structures. The drive train torsional flexibility is modeled as a linear inertia-spring-damper system. Deformations of the tower and blades are approximated with the assumed modes method [31, 32]. To apply this method, the tower and blades are treated as cantilever beams with properties varying along their length. These properties are specified at desired points on the tower and blades. Linear interpolation is used between these points. For example, airfoil properties and the mass distribution of the blade can be specified along the blade length. Deformations of the tower and blades are approximated by superposition of basis functions known as mode shapes. Each mode shape corresponds to a particular deformation shape and is defined as one degree of freedom. Though there are couplings between all structure modes, it is assumed in this method that the effect of couping is small and does not affect the model response. Therefore, each mode shape of the tower and blades can be calculated independently based on properties of structures.

The FAST simulation package can model dynamics of 3-blades onshore wind turbines with up to 18 degrees of freedom (DOFs). These DOFs are described as follows. There are 4 DOFs for first and second tower bending modes in fore-aft and side-to-side directions. For each blade, there are 2 DOFs for first and second blade flapwise bending modes and 1 DOF for the first blade edgewise bending mode. For the drive train, there are 2 DOFs for the torsion and generator speed. There is 1 DOF for the nacelle yaw motion. 2 more DOFs account for the rotor and tail furl. There are extra 6 DOFs available for modeling platform motions of offshore wind turbines. That is up to 24 DOFs.

To simulate the operation of wind turbines and calculate deformations of turbine structures in FAST, information of the incoming wind flow in front of the turbine rotor and calculation of aerodynamic loads generated on turbine models are necessary. An accurate model of the incoming wind flow can be solved by computational fluid dynamics (CFD) codes. However, due to time and computational costs of CFD codes, it is much simpler to use TurbSim [33] to generate wind flow profiles. TurbSim is a stochastic, turbulent wind simulator developed by NREL. It is able to generate wind trajectories with various spatial and temporal correlation models. Some common turbulence models that are defined in IEC-61400-1 standards [34] for wind turbine testing are also included. Inputs of TurbSim are various turbulence properties. Its output is a description of the turbulent wind field as a function of time on the turbine rotor. There are two options of fidelity for wind field descriptions. In the first option which is called the hub-height wind profile, it is assumed that turbulence properties are the same all over the rotor by averaging the wind data on the hub height. The wind profile is described by 7 parameters as functions of time. This option is suitable for the linearization of turbine models and control oriented analysis and design. In the second option which is called the

full-field wind profile, the turbulence has spatial variations over the rotor. Therefore the wind profile is described by detailed wind speed data in three directions as function of time on a raster grid over the rotor. It provides a more realistic modeling of the wind field by capturing spatial variations of the wind field. It is therefore suitable for simulation purposes. These pre-calculated wind trajectories can be used by FAST for calculations of aerodynamic loads. This is achieved in FAST by integrating with the AeroDyn code [35], which is also developed by NREL. AeroDyn uses blade element momentum theory [36] to calculate aerodynamic forces and moments. FAST, AeroDyn and TurbSim codes are of sufficient fidelity for control design and simulation purposes. Therefore, they are adopted in this thesis.

The dynamics of wind turbines are essentially nonlinear. This nonlinear wind turbine model in FAST can be expressed as

$$\ddot{q} = f(\dot{q}, q, d, u, t)$$
$$y = g(\dot{q}, q, d, u, t)$$

(2.5)

where $q \in \mathbb{R}^{n_q}$ and $\dot{q} \in \mathbb{R}^{n_q}$ are the turbine states. As described above, there are up to 24 DOFs available in FAST. Therefore the maximum value for n_q is 24. For simplicity, however, it is common to disable some DOFs in FAST to obtain a lower order nonlinear or linear model for simulations or control design. $u \in \mathbb{R}^5$ is the control input vector which includes the generator torque, individual pitch angles for three blades and yaw angle. $y \in \mathbb{R}^{n_y}$ is the measurement vector and its dimension depends on chosen outputs. A full list of available outputs in FAST can be found in [17]. $d \in \mathbb{R}^7$ is the wind disturbance input which consists of the hub-height average wind speed, horizontal wind direction, vertical wind speed, horizontal wind shear, vertical power law wind shear, linear vertical wind shear and horizontal hub-height wind gust. This is the simplified wind field description in FAST. The 7 disturbance inputs in this simplified description correspond to the 7 parameters in the hub height wind profile generated by TurbSim. In contrast, the more complex full-field wind description in TurbSim is not suitable for control oriented linearizations in FAST, as there are much more inputs required to match parameters of the full-field wind profile.

2.3.2 Model Linearization

To apply well established linear control techniques, the FAST simulation package provides an option to generate linear system approximations of nonlinear wind turbine dynamics. The linearization in FAST is started by first simulating the nonlinear turbine model under steady wind conditions until it reaches a trim operating trajectory $\bar{q}(t)$. Therefore, the wind disturbance input and control inputs are held constant at trim values $\bar{d}$ and $\bar{u}$ specified by users in the simulation. The resulting trim trajectory $\bar{q}(t)$ is periodic with the trim rotor

rotation period T, i.e. $\bar{q}(t) = \bar{q}(t + T)$, and satisfies

$$
\begin{aligned}
\ddot{\bar{q}} &= f(\dot{\bar{q}}, \bar{q}, \bar{d}, \bar{u}, t) \\
\bar{y} &= g(\dot{\bar{q}}, \bar{q}, \bar{d}, \bar{u}, t)
\end{aligned}
\tag{2.6}
$$

The nonlinear model in Equation 2.5 is linearized around $\bar{q}(t)$ through numerical perturbations [17] and a periodic linear time varying (PLTV) model is generated with state space equations as

$$
\begin{aligned}
\dot{\delta}_x &= A(\bar{q}(t))\delta_x + B_d(\bar{q}(t))\delta_d + B_u(\bar{q}(t))\delta_u \\
\delta_y &= C(\bar{q}(t))\delta_x + D_d(\bar{q}(t))\delta_d + D_u(\bar{q}(t))\delta_u
\end{aligned}
\tag{2.7}
$$

Where

$$
\begin{aligned}
\delta_x(t) &:= \begin{bmatrix} \delta_q(t) \\ \dot{\delta}_q(t) \end{bmatrix} = \begin{bmatrix} q(t) - \bar{q}(t) \\ \dot{q}(t) - \dot{\bar{q}}(t) \end{bmatrix} \\
\delta_d(t) &:= d(t) - \bar{d} \\
\delta_u(t) &:= u(t) - \bar{u} \\
\delta_y(t) &:= y(t) - \bar{y}(t)
\end{aligned}
\tag{2.8}
$$

The dimensions of $\delta_x(t)$, $\delta_d(t)$, $\delta_u(t)$ and $\delta_y(t)$ directly follow from the state, input and output signal dimensions in Equation 2.6. Since the trim trajectory $\bar{q}(t)$ is periodic with the rotor rotation period T, the state space equations of the model in Equation 2.7 are also periodic with the same period.

As control techniques for linear time invariant (LTI) systems are mature and common in theory and practice, it is desirable to approximate the PLTV model in Equation 2.7 by an LTI one. There are several approaches available to generate the approximated LTI model. The simplest methods are to evaluate the PLTV model at a fixed rotor position or to average state space matrices of the PLTV model over one rotor period. However, these methods ignore periodic properties of the system and typically do not provide an approximated LTI model with sufficient accuracy. Another approach is developed based on Floquet theory [37, 38]. It converts a PLTV system into a system with a constant state "A" matrix by using a time varying coordinate transformation. The Floquet transformation retains periodic properties of the system but loses the physical intuition on system states. The most common approach in the wind turbine industry is to use the multi-blade coordinate (MBC) transformation [39–41] to generate a weakly PLTV system from the original one. An LTI model is therefore approximated by averaging the weakly PLTV system without losing too many periodic properties. This approach is adopted in this thesis for model based control design of wind turbines. Details on the MBC transformation will be given in Section 2.3.3.

2.3.3 Multi-Blades Coordinate Transformation

The nonlinear and linear wind turbine models (Equation 2.5 and 2.7) presented in Sections 2.3.1 and 2.3.2 are defined in various coordinate systems. Details on these coordinate systems can be found in [17]. Specifically, the tower, drive train and generator DOFs are defined in an earth fixed coordinate system while DOFs associated with turbine blades are defined in a system that rotates with the rotor. As mentioned in Section 2.3.2, the LTI model approximated by averaging the PLTV model defined in the mixed coordinate system (both rotating and non-rotating) is not sufficiently accurate as the averaging ignores periodic properties of the model. For instance, dynamics of tower motion coupled by blade modes will be removed in this approach. Therefore, it is desirable to use the MBC transformation to transform states, inputs and outputs of the model from the mixed coordinate system to a purely non-rotating coordinate system.

Ideally, the MBC transformation converts the PLTV system into an LTI system. In practice, however, the transformed system is still "weakly" periodic, i.e. it is periodic but with significantly less time variation compared to the original PLTV system. An LTI system can be approximated by averaging state space matrices of the weakly PLTV system over one rotor period. This approximation LTI model is of sufficient accuracy for control oriented design purposes.

The MBC transformation is defined by a transformation matrix $M : \mathbb{R} \to \mathbb{R}^{3 \times 3}$ as a function of the rotor position ψ:

$$M(\psi) = \begin{bmatrix} 1 & \sin(\psi) & \cos(\psi) \\ 1 & \sin(\psi + \frac{2\pi}{3}) & \cos(\psi + \frac{2\pi}{3}) \\ 1 & \sin(\psi + \frac{4\pi}{3}) & \cos(\psi + \frac{4\pi}{3}) \end{bmatrix} \tag{2.9}$$

For a given rotor position ψ, $M(\psi)$ transforms variables associated with 3 blades from the non-rotating frame to the rotating frame. Conversely, the inverse of $M(\psi)$ transforms variables from the rotating frame to the non-rotating frame:

$$M^{-1}(\psi) = \frac{2}{3} \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \\ \sin(\psi) & \sin(\psi + \frac{2\pi}{3}) & \sin(\psi + \frac{4\pi}{3}) \\ \cos(\psi) & \cos(\psi + \frac{2\pi}{3}) & \cos(\psi + \frac{4\pi}{3}) \end{bmatrix} \tag{2.10}$$

The MBC transformation has a very straight forward physical interpretation in the application of wind turbines. As shown in Figure 2.6, consider a force F_i acting on the tip of the i-th blade in the direction that is perpendicular to the rotor plane. Assuming that the rotor position ψ is defined as the azimuth angle of the 1-st blade, $\psi + (i - 1)\frac{2\pi}{3}$ is therefore the azimuth angle for the i-th blade. Applying the inverse MBC transformation to the force

$F = \begin{bmatrix} F_1 & F_2 & F_3 \end{bmatrix}^T$ results in the force in the non-rotating frame as $F^{nr} = \begin{bmatrix} F_{avg}^{nr} & F_{yaw}^{nr} & F_{tilt}^{nr} \end{bmatrix}^T$. The superscript nr denotes variables defined in the non-rotating frame. After the transformation, these variables have meanings in terms of rotor motion instead of individual blades. F_{avg}^{nr} is the averaged force of F that causes the rotor to bend as a cone. F_{yaw}^{nr} and F_{tilt}^{nr} are the forces resulting in rotor yaw and tilt, respectively [42].
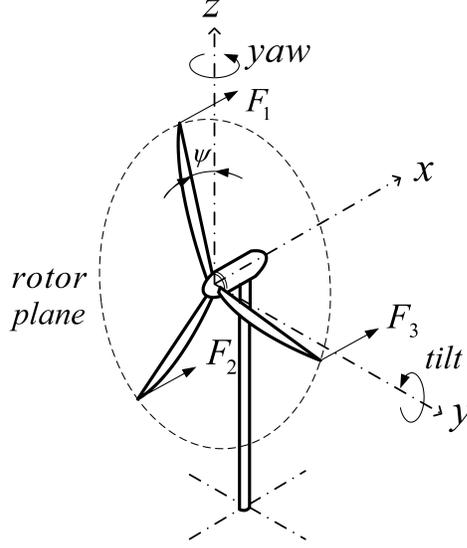
Figure 2.6: Effects of blade tip forces in the non-rotating frame.

The MBC transformation converts the original PLTV model into a "weakly" PLTV model. This is due to the fact that periodic dynamics of blades usually have a phase shift of $\frac{2\pi}{3}$ between each blade. Specifically, assume the force F_i mentioned above is sinusoid $F_i = F_0 \sin(\psi + (i-1)\frac{2\pi}{3} + \theta)$ while θ is an initial phase shift. As the azimuth angle for the i-th blade is $\psi + (i-1)\frac{2\pi}{3}$, F_i has the same value for the i-th blade at the same azimuth angle. Performing the MBC transformation on the force

$$F = F_0 \begin{bmatrix} \sin(\psi + \theta) \\ \sin(\psi + \frac{2\pi}{3} + \theta) \\ \sin(\psi + \frac{4\pi}{3} + \theta) \end{bmatrix} \tag{2.11}$$

results in the force in the non-rotating frame as $F^{nr} = F_0 \begin{bmatrix} 0 & \cos(\theta) & \sin(\theta) \end{bmatrix}^T$ which is constant instead of periodic. In a more general setting, assume q_i is the state of the i-th blade that has the same period as the rotor motion and Fourier series of q_i can be expressed as

$$q_i = \sum_{n=0}^{\infty} a_n \sin(n(\psi + (i-1)\frac{2\pi}{3}) + \theta_n) \tag{2.12}$$

After the MBC transformation, the n-th order harmonic term (which is usually called the

16

np term while p stands for the periodic motion of the rotor)

$$q^{(n)} = a_n \begin{bmatrix} \sin(n\psi + \theta_n) \\ \sin(n(\psi + \frac{2\pi}{3}) + \theta_n) \\ \sin(n(\psi + \frac{4\pi}{3}) + \theta_n) \end{bmatrix} \tag{2.13}$$

will be converted in the non-rotating frame as $q^{nr(n)} = \begin{bmatrix} q_{avg}^{nr(n)} & q_{yaw}^{nr(n)} & q_{tilt}^{nr(n)} \end{bmatrix}^T$. The detailed value of each entry in $q^{nr(n)}$ depends on the order n and can be generalized as shown in Table 2.2.

Table 2.2: Mapping of np terms from the rotating frame by MBC.

n	$q_{avg}^{nr(n)}$	$q_{yaw}^{nr(n)}$	$q_{tilt}^{nr(n)}$
0	$a_0 \sin\theta_0$	0	0
1	0	$a_1 \cos\theta_1$	$a_1 \sin\theta_1$
2	0	$-a_2 \cos(3\psi + \theta_2)$	$a_2 \sin(3\psi + \theta_2)$
...	...	...	...
$3k + 0$	$a_n \sin(3k\psi + \theta_n)$	0	0
$3k + 1$	0	$a_n \cos(3k\psi + \theta_n)$	$a_n \sin(3k\psi + \theta_n)$
$3k + 2$	0	$-a_n \cos\left(3(k+1)\psi + \theta_n\right)$	$a_n \sin\left(3(k+1)\psi + \theta_n\right)$

It can be concluded from Table 2.2 that $0p$ and $1p$ terms which are usually dominating parts in q_i will be mapped to constant values in the non-rotating frame. Higher order harmonic terms in the rotating frame will be mapped to harmonic terms whose orders are multiples of $3p$ in the non-rotating frame and result in the "weakly" periodic property due to insignificant weightings $\{a_n\}_{n=2}^{\infty}$ on these terms.

The complete MBC transformation for PLTV systems contains a collection of state, input, and output transformations which can be derived using results in [39] and the manual for NREL Matlab utilities that implement the MBC transformation [41]. Applying these transformations to the PLTV model in Equation 2.7 leads to a weakly PLTV model with significantly less periodic variation in state space matrices. Averaging the weakly PLTV system over one rotor period gives an LTI model with sufficient accuracy [40].

The application of the MBC transformation will be shown in the following example. In this simplified example, consider the C96 wind turbine model with 10 DOFs that include rotor position, drive train torsion, first tower fore-aft and side-to-side bending modes, and first flapwise and edgewise bending modes for each blade. The turbine is simulated at a trim point as shown in Table 2.3.

For simplicity, trim values for wind disturbance inputs d and yaw angle u_5 as defined in 2.3.1 are set to 0 except for the hub-height average wind speed $\bar{d}_1 = 16\,\text{m/s}$. This is a typical operation condition in Region 3 as the wind speed is above the rated and the turbine is

Table 2.3: Trim conditions of the C96 2.5 MW wind turbine for linearization.

Trim variables	Values
Hub-height average wind speed	16 m/s
Rotor speed	15.49 RPM
Generator torque	23473 N · m
Blade pitch angles	14.36 deg
Yaw angle	0 deg

operating at the rated rotor speed and generator torque.

Here, the interest is focused on open loop dynamics from the disturbance input of hub-height averaged wind speed δ_{d_1} to the output of tower side-to-side bending moment δ_y [kN · m]. Therefore, all 5 control inputs of δ_u and the remaining 6 wind disturbance inputs of δ_d are disabled. A linearization of the nonlinear FAST model with the interested input and output yields a SISO (single input single output) PLTV model G_ψ as expressed by state space matrices $\begin{bmatrix} A(\psi) & B(\psi) \\ C(\psi) & D(\psi) \end{bmatrix}$. Applying the MBC transformation to G_ψ results in the PLTV model G_ψ^{nr} with all states in the non-rotating coordinate frame. Define the state space matrices for G_ψ^{nr} as $\begin{bmatrix} A^{nr}(\psi) & B^{nr}(\psi) \\ C^{nr}(\psi) & D^{nr}(\psi) \end{bmatrix}$. Figure 2.7 shows entry values in the 1-st column of $A(\psi)$ and $A^{nr}(\psi)$ as functions of ψ in two subplots respectively. It should be noted that there are 20 entries for each column and some entries overlap with each other in the subplots as their values are very close. It is seen that some entries in the first column of $A(\psi)$ have significant variations within one period of ψ while all entries in the 1-st column of $A^{nr}(\psi)$ have much smaller variations with ψ. Checking entries in other columns or rows of state space matrices leads to similar comparison results. It is therefore valid to conclude that G_ψ^{nr} is a "weakly" PLTV system.

In the next step, state space matrices of G_ψ and G_ψ^{nr} are averaged over one rotor period to generate LTI models $G(s)$ and $G^{nr}(s)$ respectively. Bode magnitude plots of $G(s)$ and $G^{nr}(s)$ are shown in Figure 2.8. As a reference, the grey plot is generated by identifying the C96 model with the same 10 DOFs enabled in FAST, denoted as $G^{id}(s)$. This identification is started by first simulating the model in trim conditions as specified in Table 2.3. The simulation lasts for 400 s and the output is recorded as $\bar{y}$. In the second simulation, a 400 s chirp signal with small magnitude (± 0.5 m/s) is added to the channel of hub-height wind speed as disturbance δ_{d_1}. The frequency of the chirp signal linearly varies from 0.1 rad/s to 100 rad/s and therefore can be expressed as $\omega(t) = 0.1 + \frac{100-0.1}{400}t$. Subtracting $\bar{y}$ from the output y in the second simulation gives δ_y. By taking Fourier transforms of δ_{d_1} and δ_y, the dynamic response of the model in the frequency domain can be identified as $G^{id}(j\omega) = \frac{\Delta_y(j\omega)}{\Delta_{d_1}(j\omega)}$. Figure 2.8 shows that $G^{nr}(s)$ has a much closer frequency response to the identified model $G^{id}(s)$. Especially, $G^{nr}(s)$ and $G^{id}(s)$ have similar peak magnitudes at 2rad/s, which corresponds to the 1-st tower side-to-side bending mode. In addition, $G^{nr}(s)$ and $G^{id}(s)$
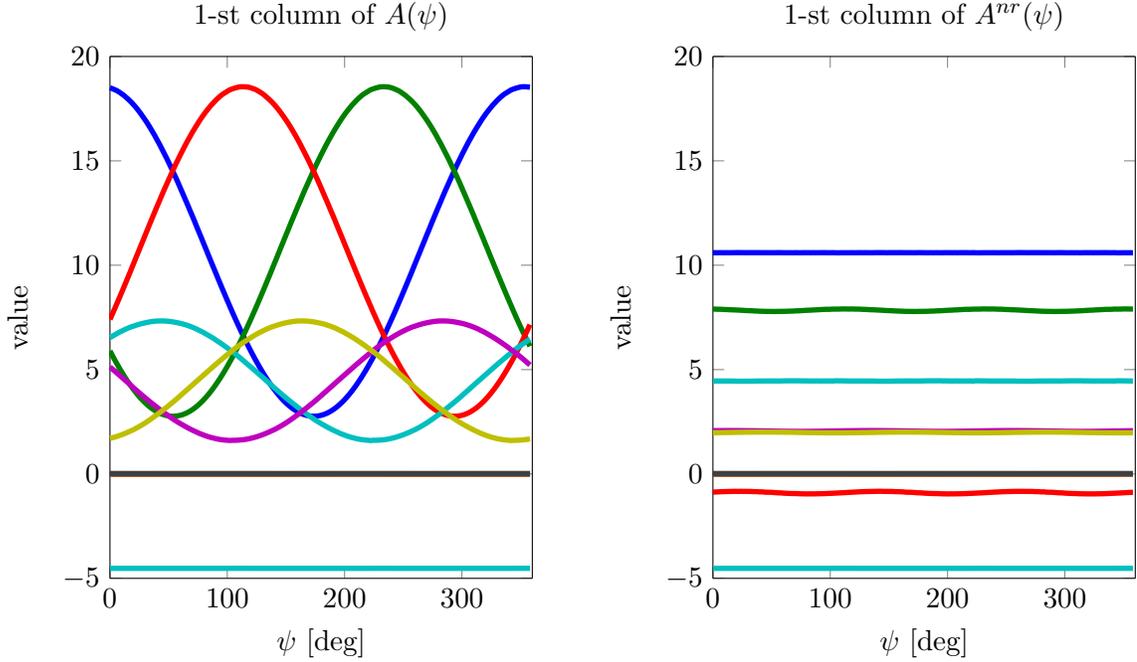
Figure 2.7: Entry values in the 1-st column of $A(\psi)$ and $A^{nr}(\psi)$.

both have 2 more peaks at $4.93\,\mathrm{rad/s}$ and $8.24\,\mathrm{rad/s}$. These 2 peaks correspond to 2 pairs of conjugate poles $-0.05 \pm 4.93\,\mathrm{j}$ and $-0.05 \pm 8.24\,\mathrm{j}$ introduced by blade edgewise bending modes in the non-rotating frame. However, this coupling to tower structural dynamics is lost in $G(s)$ by directly averaging state space matrices of G_ψ. Comparisons in Figure 2.7 and 2.8 indicate that the MBC transformation is a necessary step to get an LTI model with sufficient accuracy for control design purposes.

2.4 Actuators and Sensors

2.4.1 Actuators

As described in Sections 2.1 and 2.3, there are mainly 5 control inputs available for the control of wind turbines: the generator torque, 3 blade pitch angles and turbine yaw angle. However, the use of yaw motor is limited to a very low rate (usually less than $1\,\mathrm{deg/s}$) for avoiding dangerous gyroscopic forces [43]. Due to the slow dynamics of yaw motion, research on yaw control is not of great interest. In this thesis, the yaw actuator model will not be considered in the design and the yaw angle is held constant to the direction of incoming wind flow in simulations.

The dynamics of generator torque actuation is also ignored in this thesis. This is because
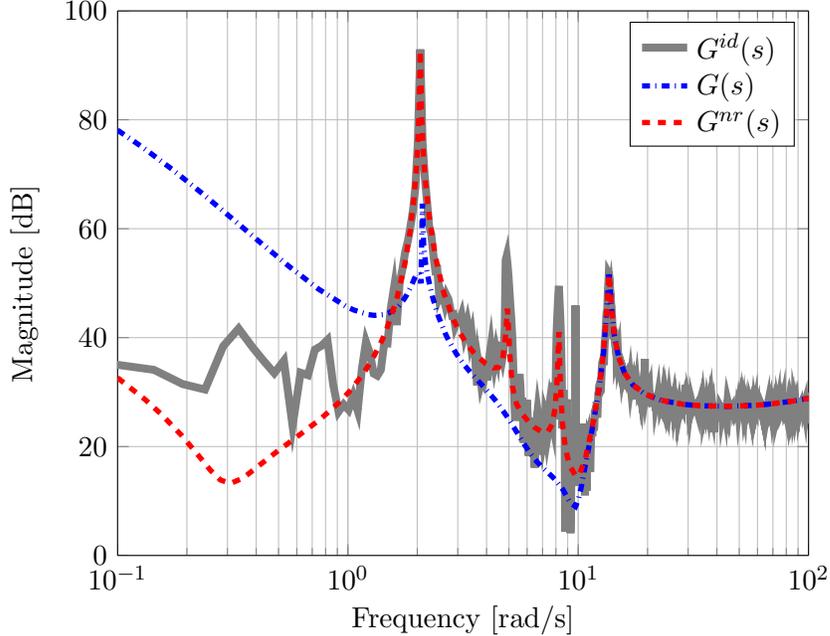
Figure 2.8: Bode magnitude plots of the identified model $G^{id}(s)$, averaged LTI models $G(s)$ (before MBC) and $G^{nr}(s)$ (after MBC).

the power electronics on modern utility scale wind turbines has very small time constant and the dynamics of generator torque actuation is much faster than the turbine dynamics of interest [43]. It is reasonable to assume that the generator torque command can be responded almost immediately for the desired bandwidth. Generator torque is therefore an effective control input for maximizing the captured power [29] and minimizing the load of torsion in the drive train [11].

Blade pitch actuators have restrictive bandwidths for wind turbine control systems. The dynamics of pitch actuators can be modeled as first or second order LTI systems [24]. In addition, there are usually hard bounds on pitch actuation rates. Therefore, these constraints are considered in cases that the control bandwidth is close to the actuation limit or the wind turbulence level is high. It should also be noted that the three blade pitch angles can be controlled collectively or individually, as there is one actuator for each blade. In this thesis, the collective pitch control is considered.

2.4.2 Sensors

Traditionally, there are two types of sensors that have been used on utility scale wind turbines. As introduced in Section 2.1, the sensors for rotor speed and/or generator speed measurements are widely used for the closed loop control of wind turbines. It will be seen

in Section 2.5 that the rotor speed measurement is the most important feedback signal for the baseline controller.

The anemometer is another sensor that wind turbines usually have for wind speed measurements. It is commonly used for supervisory control of wind turbines, e.g. to determine if the wind speed is sufficient to start the turbine [43]. However, due to the interaction between the rotor and the wind, anemometers usually can not provide accurate wind speed measurements. They are therefore limited for applications on the closed loop feedback control.

Instead, advanced wind speed measurement technology, such as LIDAR, [44, 45] has been investigated by researchers for improving performance of turbine control systems. Here, LIDAR stands for light detection and ranging systems. It is capable of measuring the speed of incoming wind flow before it interacts with the turbine rotor. Advantages of LIDAR on improving turbine performance have been validated by high fidelity simulations and/or experiments [46–50]. The use of LIDAR in wind industry is promising as the cost for installation decreases and the corresponding control system becomes mature. The LIDAR model can be implemented with the FAST Simulink model by reading turbulent wind files generated by TurbSim before simulations. In this thesis, the wind speed measurement is assumed to be available from LIDAR for robust LPV control design.

Other advanced sensors that could be used for monitoring and/or control purposes provide measurements of turbine states and loads such as power genearation, tower top accelerations, tower base bending moments and blade root bending moments, etc [27, 51].

2.5   Baseline Control of Wind Turbines

As discussed in Section 2.1, the control of modern variable speed wind turbines is mainly focused on the wind speed range of Regions 2 and 3, since the turbine will be shut down in Regions 1 (below the cut-in wind speed) and 4 (above the cut-out wind speed). This section reviews the baseline controller for Regions 2 and 3. The structure of this simple control system is the starting point for many advanced designs. Explorations on the baseline controller are therefore helpful to better understand the principle of turbine operations and objectives in the control design.

Between the cut-in and rated wind speeds (Region 2), the control objective is to maximize the power output. As shown in Figure 2.3, the power coefficient attains its optimal value when $\lambda_* = 8.4$ and $\beta_* = 1.6 \deg$. Thus the captured power is maximized by holding blade pitch angles constant at β_* and commanding the generator torque τ_g such that the turbine operates at λ_*. As $\lambda = \frac{\omega R}{v}$, the turbine needs to operate at an optimal rotor

speed of $\omega_* = \frac{\lambda_* v}{R}$. This corresponds to an equilibrium point of the nonlinear one state model (Equation 2.4) described in Section 2.2. It can be shown that the standard control law [28, 29] achieves this goal in steady winds:

$$\tau_g = K_g \omega^2 \tag{2.14}$$

$$\beta_i = \beta_* \, (i = 1, \, 2, \, 3) \tag{2.15}$$

where the gain is chosen as $K_g = \frac{C_{p*} \rho \pi R^5}{2\lambda_*^3 N}$ and β_i is the pitch command for the i-th blade. In Equation 2.4, the generator torque τ_g is in the direction that decelerates the rotor speed. The standard control law in Equation 2.14 therefore forms a stable closed loop system by taking the sensor measurement of the rotor speed. Specifically, the rotor speed is accelerated due to the difference between aerodynamic and generator torques as wind speed changes. Commanding generator torque according to Equation 2.14 adjusts the rotor speed so that the equilibrium point ω_* is achieved in constant wind conditions.

Between the rated and cut-out wind speeds (Region 3), the objective is to maintain the rated power while minimizing the structural loads on the turbine. To maintain the rated power output, the generator torque is held constant at its rated value τ_{g_rated} in Region 3. The blade pitch angles are collectively controlled to maintain rotor speed at its rated value ω_{rated}. Therefore, the baseline controller in Region 3 can be expressed as:

$$\tau_g = \tau_{g\_rated} \tag{2.16}$$

$$\delta_{\beta_i}(s) = K_b(s)\delta_\omega(s) \, (i = 1, \, 2, \, 3) \tag{2.17}$$

where $\delta_{\beta_i} = \beta_i - \beta_0$ and $\delta_\omega = \omega - \omega_{rated}$. Here β_0 is a constant, trim blade pitch angle. A classical PI or PID controller $K_b(s)$ [28,29] can be designed based on the linearized model of the wind turbine at the trim condition of $(\omega_{rated}, \tau_{g\_rated}, \beta_0)$. It should be noted that the same pitch command is used for all three blades. This is called "collective" pitch control.

The transition between Regions 2 and 3 is commonly referred to as Region 2.5. Region 2.5 is introduced because the rated rotor speed is usually reached before the Region 2 control law reaches the rated torque. A linear torque vs. rotor speed relation is typically used to ramp from the standard $\tau_g = K_g \omega^2$ to the rated torque [8] as the wind speed approaches the rated value. This kind of blending ensures a smooth transition between Region 2 and Region 3 control objectives.

Chapter 3

LPV Control for Traditional Operations

3.1   Motivation

Modern utility scale, variable speed wind turbines are essentially nonlinear MIMO systems with distinct objectives in different wind conditions, as described in Chapter 2. The aerodynamics and structure dynamics of wind turbines also vary with the wind speed. Therefore, the baseline controller in Section 2.5 uses two independent control loops to achieve specific objectives in different regions of wind speed and ensure a smooth transition when the wind condition changes. This baseline controller has been widely accepted by the industry as an effective design. However, as the size of wind turbines grows and the structural dynamics become more flexible, considerations on load reduction are more critical for larger wind turbines [10, 27]. Therefore, extra control loops were proposed to improve the load reduction performance, such as individual pitch control [9, 42, 52] and tower and drive train dampers [10–13]. These methods significantly decreased the turbine loads but the control structure also became more complicated. There are also other concerns, such as potential dynamic couplings between different control loops. An alternative approach is to consider multiple control objectives in a systematic MIMO design [53]. This approach has been adopted in some papers and shown as a better solution than the design with multiple SISO loops [23, 54, 55]. Therefore, this chapter will consider a MIMO design for traditional operations of the C96 2.5 MW turbine using linear parameter varying (LPV) control techniques. Here, traditional operations refer to objectives of maximizing power generation in low wind speeds and tracking the rated power in high wind speeds as described in Section 2.1. They are in contrast to the mode of active power control (APC), which will be introduced in Chapter 5.

Theories on LPV systems and control have been developed since more than 2 decades ago [56–58]. The control design for LPV systems has been verified in various applications, either by high fidelity simulations or experiments [59]. It is promising to apply LPV control techniques on wind turbines, as it is essentially developed for MIMO control purposes. Therefore, existing results based on SISO linear control design as discussed above can be integrated into a uniform structure. Moreover, comparing to the classical LTI control method, LPV control takes dynamics variations of the system into considerations. In the application to wind turbines, it will be capable of achieving multiple objectives in different wind conditions and ensuring uniform performance and smooth transitions when the system dynamics changes with the wind speed. For these reasons, LPV control is an interesting topic in the field of wind energy [12,23,55,60–64]. Existing results on this topic can be first categorized by their specific objectives. For instance, LPV control is used in [12] to cover the wind speed range of Region 3 for better generator speed tracking. The complete control system also contains additional loops for tower loads mitigation using the H_∞ design. In [55], an LPV controller is synthesized to improve the load reduction performance in Region 3 and an extra anti-windup LPV design in Region 2.5 is used to ensure bumpless transfer to Region 2. In [23] and [61], a uniform LPV design is proposed that covers all operation regions of the wind speed. Parameter varying weighting functions are included in the design for multi-objectives in different wind conditions. LPV control has also been investigated for fault tolerant control of wind turbines [65]. Other related research directions in this field include LPV model reductions [66] and integrated designs for structural and control improvements [67].

Existing results on LPV control of wind turbines can be further categorized by modeling and design methods. There are traditionally two ways for modeling LPV systems. In the first way, state space matrices of the model have a rational dependence on scheduling parameters [56,68]. Models with rational parametric dependence are called LFT (linear fractional transformation) based LPV systems, as they can be expressed as a feedback interconnection of an LTI system and a diagonal block of scheduling parameters. LFT based LPV systems are usually derived by approximating a rational dependence of existing LTI models at different trim conditions. The control synthesis for LFT based LPV system has been developed in [56,69] and lead to finite dimensional linear matrix inequalities (LMIs). Related designs for wind turbines can be found in [12,60]. Another group of LPV systems are called gridding based LPV systems [57,58]. State space matrices of gridding based LPV systems have an arbitrary dependence on scheduling parameters. They are usually derived by linearization of nonlinear models and are more general in applications due to the assumption of the arbitrary dependence. However, the control synthesis for gridding based LPV systems leads to infinite LMIs [57,58]. A remedy to this problem is to approximate the LPV system with LTI models on a finite gridding set of scheduling parameters. Wind

turbine control designs using this approach can be found in [55, 60–62].

In this chapter, a uniform LPV control design that covers all wind conditions will be proposed. A gridding based LPV model of the turbine will be constructed from linearizations of the FAST turbine model at different wind speeds. The scheduling parameter is therefore naturally chosen as the trim wind speed. The proposed LPV controller is able to maximize the power generation in Region 2 and track the rated generator speed in Region 3. Considerations on load reduction will also be parts of the design. The recently developed LPV toolbox in Matlab [18] will be used to synthesize the LPV controller. The stability and performance are therefore guaranteed when the system dynamics changes with the scheduling wind speed. To overcome the conservativeness in the design, parameter varying rates will be considered. Consequently, parameter dependent Lyapunov functions will be used to solve the LMIs with extra computational and time consumptions. The synthesized controller will be compared with a baseline controller which is similar to the design in Section 2.3. Simulations and analysis show that the proposed LPV controller meets all performance objectives in different wind conditions and has better load reduction effects than the baseline controller.

The contents in this chapter are organized as follows. In Section 3.2, a brief review will be presented on modeling and control for gridding based LPV systems. Section 3.3 will provide an overview of the proposed LPV controller. Details on the modeling and design of the LPV controller will be shown in Section 3.4. Simulations and load analysis will be provided in Section 3.5.

3.2   Induced L_2 Control of LPV Systems

Linear parameter varying (LPV) systems are a class of systems whose state-space matrices depend on a time-varying parameter vector $\rho : \mathbb{R}^+ \to \mathbb{R}^{n_\rho}$. The parameter vector is assumed to be a continuously differentiable function of time. In addition, admissible trajectories are restricted, based on physical considerations, to lie in a known compact subset $\mathcal{P} \subset \mathbb{R}^{n_\rho}$ at each point in time. In many cases, the bounds on the parameters take the simple form of a hyperrectangle, i.e. $\mathcal{P} := \{\rho \in \mathbb{R}^{n_\rho} | \underline{\rho}_i \leq \rho_i \leq \bar{\rho}_i, \ i = 1, \ldots, n_\rho\}$. The set of admissible trajectories is defined as $\mathcal{T} := \{\rho : \mathbb{R}_+ \to \mathbb{R}^{n_\rho} \ : \ \rho(t) \in \mathcal{P} \ \forall t \geq 0 \text{ and } \rho(t) \text{ is continuously differentiable}\}$. In some applications, the parameter rates of variation $\dot{\rho}$ are assumed to be bounded. However, only the rate unbounded case is listed here for simplicity. Full results with the rate bounded case can be found in [57, 58].

The state-space matrices of an LPV system are continuous functions of the parameters: $A : \mathcal{P} \to \mathbb{R}^{n_G \times n_G}$, $B : \mathcal{P} \to \mathbb{R}^{n_G \times n_d}$, $C : \mathcal{P} \to \mathbb{R}^{n_e \times n_G}$ and $D : \mathcal{P} \to \mathbb{R}^{n_e \times n_d}$. An n_G^{th} order

LPV system, G_ρ, is defined by

$$\begin{bmatrix} \dot{x}(t) \\ e(t) \end{bmatrix} = \begin{bmatrix} A(\rho(t)) & B(\rho(t)) \\ C(\rho(t)) & D(\rho(t)) \end{bmatrix} \begin{bmatrix} x(t) \\ d(t) \end{bmatrix} \tag{3.1}$$

The state matrices at time t depend on the parameter vector at time t. Hence, LPV systems represent a special class of time-varying systems. Throughout the remainder of the thesis the explicit dependence on t is occasionally suppressed to shorten the notation. Moreover, it is important to emphasize that the state matrices are allowed to have an arbitrary dependence on the parameters. This is in contrast to the LFT based LPV systems in [56, 68], where the state matrices are assumed to be rational functions of ρ. The performance of an LPV system G_ρ can be specified in terms of its induced L_2 gain from input d to output e assuming the initial condition $x(0) = 0$, i.e. it is defined as

$$\|G_\rho\| := \sup_{0 \neq d \in L_2, \ \rho \in \mathcal{T}} \frac{\|e\|}{\|d\|}. \tag{3.2}$$

In words, this is the largest input/output gain over all possible inputs $d \in L_2$ and allowable trajectories $\rho \in \mathcal{T}$. The notation $\rho \in \mathcal{T}$ refers to the entire (admissible) trajectory as a function of time. The analysis and synthesis theorems summarized below involve conditions on the parameters at a single point in time, i.e. $\rho(t)$. The parametric description $\rho \in \mathcal{P}$ is introduced to emphasize that such conditions only depend on the (finite-dimensional) set $\mathcal{P}$. A generalization of the Bounded Real Lemma is stated in [58] which provides a sufficient condition to upper bound the induced L_2 gain of an LPV system. The sufficient condition uses a quadratic, parameter-dependent storage function. The next theorem states the condition provided in [57, 58] but simplified for the special case of rate unbounded LPV systems.

Theorem 1 ([57,58]). *Let $\mathcal{P}$ be a given compact set and G_ρ an LPV system (Equation 3.1). G_ρ is exponentially stable and $\|G_\rho\| \leq \gamma$ if there exists a matrix $P = P^T \geq 0$ such that $\forall \rho \in \mathcal{P}$*

$$\begin{bmatrix} PA(\rho) + A(\rho)^T P & PB(\rho) \\ B^T(\rho)P & -I \end{bmatrix} + \frac{1}{\gamma^2} \begin{bmatrix} C(\rho)^T \\ D(\rho)^T \end{bmatrix} \begin{bmatrix} C(\rho) & D(\rho) \end{bmatrix} < 0 \tag{3.3}$$

Proof. The proof is based on a dissipation inequality satisfied by the storage function $V(x) = x^T P x$. The proof is sketched as similar arguments are used throughout the thesis. Let $d \in L_2$ be an arbitrary input and $\rho \in \mathcal{T}$ be any admissible parameter trajectory. Let x and e denote the state and output responses of G_ρ for the input d and trajectory ρ assuming

$x(0) = 0$. Multiplying Equation 3.3 on the the left/right by $[x^T, d^T]$ and $[x^T, d^T]^T$ gives

$$\dot{V}(t) \leq d(t)^T d(t) - \gamma^{-2} e(t)^T e(t) \tag{3.4}$$

Integrating this dissipation inequality yields the conclusion $\|G_\rho\| \leq \gamma$. The proof of exponential stability is similar. $\qquad\square$

This analysis theorem forms the basis for the induced L_2 norm controller synthesis in [57,58]. The results in [57,58] are briefly summarized for the rate unbounded case. Consider an open loop LPV system G_ρ defined as

$$\begin{bmatrix} \dot{x} \\ e \\ y \end{bmatrix} = \begin{bmatrix} A(\rho) & B_d(\rho) & B_u(\rho) \\ C_e(\rho) & D_{ed}(\rho) & D_{eu}(\rho) \\ C_y(\rho) & D_{yd}(\rho) & D_{yu}(\rho) \end{bmatrix} \begin{bmatrix} x \\ d \\ u \end{bmatrix} \tag{3.5}$$

where $x \in \mathbb{R}^{n_G}$, $d \in \mathbb{R}^{n_d}$, $e \in \mathbb{R}^{n_e}$, $u \in \mathbb{R}^{n_u}$ and $y \in \mathbb{R}^{n_y}$. The goal is to synthesize an LPV controller K_ρ of the form:

$$\begin{bmatrix} \dot{x}_K \\ u \end{bmatrix} = \begin{bmatrix} A_K(\rho) & B_K(\rho) \\ C_K(\rho) & D_K(\rho) \end{bmatrix} \begin{bmatrix} x_K \\ y \end{bmatrix}. \tag{3.6}$$

The controller generates the control input u. It has a linear dependence on the measurement y but an arbitrary dependence on the (measurable) parameter vector ρ. The closed-loop interconnection of G_ρ and K_ρ is given by a lower linear fractional transformation (LFT) and is denoted $\mathcal{F}_l(G_\rho, K_\rho)$. The objective is to synthesize a controller K_ρ of the specified form to minimize the closed-loop induced L_2 gain from disturbances d to errors e:

$$\min_{K_\rho} \|\mathcal{F}_l(G_\rho, K_\rho)\|. \tag{3.7}$$

The notation for the synthesis result below is greatly simplified by assuming the feedthrough matrices satisfy $D_{ed}(\rho) = 0$, $D_{yu}(\rho) = 0$ and $D_{eu}(\rho)^T = [0, \ I_{n_u}]$, $D_{yd}(\rho) = [0, \ I_{n_y}]$. Under some technical rank assumptions, this normalized form can be achieved through a combination of loop-shifting and scaling [57, 70]. The input matrix is partitioned as $B_d(\rho) := \begin{bmatrix} B_{d1}(\rho) & B_{d2}(\rho) \end{bmatrix}$ compatibly with the normalized form of D_{yd}. Similarly, the output matrix is partitioned as $C_e^T(\rho) := \begin{bmatrix} C_{e1}^T(\rho) & C_{e2}^T(\rho) \end{bmatrix}$ compatibly with D_{eu}. Given these simplifying assumptions, the solution to the induced L_2 control synthesis problem is stated in the next theorem.

Theorem 2 ([57,58])**.** *Let $\mathcal{P}$ be a given compact set and G_ρ an LPV system (Equation 3.5)*

27

that satisfies the normalizing assumptions above. There exists a controller K_ρ as in Equation 3.6 such that $\|\mathcal{F}_l(G_\rho, K_\rho)\| \leq \gamma$ if there exist matrices $X = X^T > 0$ and $Y = Y^T > 0$ such that $\forall \rho \in \mathcal{P}$

$$\begin{bmatrix} X & I_{n_x} \\ I_{n_x} & Y \end{bmatrix} \geq 0 \tag{3.8}$$

$$\begin{bmatrix} Y\hat{A}(\rho)^T + \hat{A}(\rho)Y - \gamma B_u(\rho)B_u(\rho)^T & YC_{e1}(\rho)^T & B_d(\rho) \\ C_{e1}(\rho)^TY & -\gamma I_{n_{e1}} & 0 \\ B_d(\rho)^T & 0 & -\gamma I_{n_d} \end{bmatrix} < 0 \tag{3.9}$$

$$\begin{bmatrix} \tilde{A}(\rho)^TX + X\tilde{A}(\rho) - C_y(\rho)^TC_y(\rho) & XB_{d1}(\rho) & C_e(\rho)^T \\ B_{d1}(\rho)^TX & -\gamma I_{n_{d1}} & 0 \\ C_e(\rho) & 0 & -\gamma I_{n_e} \end{bmatrix} < 0 \tag{3.10}$$

where $\hat{A}(\rho) := A(\rho) - B_u(\rho)C_{e2}(\rho)$ and $\tilde{A}(\rho) := A(\rho) - B_{d2}(\rho)C_y(\rho)$.

Proof. The proof uses a matrix elimination argument similar to that used in the LMI approach to H_∞ synthesis for linear time invariant (LTI) systems [15]. □

If the conditions in Theorem 2 are satisfied then an LPV controller with the state space form of $(A_K(\rho), B_K(\rho), C_K(\rho), D_K(\rho))$ can be constructed from the open loop plant matrices and the feasible values of X, Y, and γ. The controller reconstruction procedure is given in [57, 58]. Moreover, a storage function matrix $P \geq 0$ can be constructed from X and Y such that the closed loop satisfies the nominal performance LMI condition (Equation 3.3) in Theorem 1. Finally, the closed-loop performance (upper bound) can be optimized by minimizing γ subject to the LMI constraints in Theorem 2. This yields a semidefinite programming formulation for the LPV synthesis problem.

It should be noted that both Theorem 1 and Theorem 2 lead to infinite collection of LMI constraints due to their arbitrary dependence on parameter $\rho \in \mathcal{P}$. A remedy to this problem, which works in many practical examples, is to approximate the set $\mathcal{P}$ by a finite set $\mathcal{P}_{grid} \in \mathcal{P}$ that represents a gridding over $\mathcal{P}$. This gridding based approach with insufficient gridding points might lead to performance degradation using the controller synthesized. Increasing the density of gridding will alleviate the risk. However, it comes with more computation time. Therefore, a compromise in practice is to use a sparse gridding set for synthesis and verify the performance afterwards using a denser gridding set, since it is faster to solve the analysis problem. If there is no significant degradation, the synthesis on the sparse gridding set can be adopted for control design. Otherwise, a denser gridding set is required for further synthesis. This gridding based approach will also be used in following chapters of this thesis.

3.3 Structure of The Proposed LPV Control System

Wind turbines are traditionally operated to maximize the generator power in below rated wind speed and to track the rated generator speed in above rated wind speed. These two main objectives can be achieved by using the baseline controller as described in Section 2.5. The baseline controller uses two SISO control loops for operating the turbine in both Region 2 and Region 3. Transition between the two control loops is required when the wind condition changes. To satisfy the performance requirements for load reduction, model based control designs can be embedded into the control system as extra SISO loops. An alternative approach is to design a MIMO controller to operate the turbine in all regions and satisfy performance objectives. This section provides an overview of the proposed LPV controller using this approach.

As shown by the block diagram in Figure 3.1, this LPV controller takes the error between the generator speed command ω_{g_cmd} and measurement ω_g as input. Outputs of this controller are generator torque τ_g and collective pitch angle β. This LPV controller has a dependence on the scheduling parameter ρ, which is the trim wind speed v_{trim}. It is assumed that an accurate and real time measurement of the wind speed is available. As shown in this figure, an estimate of the wind speed $\hat{v}$ could be obtained from a LIDAR [71]. Alternatively, an estimate of the effective wind speed could be constructed [72]. In either case, the wind speed fluctuates due to turbulence and hence low-pass filtering, denoted LPF in the figure, is used to smooth out these fluctuations and generate the trim wind speed v_{trim} from the measurement $\hat{v}$.
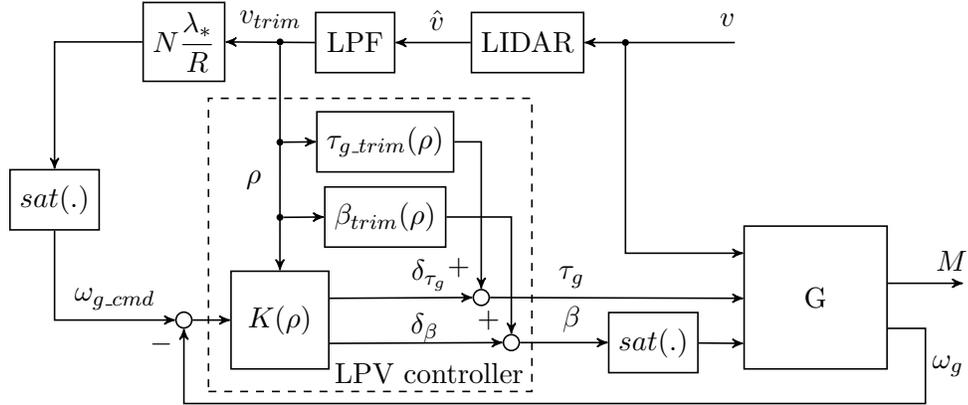
Figure 3.1: Structure of the LPV controller for traditional operations.

This controller can be designed to operate the turbine in different wind conditions according to measurement of the scheduling parameter ρ. Specifically, in below rated wind speed, the controller tracks a time varying generator speed command ω_{g_cmd} to maximize the power generation. This command ω_{g_cmd} is calculated based on the trim wind speed as

$\omega_{g\_cmd} = N\frac{\lambda_*}{R}v_{trim}$ such that the TSR is expected to stay at the optimal value λ_*. In this case, generator torque τ_g will be the main control actuation to the wind turbine while the blade pitch needs to be suppressed such that the pitch angle β will be close to the optimal value β_*. This objective can be achieved in the design by penalizing less on the generator torque and more on the blade pitch angle in low wind speed. It can be concluded from the contour plot of power coefficient C_p in Figure 2.3, that the partial derivative of C_p to β is close to 0 when C_p is close to its optimal value C_{p*}. If the optimal TSR λ_* can be well tracked through the control of generator torque, it is expected that the blade pitch actuation would have minor contributions to the power generation. Therefore the power generation will not be affected too much by the blade pitch actuation.

It is noted that this control strategy for maximizing the power generation is different from the nonlinear control law used in the baseline controller (Equations 2.14 and 2.15), which takes only the generator speed measurement for calculation of the generator torque. However, it provides a consistent structure of the feedback control loop for operating the turbine in above rated wind condition. As shown in Figure 3.1, the generator speed command ω_{g_cmd} will be saturated at the rated value ω_{g_rated} when the wind speed goes above rated. In this case, the control system looks similar to the baseline controller in Region 3 for the constant generator speed tracking (Equations 2.16 and 2.17). However, both the generator torque τ_g and blade pitch β will be used here to control the turbine. The blade pitch will be the main control input to the turbine for maintaining the rated power when wind speed goes high. The generator torque will serve as a complementary control input for alleviating the usage of blade pitching and damping out vibrations on the tower and rotor shaft. This objective can be achieved in the design by penalizing more on the generator torque and less on the blade pitch angle in high wind speed.

The use of this MIMO LPV control architecture allows the turbine to meet objectives in different wind conditions. This uniformly designed controller naturally ensures stability and performance for a smooth transition between different wind speed regions. Inside each region, the performance is also expected to be uniform by taking the parameter varying dynamics into account. Moreover, the MIMO structure of the controller takes other performance objectives, such as load reduction, into a systematic consideration. The design procedure is therefore simplified and potential side effects due to the dynamics coupling between multiple SISO control loops can be avoided. For simplicity, only collective pitch control is considered in this proposed controller. Therefore, the load reduction objective is more focused on the tower and rotor shaft. However, it will be natural to extend the design to include individual pitch control in the MIMO architecture for periodic load reduction on the blades. In the next section, details will be provided on the design of the proposed LPV controller.

3.4 LPV Design

3.4.1 LPV Model Construction

The first step of the design is to generate a gridded based LPV model of the C96 2.5 MW turbine. This LPV model is chosen to be dependent on the trim wind speed v_{trim} whose value varies from 4 m/s to 25 m/s. As discussed in Section 3.2, this will lead to an infinite collection of parameter dependent LMIs in the control synthesis. A practical way to deal with this problem is to take a finite gridding set of the scheduling parameter and generate the LTI model at each gridding point. To balance the accuracy of modeling and the complexity of design, here, 7 points are chosen uniformly in increments of 3 m/s in wind speed from 6 m/s to 24 m/s.

As described in Section 2.3, the LTI model at each trim point is generated by the FAST simulation and post analysis which consists of the MBC transformation and averaging of the resulting "weakly" PLTV system. To start the simulation, trim values for the generator speed ω_{g_trim}, collective blade pitch angle β_{trim} and generator torque τ_{g_trim} need to be determined uniquely. This step can be done once the trim wind speed v_{trim} is selected. Figure 3.2 shows these trim values as functions of v_{trim}. The nonlinear turbine model therefore can be linearized around the selected gridding point based on these trim values. As shown in Figure 3.1, trim values for the generator torque τ_{g_trim} and blade pitch angle β_{trim} will also be used in the construction of the complete LPV controller by feed-forwarding them to the control inputs.

The FAST simulation package includes up to 24 DOFs for the linearization. The wind turbine model in this chapter uses 9 of them that include the rotor position, first tower fore-aft and side-to-side bending modes, and first flapwise and edgewise bending modes for each blade. This choice of model complexity will be sufficient for the modeling accuracy and also helpful for saving the computational cost in the synthesis of the LPV controller. The corresponding linearized model as described in Equation 2.7 contains 18 states. To avoid numerical issues in the control synthesis, the state of the rotor position is removed after the post analysis and the resulting LTI model has 17 states. It should be noted that the DOF of the drive train torsion is usually included in the model to ensure that the designed controller damps out the vibration on the rotor shaft. However, the C96 turbine has a rigid design of the rotor shaft such that the mode for the drive train is lightly damped at 178 rad/s. This is far beyond the bandwidth of actuators and the wind turbulence usually has minor effects at this frequency. Therefore, it is reasonable to exclude this DOF in the model.

As introduced in Section 2.3.1, there are up to 5 control inputs for the wind turbine, which
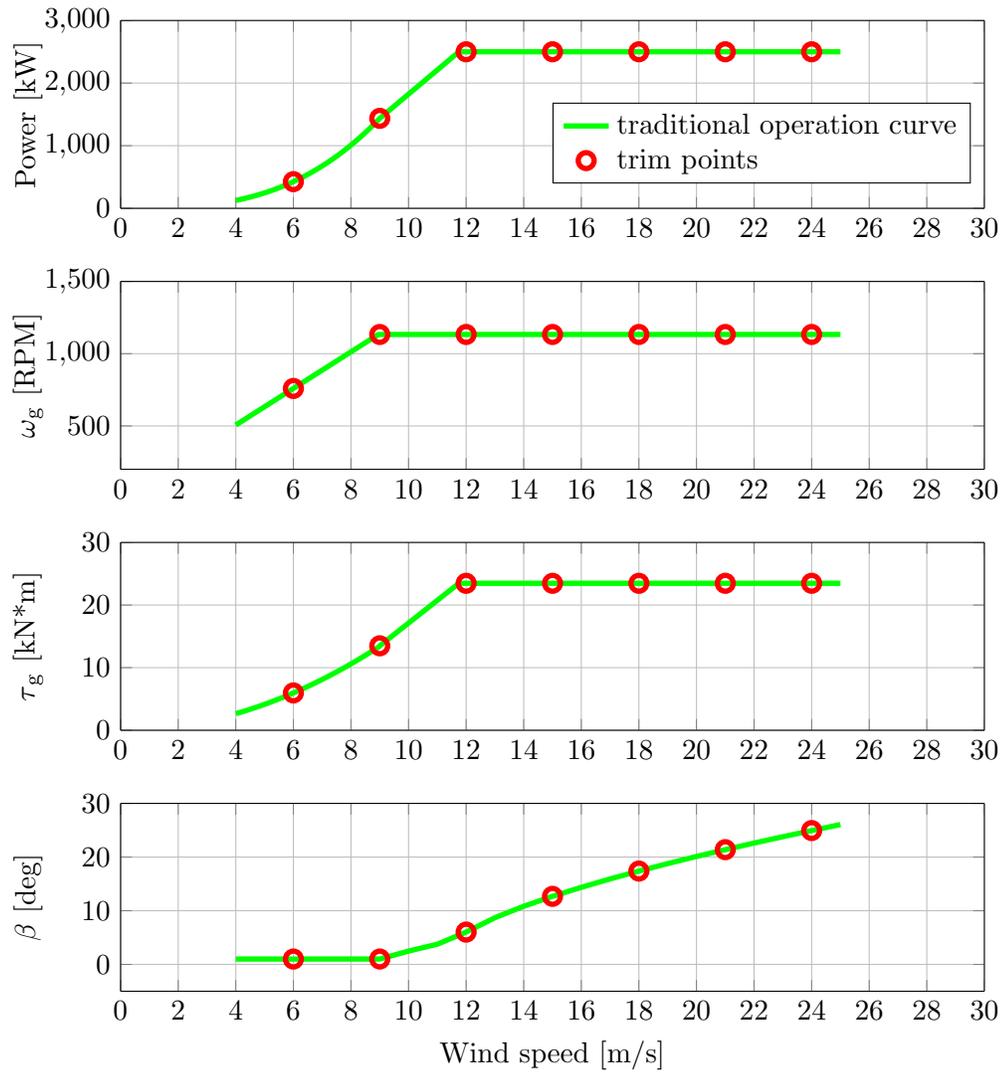
Figure 3.2: Trim values for linearization.

include the generator torque, individual pitch angles for three blades and yaw angle. Since collective pitch control is considered here, there is only 1 control input channel for the blade pitch actuation. Moreover, the yaw angle input is disabled here as it is held constant at $0 \deg$ to the incoming wind flow. Therefore, the generator torque τ_g and collective pitch angle β will be used in the model. Wind disturbance inputs are also simplified to 1 channel for the hub-height wind speed v which captures the most important characteristics of wind conditions. The first output of the model is the sensor measurement of the generator speed ω_g. It will be used as a feedback signal in the control loop. To penalize the tower loads in the design, tower fore-aft and side-to-side bending moments M_{tfa} and M_{tss} are selected as the other two outputs. However, it should be noted that M_{tfa} and M_{tss} are only used in the model for control design and need not to be measured for feedback purposes in simulations. This is in contrast to the stardard damper designs proposed in [10–13].

To conclude, the resulting LPV model of the turbine $G(\rho)$ is given by state space expression of the form

$$
\begin{bmatrix} \dot{x} \\ y \end{bmatrix} = \begin{bmatrix} A(\rho) & B_d(\rho) & B_u(\rho) \\ C(\rho) & D_d(\rho) & D_u(\rho) \end{bmatrix} \begin{bmatrix} x \\ d \\ u \end{bmatrix}
\tag{3.11}
$$

where $x \in \mathbb{R}^{17}$ is the state, $d := \delta v \in \mathbb{R}$ is the disturbance, $u := \begin{bmatrix} \delta\tau_g & \delta\beta \end{bmatrix}^T \in \mathbb{R}^2$ is the vector of inputs and $y := \begin{bmatrix} \delta M_{tfa} & \delta M_{tss} & \delta\omega_g \end{bmatrix}^T \in \mathbb{R}^3$ is the vector of outputs.

It should be noted that the MBC transformation is used to get the LTI model at each trim point. It takes inputs and outputs of the model in the rotating frame to the non-rotating frame. Model based control designs are therefore all in the non-rotating frame. If these inputs and outputs are included in the feedback control loop, the MBC transformation must also be implemented as a part of the controller. For example, three pitch angles in the non-rotating frame are expressed as $\beta^{nr} = \begin{bmatrix} \beta^{nr}_{avg} & \beta^{nr}_{yaw} & \beta^{nr}_{tilt} \end{bmatrix}^T$ for the individual blade pitch control. In FAST simulations, these control signals need to be transformed back to the rotating frame. However, this transformation will not be included in the proposed controller as only collective pitch control is considered and there are no sensor measurements from blades to be used in the design.

3.4.2 Weights Tuning

Due to the dependence on scheduling parameters, LPV systems do not have a valid frequency interpretation. However, as a state space design method in the time domain, the LPV control synthesis can still be achieved by adopting loop shaping techniques to satisfy the required performance. This property makes the tuning for LPV controllers similar to the

process in the classical H_∞ design. Specifically, in the LPV control design, performance objectives can be specified by weighting transfer functions in the frequency domain. These weights will be transformed into the state space form and interconnected with the original plant as an augmented system for synthesis. The freedom for weight tuning is even extended in the LPV control as different weights can be specified at the same channel for performance trade off when the system dynamics vary with scheduling parameters. Therefore, these weights are also considered as linear parameter varying.

Figure 3.3: The augmented system for LPV synthesis.

Figure 3.3 shows the augmented system for synthesis of the proposed LPV controller. Here, 5 weights (W_e, W_τ, W_β, W_v and W_m) need to be selected for the loop shaping. Considerations on using parameter varying weights are more critical in such a design as the proposed LPV controller has distinct objectives in different wind conditions. Table 3.1 shows detailed expressions of these weights in different trim points. Bode magnitude plots are also provided in Figure 3.4 for W_e, W_τ, W_β and W_v. As shown in Table 3.1, one group of weights (denoted as $W_e^{(1)}$, $W_\tau^{(1)}$, $W_\beta^{(1)}$, $W_v^{(1)}$ and $W_m^{(1)}$ in Figure 3.4) are selected for ρ at $6\,\text{m/s}$ and $9\,\text{m/s}$, which correspond to the operation in Region 2. Another group of weights ($W_e^{(3)}$, $W_\tau^{(3)}$, $W_\beta^{(3)}$, $W_v^{(3)}$ and $W_m^{(3)}$) are chosen for ρ varying from $15\,\text{m/s}$ to $24\,\text{m/s}$, which correspond to the operation in Region 3. The weights at $\rho = 12\,\text{m/s}$ ($W_e^{(2)}$, $W_\tau^{(2)}$, $W_\beta^{(2)}$, $W_v^{(2)}$ and $W_m^{(2)}$) are used to ensure a smooth transition between different wind conditions.

W_e is the performance weight that specifies the objective for generator speed tracking. It is chosen to limit the low frequency error with less emphasis on high frequency tracking. The bandwidth of W_e is selected to be $0.2\,\text{rad/s}$. This is a conservative setting to ensure that the induced L_2 norm of the closed loop LPV system will be close to 1. However, this value should be sufficient fast for the objective of tracking. Choices for the low frequency gain of W_e varies for different wind conditions. In Region 3, the generator speed tracking is the main objective and the low frequency gain of W_e is close to 5 which corresponds to a desired steady-state error of $0.2\,\text{RPM}$. In Region 2, the generator speed is allowed to be

34

Figure 3.4: Bode magnitude plots of weights.

Table 3.1: Weights at different trim points.

ρ	$\begin{bmatrix} 6\ 9 \end{bmatrix}$	$\begin{bmatrix} 12 \end{bmatrix}$	$\begin{bmatrix} 15\ 18\ 21\ 24 \end{bmatrix}$
W_e	$\frac{0.2s+0.1058}{s+0.1323}$	$\frac{0.25s+0.1323}{s+0.07937}$	$\frac{0.25s+0.1323}{s+0.02646}$
W_τ	$\frac{0.04s+0.12}{s+60}$	$\frac{0.12s+0.6}{s+100}$	$\frac{0.16s+0.8}{s+100}$
W_β	$\frac{1000s+1890}{s+189}$	$\frac{500s+1134}{s+226.8}$	$\frac{500s+1134}{s+226.8}$
W_v	$\frac{0.0625s+0.04147}{s+1.327}$	$\frac{0.0625s+0.04147}{s+1.327}$	$\frac{0.1333s+0.1418}{s+1.772}$
W_m	$\begin{bmatrix} 10^{-3.65} & 0 \\ 0 & 10^{-3.25} \end{bmatrix}$	$\begin{bmatrix} 10^{-3.65} & 0 \\ 0 & 10^{-3.25} \end{bmatrix}$	$\begin{bmatrix} 10^{-3.65} & 0 \\ 0 & 10^{-3.25} \end{bmatrix}$

more sensitive to the wind disturbance in the low frequency for capturing more power and relaxing the usage of generator torque. Therefore, the low frequency gain is adjusted to 0.8 in Region 2.

Next, W_τ and W_β are two weights used to penalize actuations of the generator torque and blade pitch angle, respectively. Both weights are chosen as high pass filters to penalize high frequency control effort. The generator torque can be actuated sufficiently fast that its actuator dynamics can be neglected. However, the weight W_τ is still required to avoid aggressive generator torque commands and hence accommodate the use of the blade pitch actuator. W_τ is chosen to have a bandwidth of $3\,\mathrm{rad/s}$ in Region 2 and $5\,\mathrm{rad/s}$ in Region 3. The bandwidth for W_β is chosen to be $5\,\mathrm{rad/s}$ in Region 2 and $6\,\mathrm{rad/s}$ in Region 3. These values are lower than the bandwidth of blade pitch actuator which is usually around $10\,\mathrm{to}\,15\,\mathrm{rad/s}$. As discussed in Section 3.3, the trade off between these two actuators varies with wind conditions. Therefore, the low and high frequency gains of W_τ in Region 3 is 4 times higher than the values in Region 2. Correspondingly, these gains of W_β in Region 2 is twice higher than the values in Region 3. This enables a proper coordination of the two control inputs for objectives in different wind conditions. The difference of these two weights in Region 2 and Region 3 are also shown in Figure 3.4.

The weight W_v is used to shape the frequency property of wind disturbance. The power spectrum of wind turbulence usually has a high gain in the low frequency and then gradually decays in the high frequency [52]. Therefore, W_v is chosen to be as a low pass filter in [52]. However, it is shown in the tuning process that this choice of W_v does not have enough emphasis on tower modes (around $2\,\mathrm{rad/s}$) and one of blade edgewise bending modes (around $13\,\mathrm{rad/s}$). These modes have significant effects on loads of the tower and rotor shaft. Therefore, W_v is adjusted to be as a high pass filter with slightly higher gain in the high frequency region. Moreover, W_v is more relaxed in Region 2 than in Region 3 as the

requirement for load reduction is less critical in Region 2.

Finally, the weight W_m is used to penalize the bending moments on the tower base in both fore-aft and side-to-side directions. W_m is assumed to be diagonal without considering the coupling between these two loads. 2 diagonal entries in W_m are constant in all wind conditions. The specific values of these two gains are chosen based on $W_v^{(3)}$ and the channels of $G(\rho)$ from the wind disturbance to these two loads in Region 3. For example, the peak gain for the augmented system from $\delta \tilde{d}$ to δM_{tfa} is 70 dB at $\rho = 18$ m/s. Therefore, the first diagonal term of W_m is chosen as $10^{-3.65}$ such that the weighted open loop gain from $\delta \tilde{v}$ to $\delta \tilde{M}_{tfa}$ is smaller than 1.

Once these weights have been selected, the synthesis of the LPV controller can be performed using the LPV toolbox in Matlab [18]. While tuning all these weights for different wind conditions in one time might be complicated and time consuming, it is more convenient to do the tuning separately for two LTI models in Region 2 and Region 3 using the classical H_∞ design method. For instance, two H_∞ controllers can be designed after several iterations for trim points at 9 m/s and 18 m/s respectively. The weights used for the LTI model at $\rho = 9$ m/s will be denoted as $W_e^{(1)}$, $W_\tau^{(1)}$, $W_\beta^{(1)}$, $W_v^{(1)}$ and $W_m^{(1)}$ for Region 2. Similarly, the weights used at $\rho = 18$ m/s will be adopted for Region 3. As described above, weights at $\rho = 12$ m/s should ensure a smooth transition as objectives change with wind conditions. Therefore, these weights will be determined at last. An even more careful design is possible by tuning weights individually at each trim point. However, the price is more time and work. As what will be shown in Section 3.4.3, the weights selected in this section are sufficient to ensure the required performance.

3.4.3   Synthesis Results

The synthesized LPV controller $K(\rho)$ for the augmented system described in Section 3.4.2 has a dependence on the parameter varying rate, which is the acceleration of the trim wind speed v_{trim}. The choice of an unbounded rate leads to two constant Lyapunov matrices in the synthesis, as described in Section 3.2. This will greatly accelerate the process for solving LMIs but lead to a significant conservativeness in the result. Therefore, it is necessary to choose a reasonable parameter varying rate in the synthesis. Here, the value is selected as 0.1 m/s^2. This value is calculated by first low pass filtering a 11 % turbulent wind profile at an averaged wind speed of 20 m/s. The bandwidth of the low pass filter is 0.02 rad/s and the resulting signal is considered as the trim wind speed for scheduling. This trim wind speed is further high-pass filtered to get the acceleration. The turbulent wind profile used here is a Class C level wind turbulence of IEC 64000-1 standards [33], which corresponds to normal wind conditions in practice. The choice of a bounded rate results in the requirement

for parameter dependent Lyapunov matrices in the synthesis. For simplicity, an affine dependence is chosen here for the synthesis and the calculated upper bound of the induced L_2 norm of the closed loop LPV system is close to 1. The time consumption for the synthesis is around 180 s. As a comparison, an unbounded parameter varying rate leads to an upper bound of the nominal gain at 3.5 and the computation time is 13 s. Figure 3.5 shows more synthesis results using different parameter varying rates for the proposed LPV controller with 7 gridding points. Results for the LPV controller with 20 gridding points in this figure will be discussed later. It is seen in the figure that the upper bound of the induced L_2 norm does not increase significantly until $10 \, \mathrm{m/s^2}$, which is unrealistic in practice.
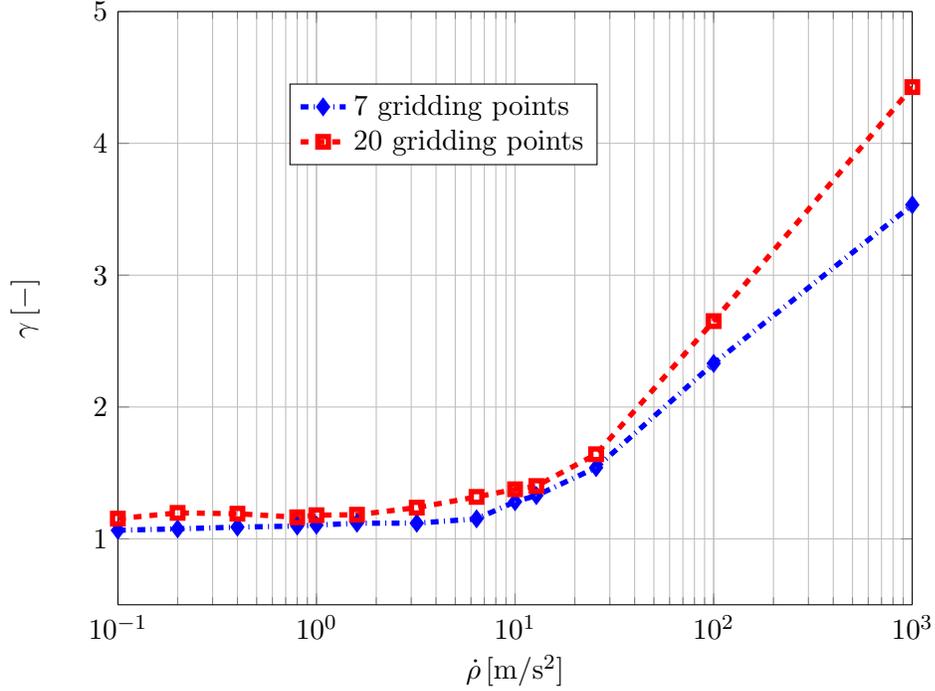
Figure 3.5: Upper bound of the induced L_2 norm γ with $\dot{\rho}$.

To verify that the choice of 7 gridding points in the LPV model is sufficient for ensuring the accuracy, a denser gridding set is chosen in increments of $1 \, \mathrm{m/s}$ in wind speed from $5 \, \mathrm{m/s}$ to $24 \, \mathrm{m/s}$. The resulting LPV model contains 20 trim points. Synthesis results using this model are also shown in Figure 3.5. It is noted that the upper bound of the induced L_2 norm with 20 gridding points is slightly larger than the value with 7 gridding points. However, the computation time is around 700 s using the affine parametric dependence. The unbounded parameter varying rate is also considered for the model using the denser gridding set. It leads to an upper bound of the nominal gain at 4.4 and the time consumption is 41 s. Therefore, the selected gridding set with 7 points for the LPV model does not lead to a significant performance degradation.

Before implementing the synthesized LPV controller in the FAST simulation environment, it is necessary to check that it would meet desired performance objectives. In the traditional LTI control design, this step is achieved by checking frequency responses of the closed loop system. As discussed in Section 3.4.2, LPV systems do not have a valid frequency interpretation. However, it is still valuable to perform the frequency analysis for LTI systems on some representative frozen trim points. This frequency analysis will be helpful at least on exploring the local performance of the LPV controller at neighborhoods of these trim points.

Figures 3.6, 3.7 and 3.8 show Bode magnitude plots of the closed loop system from the wind disturbance δv [m/s] to some outputs at 3 trim wind conditions. The 4 outputs selected here are generator speed $\delta \omega_g$ [RPM], torque on the high speed shaft $\delta \tau_{hss}$ [kN $\cdot$ m], tower base fore-aft bending moment δM_{tfa} [kN $\cdot$ m] and side-to-side bending moment δM_{tss} [kN $\cdot$ m]. These figures also show corresponding Bode magnitude plots for the open loop system and the closed loop system using a baseline controller which is similar to the one introduced in Section 2.5. This baseline controller is currently implemented on the C96 turbine for operations. Due to the reason of intellectual properties, however, details on the baseline controller are hidden here and these plots for the baseline controller are generated using the method of system identification as in Section 2.3.3.

Figure 3.6 shows the comparison at $\rho = 9$ m/s which is a typical wind condition in Region 2. Here, the generator speed of the LPV controller is less sensitive to the wind disturbance in the low frequency than the baseline controller, which might lead to less power generation. However, as shown in Figure 3.1, ω_g is designed to track a low frequency command for Region 2 in the proposed control system. As long as the weight W_e is enough to bound the sensitivity function from the generator speed command to the error, it is expected that the performance for maximizing power generation would not be affected too much. Here, both the LPV controller and the baseline controller have similar frequency responses for the high speed shaft torque in the low frequency. An extra actuation of the generator torque for the LPV controller at the frequency range 1 to 3 rad/s leads to a little increase on the high speed shaft torque. However, as the power in the wind turbulence decays fast with the increase of frequency, it will contribute few to $\delta \tau_{hss}$ in this frequency range. Instead, this extra actuation of the generator torque on the LPV controller significantly decreases the peak of δM_{tss}, as shown in the lower left subplot of Figure 3.6. The tower base fore-aft bending moment δM_{tfa} is less affected by the LPV controller at this trim point. One possible reason might come from choices of the constant weight W_m and the relaxed weight $W_v^{(1)}$ in Region 2.

Figure 3.7 shows frequency responses at $\rho = 12$ m/s which is a transition wind speed between Region 2 and Region 3. In this trim point, both the LPV controller and the baseline
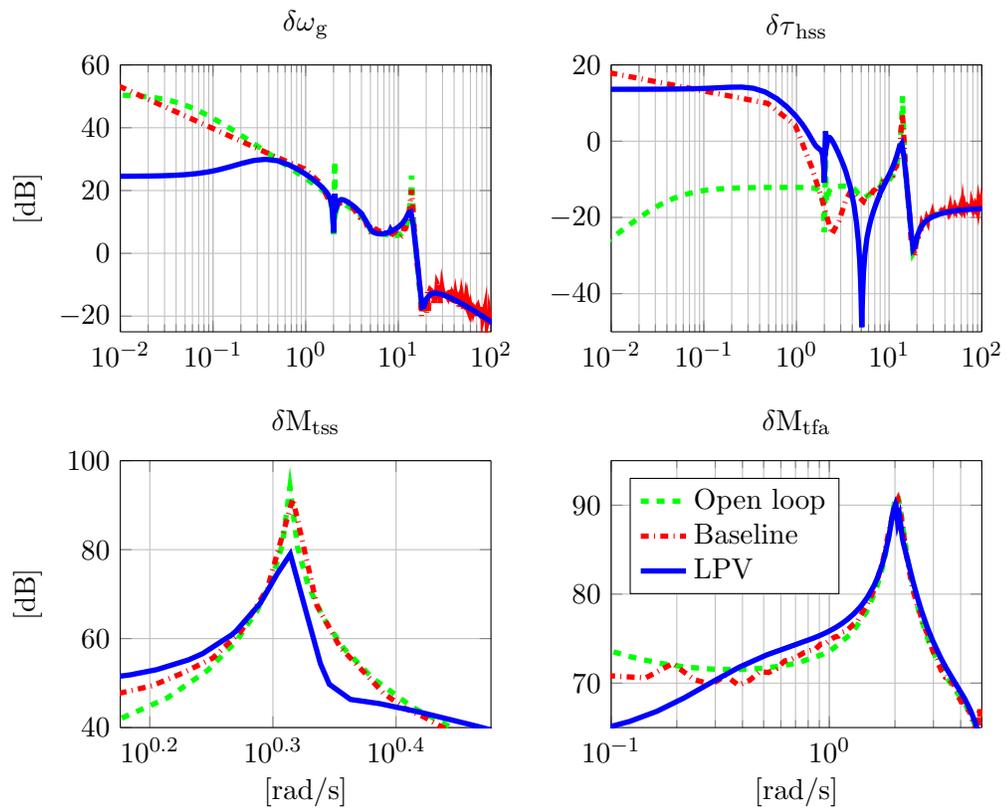
Figure 3.6: Bode magnitude plots from wind disturbance to some outputs at $\rho = 9\,\mathrm{m/s}$.

controller start to track the rated generator speed for Region 3. However, $\delta\omega_g$ is more sensitive to the wind disturbance in the low frequency for the baseline controller. It is also noted that loads on both the tower and high speed shaft start to decrease for the LPV controller. This is to be expected as load reduction is more important as the wind speed increases.
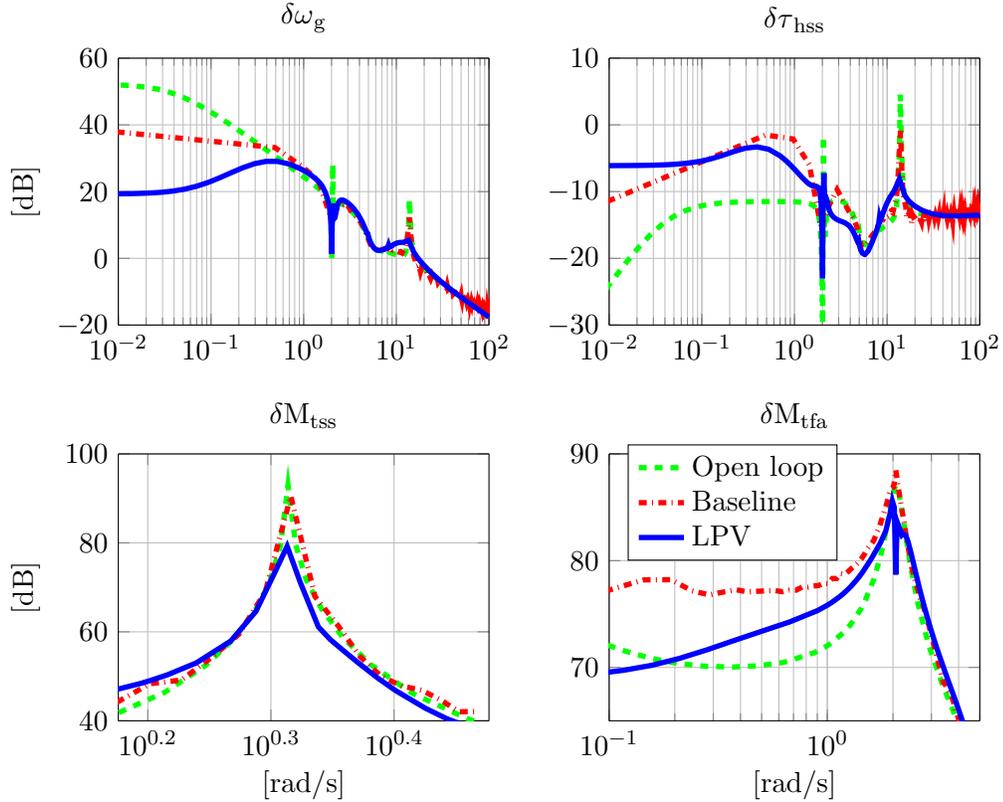
Figure 3.7: Bode magnitude plots from wind disturbance to some outputs at $\rho = 12\,\text{m/s}$.

The performance for the LPV controller is even better in Region 3. As shown in Figure 3.8, frequency responses for the LPV controller at the trim point of $18\,\text{m/s}$ have lower gains almost everywhere, comparing to responses of the baseline controller. It indicates a better generator speed tracking performance and better load reductions on the tower and the high speed shaft. It is noted that the peak of δM_{tfa} does not decrease too much for the LPV controller. In the tuning process, it is found out that this peak changes less with tighter weight on W_m. However, the frequency response around this peak significantly decreases for the LPV controller. Therefore, it is still reasonable to expect that the load reduction performance will be better for the LPV controller. A possible solution to push down the peak for further load reduction in the future might be to include the tower top (nacelle) fore-aft velocity measurement for a damper design.
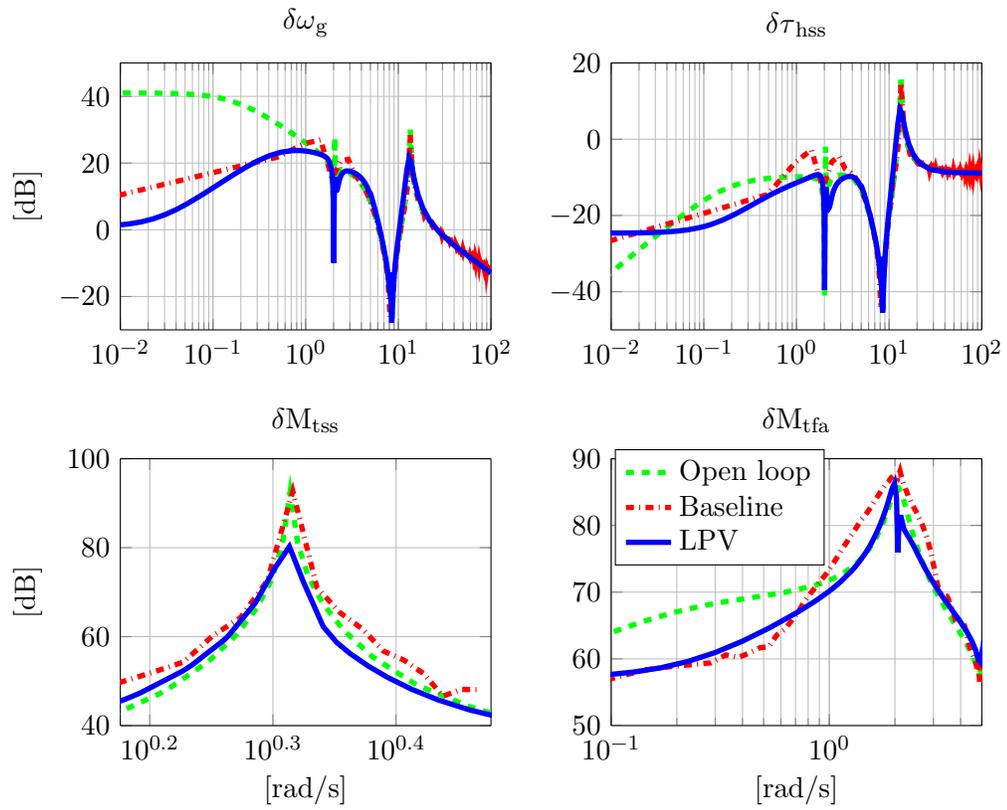
Figure 3.8: Bode magnitude plots from wind disturbance to some outputs loads at $\rho = 18\,\text{m/s}$.

To conclude results in the frequency analysis above, the designed LPV controller should meet performance requirements in different wind conditions. According to FAST simulations and post analysis which will be shown in the next section, the frequency analysis introduced here is an effective way to predict the performance of the synthesized LPV controller. This is important for the working flow as it saves more time in design iterations.

3.5   Simulations and Analysis

3.5.1   Simulation Results

The synthesized LPV controller will be tested in the FAST simulation environment. Structural modes in the FAST model include the rotor position, drive train torsion, first tower fore-aft and side-to-side bending modes, and first flapwise and edgewise bending modes for each blade. Therefore, the model used for simulations contains 10 DOFs. This is a higher fidelity model than the one used for synthesis, which contains 9 DOFs. The LPV controller and the baseline controller will be compared in 21 simulations with wind speeds varying from $5\,\mathrm{m/s}$ to $25\,\mathrm{m/s}$. As introduced in Section 2.3, TurbSim is used here to generate turbulent wind profiles. A turbulence level of $5\,\%$ is considered for all wind profiles. The time range for each simulation is $660\,\mathrm{s}$. However, results in the first $60\,\mathrm{s}$ will be ignored in the analysis as different initial responses for the two controllers might affect the comparison.

For simplicity, simulation results in 3 different wind speeds are shown in Figures 3.9 to 3.12, which cover Regions 2, 2.5 and 3 wind conditions. To get a better view of the data, plots are narrowed down to the time range from $200\,\mathrm{s}$ to $400\,\mathrm{s}$. Figure 3.9 shows simulation results at $\rho = 8\,\mathrm{m/s}$. It is seen in this figure that both the LPV controller and the baseline controller have good generator speed tracking performance to the varying wind speed for maximizing the power generation. Actuations of the generator torque are also similar for both two controllers. It is noted that the LPV controller used the blade pitch angle at about $340\,\mathrm{s}$. However, it does not affect too much to the power generation, as shown by the comparison of power generations for these 2 controllers in Figure 3.10. Similar comparison results can be concluded in other wind speeds of Region 2. This is to be expected based on the design strategy in Section 3.3 and the weights selection in Section 3.4.2.

Figure 3.11 shows simulation results at $\rho = 12\,\mathrm{m/s}$. Here, the LPV controller has better generator speed tracking performance than the baseline controller. However, detailed analysis of the results in Figure 3.10 shows that it loses about $12\,\%$ of the power generation comparing to the baseline controller. This is due to imbalanced actuations for the generator torque and blade pitch. As shown in subplots of Figure 3.11 for these two control inputs, the LPV controller attempted to use more blade pitch than generator torque for generator speed tracking, which is more similar to the operation for Region 3. Therefore, more
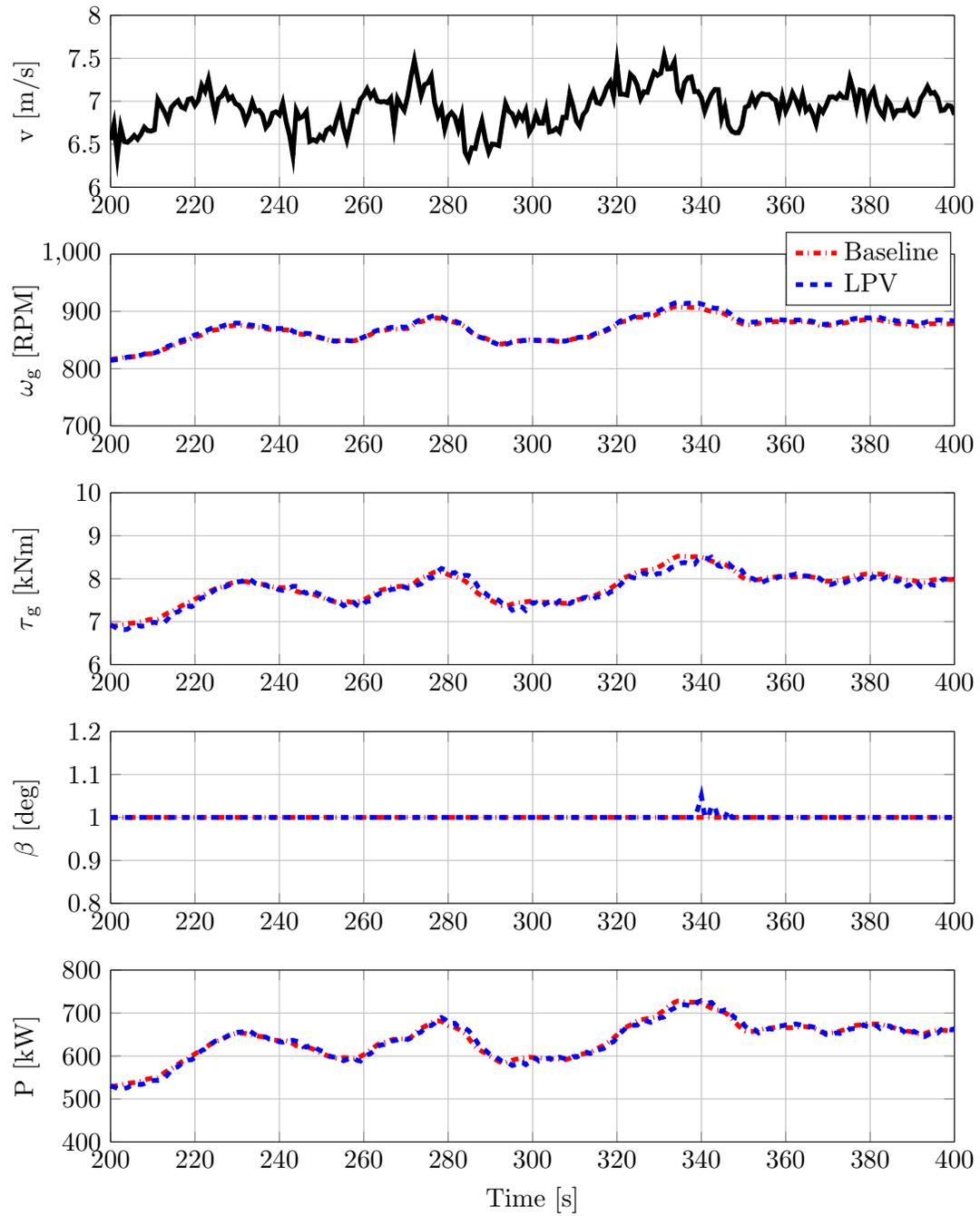
Figure 3.9: Simulations for the baseline controller and the LPV controller at $8\,\mathrm{m/s}$.
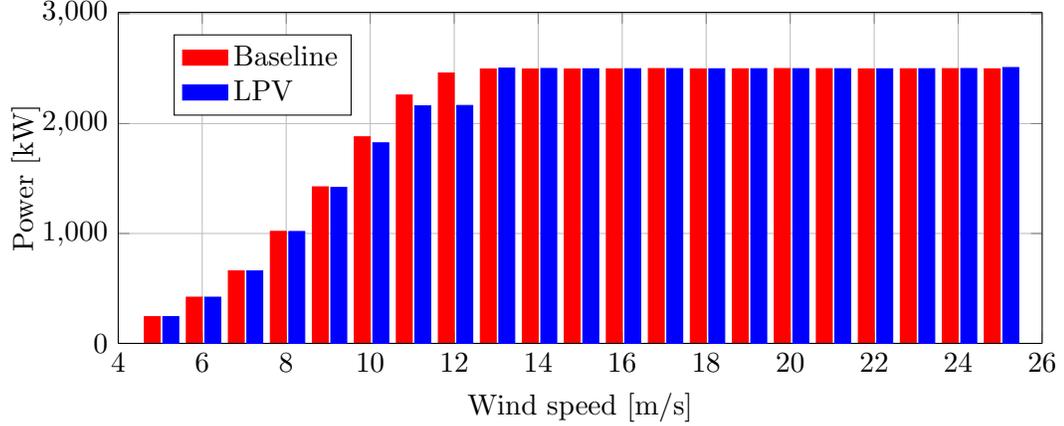
Figure 3.10: Comparison of power generations for the baseline controller and the LPV controller in different wine speeds.

power was lost for the LPV controller. This drawback should be improved in the future by re-evaluating the trade-off between two actuations in Region 2.5. It is also noted that the baseline controller has a large drop in the generator torque and power output near 280 s and 355 s. This is because the baseline controller tries to switch control loops from Region 3 to Regions 2.5 due to the wind speed drops. However, this turbulence is lower pass filtered by the LPV controller and it still operates in above rated wind speed.

Simulation results for the trim wind speed of 18 m/s are shown in Figure 3.12, which correspond to the operation in Region 3. It is clear that the LPV controller has much better performance on the generator speed tracking than the baseline controller. To quantitatively compare the tracking performance, define root mean square (RMS) of the generator speed tracking error ω_{g_RMS} as:

$$\omega_{g\_RMS} = \left(\frac{1}{T} \int_0^T |\omega_g - \omega_{g\_rated}|^2 dt \right)^{\frac{1}{2}} \tag{3.12}$$

The comparison results for these 2 controllers in different wind speeds of Region 3 are shown in Figure 3.13, which fit the observation in Figure 3.12. The variation on the power generation is also less for the LPV controller. The price for this desired tracking performance is an extra actuation of blade pitch in higher frequencies. However, the resulting damage to blade pitch actuators should be acceptable as W_β has been tuned with a bandwidth lower than the limit of actuators. In addition, as what will be shown in Section 3.5.2, this actuation does not introduce more loads on blades. It should be noted that the baseline controller tries to decrease the generator torque for maintaining the rated power generation when the generator speed increases too much with the wind speed. This phenomenon happens at around 310 s and 330 s. While this actuation of the generator torque might be

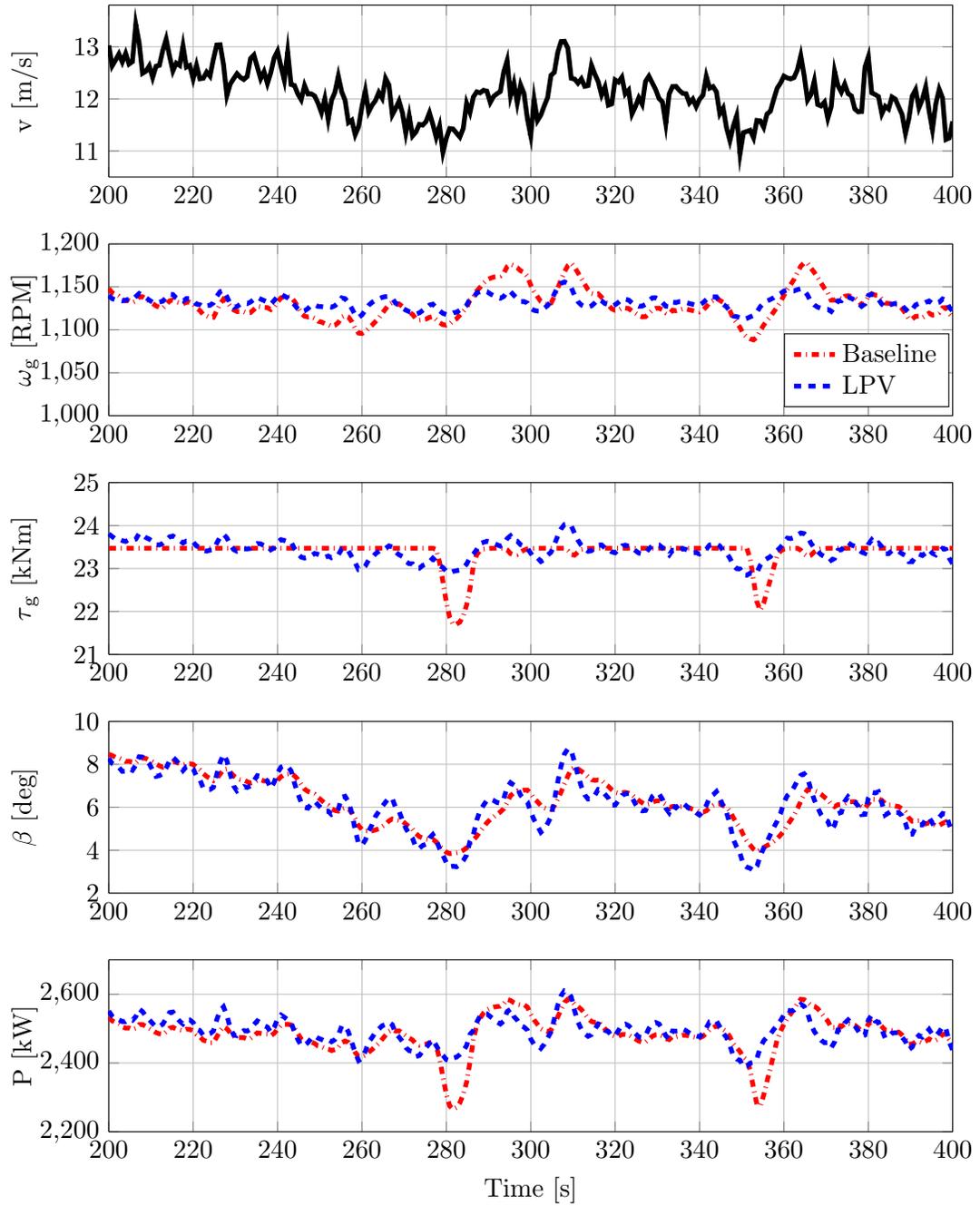Figure 3.11: Simulations for the baseline controller and the LPV controller at $12\,\mathrm{m/s}$.

helpful to decrease variations of the power generation, it leads to the potential of instability of the rotor dynamics as shown in Equation 2.2. In contrast, the LPV controller guarantees the stability and provides better load reductions by moderately varying the generator torque around the rated value. The range for this variation can be adjusted by tuning the weight W_τ, according to constraints on the generator.

3.5.2 Damage Equivalent Load Analysis

This section provides the post analysis of simulation results on loads for a further comparison between the two controllers. The metric for evaluating loads on the turbine is based on the calculation of damage equivalent loads (DELs). The DELs show the equivalent fatigue damage caused by the load on a specific component. The DELs can be calculated using the MCrunch [73] which is also developed by NREL. Results of the DELs analysis for the two controllers are shown in Figure 3.14 based on the data in 21 simulations as mentioned in Section 3.5.1. Here, 5 different loads are considered which include the tower base bending moments in fore-aft and side-to-side directions M_{tfa} and M_{tss}, high speed shaft torque τ_{hss} and averaged blade flapwise and edgewise bending moments M_{bfw} and M_{bss}.

As shown in Figure 3.14, all loads generally increase with the wind speed, except for the blade edgewise bending moment. The detailed check on this load shows that the vibration of blades in the edgewise direction is dominated by the periodic motion of the rotation. Therefore, collective pitch control has few effects on this load. It is also noted that the high speed shaft torque reaches its maximum in Region 2.5. This makes sense as the turbine still uses the generator torque as the main control input for power maximization in Region 2.5, where the wind speed is sufficient high to introduce large vibrations on the generator torque.

Improvements for the LPV controller on load reduction are shown in Figure 3.15 by comparing to the baseline controller on DELs. It is seen in this figure that the load reduction effect is not obvious in Region 2 for the LPV controller. This is because load reduction is less critical in Region 2 and less emphasis has been put in the corresponding weight of wind turbulence $W_v^{(1)}$. In Region 2.5, the LPV controller starts to show better load reduction results than the baseline controller. However, there is a trade off between the blade flapwise bending moment and the high speed shaft torque. As the high speed shaft torque is similar to the generator torque, considering the lightly damped drive train torsion mode in the high frequency, this observation reflects the balance of two actuations in Region 2.5. In Region 3, the LPV controller has much less DELs than the baseline controller. Specifically, there is about 20 % decrease of the DEL on the high speed shaft torque. The improvement for the blade flapwise bending moment is about 10 %. The load reductions on the tower fore-aft
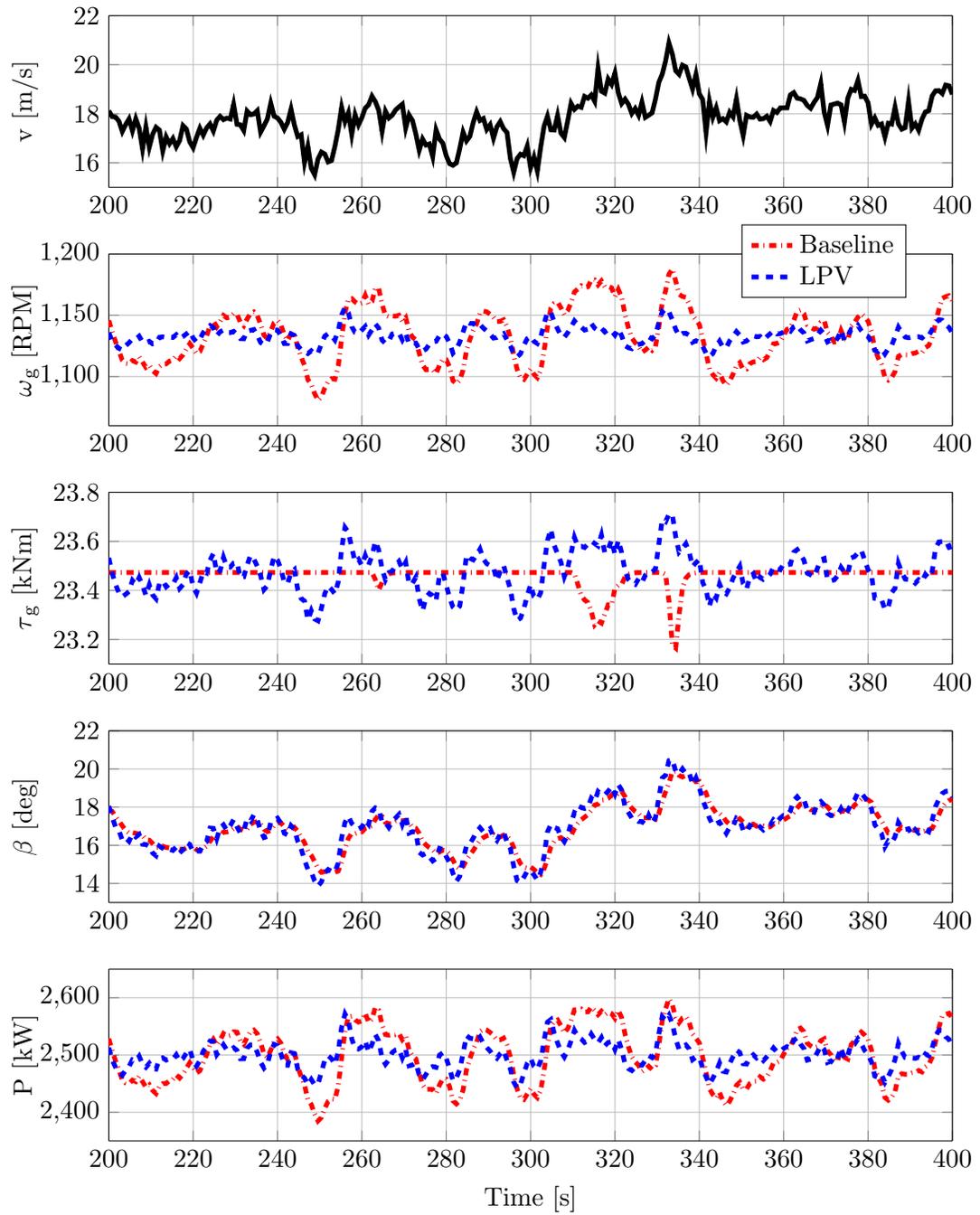
Figure 3.12: Simulations for the baseline controller and the LPV controller at $18\,\text{m/s}$.
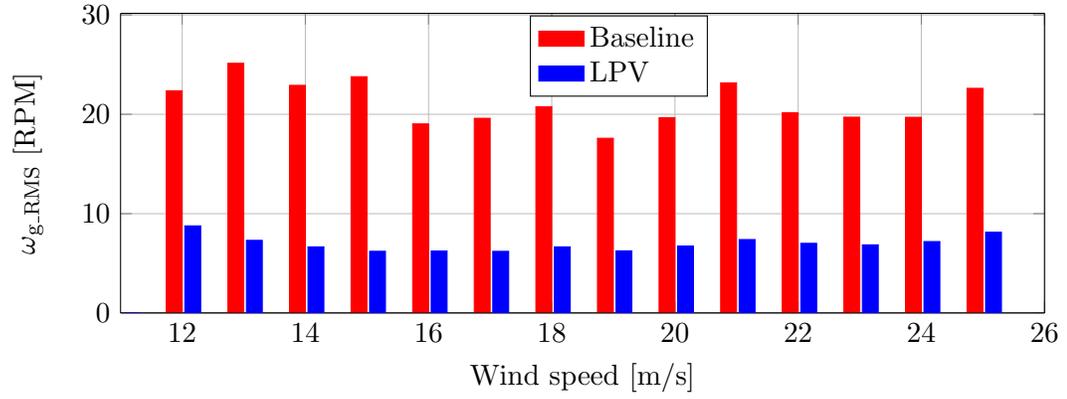
Figure 3.13: Comparison of RMS of the generator speed tracking error for the baseline controller and the LPV controller in Region 3.

and side-to-side bending moments are about 25 % and 50 % respectively.
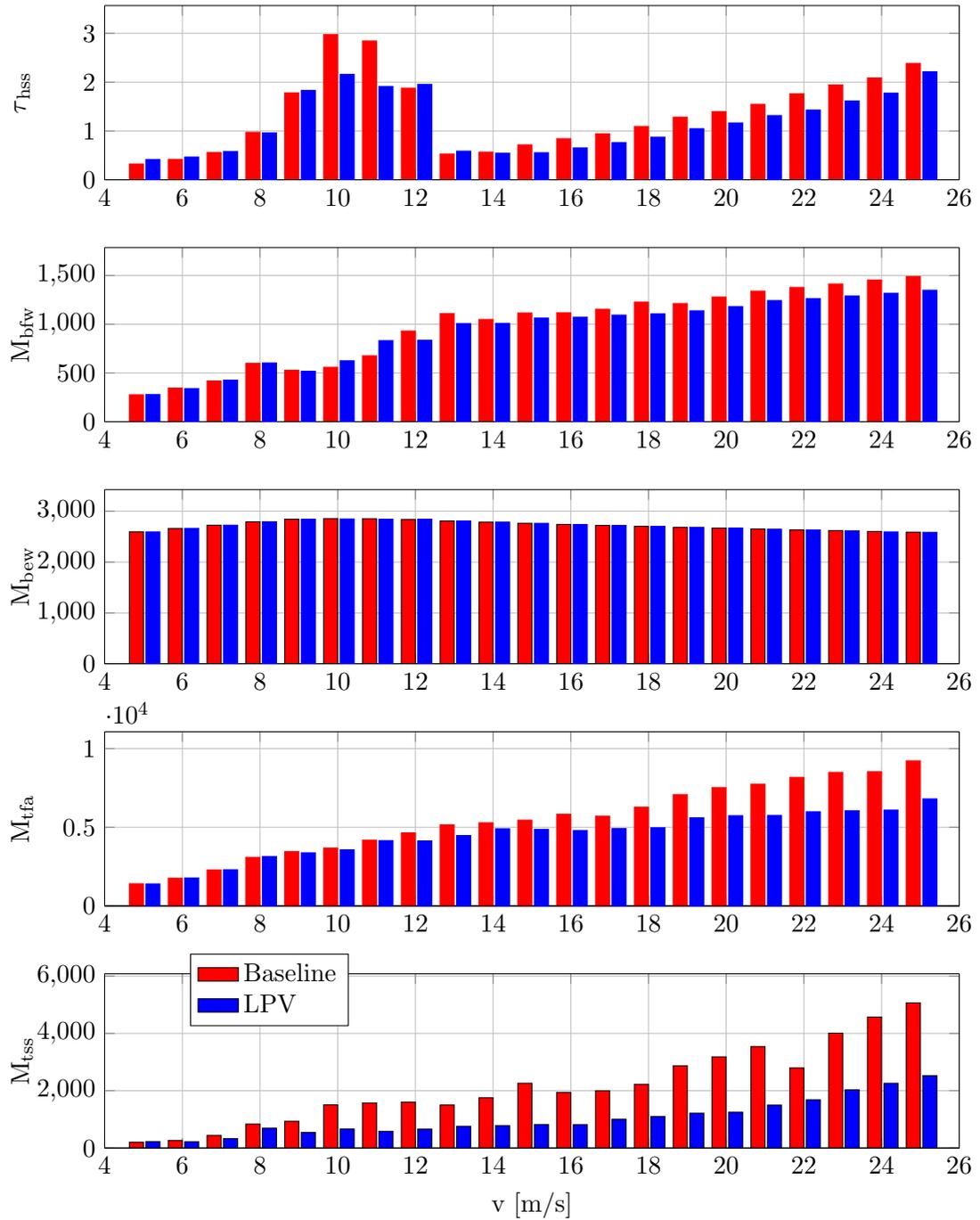
Figure 3.14: Damage equivalent loads (DELs) for the baseline controller and the LPV controller at different wind speeds.
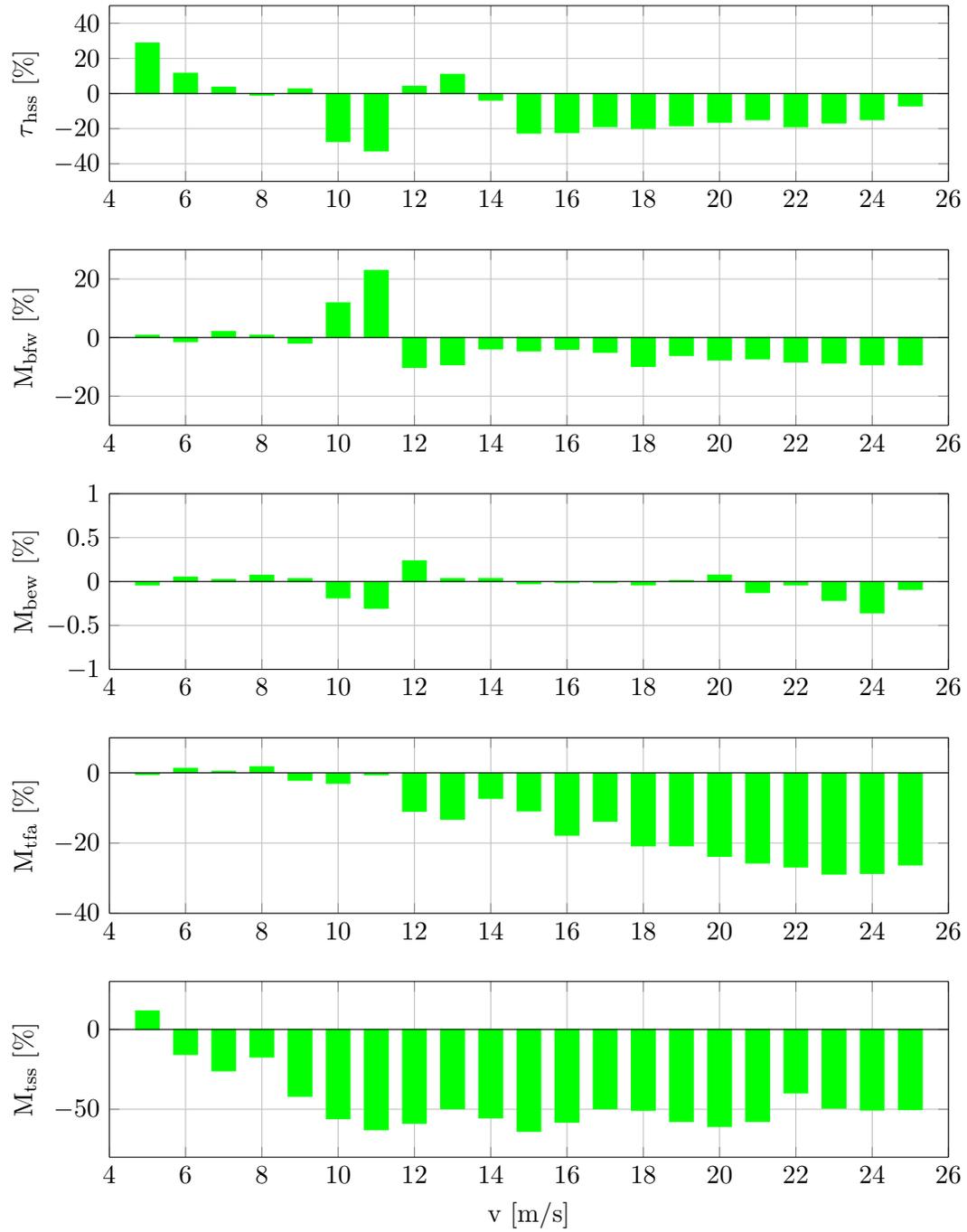
Figure 3.15: Improvements for the LPV controller on load reduction at different wind speeds.

Chapter 4

Robust Synthesis for LPV Systems Using IQCs

4.1 Introduction

This chapter considers the robust synthesis problem for a class of uncertain linear parameter varying (LPV) systems. The uncertain system is described as an interconnection of a nominal (not-uncertain) LPV system and a block structured uncertainty. The state matrices of the nominal system are assumed to have an arbitrary dependence on parameters, i.e. the nominal part is a "gridded" LPV system. Such models arise naturally in many applications via linearization of a nonlinear model around parameterized operating (trim) points. Specific examples of current interest include aeroelastic vehicles [74] and wind turbines [63,75]. The input/output behavior of the uncertainty is described by integral quadratic constraints (IQCs) [19]. The use of IQCs is sufficiently general to describe "uncertain" components that include nonlinearities, e.g. saturation, in addition to (parametric or dynamic) uncertainty.

The robust synthesis problem, formulated in Section 4.3.1, involves a search for a controller that minimizes a closed-loop robust performance metric. This leads to a non-convex optimization that involves a search for both the controller state matrices and the IQC analysis variables. The proposed algorithm, given in Section 4.3.3, consists of a coordinate-wise descent similar to the well-known DK-iteration [20,76] for μ synthesis. Specifically, the proposed algorithm alternates between an LPV synthesis step and an IQC analysis step. The synthesis step essentially relies on existing results for nominal (not) uncertain "gridded" LPV systems in [57,58]. The analysis step is performed using a matrix inequality condition to bound the robust performance of the closed-loop uncertain LPV system (Section 4.4.1). Both steps can be efficiently solved as semidefinite programs (SDPs). The effectiveness of

the proposed method is demonstrated on a simple numerical example in Section 4.5.

There are two main technical challenges in developing this algorithm. First, the nominal LPV system does not have a valid frequency response interpretation and hence the robustness analysis requires a time domain approach. Section 4.4.1 develops a matrix inequality robustness analysis condition (Theorem 3) using (time domain) dissipation inequality techniques. This result requires several technical lemmas to convert a conic combination of many frequency domain IQCs into a single, equivalent time domain IQC. This analysis condition is an extension of the worst-case gain condition in [77,78]. We note that there are alternative robust stability conditions for time-varying systems based on the ν-gap metric rather than dissipation inequalities [79]. These alternative robust stability conditions can potentially be used to develop synthesis algorithms complementary to those developed here. The second technical challenge is that an appropriate scaled system must be constructed to link the analysis and synthesis steps. In particular, the single equivalent time domain IQC from the analysis step must be combined with the nominal open-loop system to create the scaled system. This construction, described in Section 4.4.2, is such that the next synthesis step on the scaled plant yields a controller that improves the closed-loop robust performance. These technical results are used to show the following main result in Section 4.4.3: the robust performance metric is non-increasing at each iteration step and hence the algorithm converges.

This chapter builds on many known results for both LPV systems and IQCs. A brief review of these existing results is provided in Section 4.2. In addition, there are several related robust synthesis results for LPV systems [68,80–83]. These existing robust synthesis results are for the case where the state matrices of the nominal LPV system have a rational dependence on the scheduling parameters. This rational (linear fractional) dependence on the parameters is exploited in the algorithm development and leads to finite-dimensional matrix inequalities for both the synthesis and analysis steps. In contrast, the algorithm in this chapter is developed for the case where the state matrices of the nominal LPV system have an arbitrary dependence on the parameters. As noted above, this enables applications to systems, e.g. aeroelastic aircraft or wind turbines, for which arbitrary dependence on scheduling parameters is a natural modeling framework. The drawback of this approach is that it leads to parameter-dependent matrix inequalities for both the synthesis and analysis steps. As a result, parameter gridding is required to obtained finite-dimensional matrix inequality conditions. Finally, this chapter builds on a related conference paper submission [84]. The conference paper only considered LTI uncertainty while this chapter considers (possibly nonlinear) components whose input/output behavior are described by a general class of dynamic IQCs.

4.2 Integral Quadratic Constraints

Integral quadratic constraints (IQCs) [19] provide a framework for robustness analysis building on work by Yakubovich [85]. The IQC specifies a constraint on the input-output signals of the perturbation. The form of the constraint is such that it can be easily incorporated into tractable stability and performance analysis conditions. The following definitions characterize the constraint in the frequency and time domain.

Definition 1. *Let $\Pi \in \mathbb{RL}_\infty^{(n_v+n_w)\times(n_v+n_w)}$ be a rational and uniformly bounded function of $j\omega$. Two signals $v \in L_2^{n_v}[0,\infty)$ and $w \in L_2^{n_w}[0,\infty)$ satisfy the <u>frequency domain IQC</u> defined by the multiplier Π if*

$$\int_{-\infty}^{\infty} \left[\begin{array}{c} \hat{V}(j\omega) \\ \hat{W}(j\omega) \end{array} \right]^* \Pi(j\omega) \left[\begin{array}{c} \hat{V}(j\omega) \\ \hat{W}(j\omega) \end{array} \right] d\omega \geq 0 \tag{4.1}$$

where $\hat{V}$ and $\hat{W}$ are Fourier transforms of v and w. A bounded, causal operator $\Delta : L_{2e}^{n_v}[0,\infty) \to L_{2e}^{n_w}[0,\infty)$ satisfies the frequency domain IQC defined by Π if Equation 4.1 holds for all $v \in L_2^{n_v}[0,\infty)$ and $w = \Delta(v)$.

Definition 2. *Let Ψ be a stable LTI system, i.e. $\Psi \in \mathbb{RH}_\infty^{n_z \times (n_v+n_w)}$, and $M = M^T \in \mathbb{R}^{n_z \times n_z}$. Two signals $v \in L_{2e}^{n_v}[0,\infty)$ and $w \in L_{2e}^{n_w}[0,\infty)$ satisfy the <u>time domain IQC</u> defined by the multiplier Ψ and matrix M if the following inequality holds for all $T \geq 0$*

$$\int_0^T z^T(t) M z(t)\, dt \geq 0 \tag{4.2}$$

where z is the output of Ψ driven by inputs (v,w) with zero initial conditions. A bounded, causal operator $\Delta : L_{2e}^{n_v}[0,\infty) \to L_{2e}^{n_w}[0,\infty)$ satisfies the time domain IQC defined by (Ψ, M) if Inequality 4.2 holds for all $v \in L_{2e}^{n_v}[0,\infty)$, $w = \Delta(v)$ and $T \geq 0$.

IQCs can be used to model a variety of nonlinearities and uncertainties. In particular, [19] provides a library of frequency domain IQC multipliers that are satisfied by many important system components, e.g. saturation, time delay, and norm bounded uncertainty. Figure 4.1 provides a graphical interpretation for the time domain IQC. The input and output signals of Δ are filtered through Ψ. If Δ satisfies the time domain IQC defined by Ψ then the filtered signal z satisfies the constraint in Equation 4.2 for any finite-horizon $T \geq 0$.

A precise connection between the frequency and time domain IQC formulations is important for the robust synthesis algorithm described in this chapter. Assume Δ satisfies the time domain IQC defined by (Ψ, M). Taking $T \to \infty$ in Equation 4.2 yields
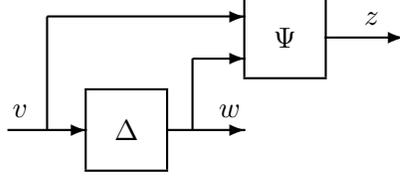
Figure 4.1: Graphical interpretation of the IQC.

$\int_0^\infty z(t)^T M z(t)\,dt \geq 0$. By Parseval's theorem [20], this is equivalent to the frequency domain constraint $\int_{-\infty}^\infty \hat{Z}(j\omega)^* M \hat{Z}(j\omega)\,d\omega \geq 0$ where $\hat{Z}(j\omega) = \Psi(j\omega) \begin{bmatrix} \hat{V}(j\omega) \\ \hat{W}(j\omega) \end{bmatrix}$. Thus if Δ satisfies the time domain IQC defined by (Ψ, M) then it satisfies the frequency domain IQC defined by $\Pi = \Psi^\sim M \Psi$.

The reverse implication is more technical and fails to hold in general. Specifically, assume Δ satisfies the frequency domain IQC defined by the multiplier Π. Any rational multiplier Π can be factorized as $\Pi = \Psi^\sim M \Psi$ where $\Psi \in \mathbb{RH}_\infty^{n_z \times (n_v + n_w)}$ is stable and $M = M^T \in \mathbb{R}^{n_z \times n_z}$. Such factorizations are not unique but can be computed using state-space calculations [86–88]. One specific numerical construction is given by Lemma 4 in Appendix A. Substitute the factorization for Π into the frequency domain IQC (Equation 4.1) and apply Parseval's theorem [20] to convert to a time domain constraint. This yields $\int_0^\infty z(t)^T M z(t)\,dt \geq 0$ where z is the output of Ψ driven by v and $w = \Delta(v)$ with zero initial conditions. This time domain constraint holds, in general, only over infinite horizons and only for finite-norm input signals $v \in L_2^{n_v}[0, \infty)$. However, the time domain IQC (Definition 2) requires the integral inequality to hold over all finite times $T \geq 0$ and for all extended-space input signals $v \in L_{2e}^{n_v}[0, \infty)$. A time domain IQC as in Definition 2 is referred to as a hard IQC in [19]. In contrast, factorizations for which the time domain constraint holds only for $T = \infty$ are called soft IQCs. This distinction is important because the dissipation theorems specified later for robustness analysis require the use of hard IQCs.[1] Lemmas 5 and 6 in Appendix A provide a specific "hard" factorization (Ψ, M) that can be constructed under additional assumptions on the frequency domain multiplier Π. To summarize these lemmas, let $\Pi = \Pi^\sim \in \mathbb{RL}_\infty^{(n_v + n_w) \times (n_v + n_w)}$ be partitioned as $\begin{bmatrix} \Pi_{11} & \Pi_{12} \\ \Pi_{12}^\sim & \Pi_{22} \end{bmatrix}$ where $\Pi_{11} \in \mathbb{RL}_\infty^{n_v \times n_v}$ and $\Pi_{22} \in \mathbb{RL}_\infty^{n_w \times n_w}$. If $\Pi_{11}(j\omega) > 0$ and $\Pi_{22}(j\omega) < 0$ $\forall \omega \in \mathbb{R} \cup \{\infty\}$, then Π has a hard factorization (Ψ, M) that yields a time domain IQC (Definition 2). (Ψ, M) can be constructed from the stabilizing solution to an Algebraic Riccati Equation (ARE) and is called a J-spectral factorization of Π.

LPV systems do not have a valid frequency response interpretation. Hence existing conditions for robust analysis of gridded LPV systems [77,78] rely on the use of valid time domain

---

[1]The terms "complete" and "conditional" IQCs in [89] are generalizations of hard and soft IQCs. The hard/soft terminology will be used here.

(hard) IQCs. Section 4.4.1 generalizes these existing results to handle factorizations (Ψ, M) that are not necessarily hard. Moreover, analysis conditions for a worst-case gain metric are provided in [77, 78]. Here, analysis conditions are derived for a robust performance metric. Definitions of these two metrics and their difference will be introduced in Section 4.3.1.

4.3 Robust Synthesis Algorithm

4.3.1 Problem Formulation

Consider the robust synthesis problem for the uncertain LPV system as shown in Figure 4.2. The uncertain LPV system is described by the interconnection of an open loop LPV system G_ρ, a perturbation Δ, and an LPV controller K_ρ. A state-space realization for G_ρ is given by:

$$
\begin{bmatrix} \dot{x}_G \\ v \\ e \\ y \end{bmatrix} = \begin{bmatrix} A(\rho) & B_w(\rho) & B_d(\rho) & B_u(\rho) \\ C_v(\rho) & D_{vw}(\rho) & D_{vd}(\rho) & D_{vu}(\rho) \\ C_e(\rho) & D_{ew}(\rho) & D_{ed}(\rho) & D_{eu}(\rho) \\ C_y(\rho) & D_{yw}(\rho) & D_{yd}(\rho) & D_{yu}(\rho) \end{bmatrix} \begin{bmatrix} x_G \\ w \\ d \\ u \end{bmatrix}
\tag{4.3}
$$

where $x_G \in \mathbb{R}^{n_G}$, $w \in \mathbb{R}^{n_w}$, $d \in \mathbb{R}^{n_d}$, $u \in \mathbb{R}^{n_u}$, $v \in \mathbb{R}^{n_v}$, $e \in \mathbb{R}^{n_e}$ and $y \in \mathbb{R}^{n_y}$. The following assumptions are made regarding G_ρ and Δ:

Assumption 1. *G_ρ is quadratically stabilizable from u and quadratically detectable from y as defined in Chapter 1 of [57].*

Assumption 2. *The perturbation is a bounded, causal operator $\Delta : L_{2e}^{n_v}[0, \infty) \to L_{2e}^{n_w}[0, \infty)$ that satisfies a collection of IQCs defined by $\{\Pi_k\}_{k=1}^N \subset \mathbb{RL}_\infty^{(n_v+n_w) \times (n_v+n_w)}$ in the frequency domain.*

Assumption 3. *Partition the frequency domain multipliers $\{\Pi_k\}_{k=1}^N$ as $\begin{bmatrix} \Pi_{k,11} & \Pi_{k,12} \\ \Pi_{k,12}^\sim & \Pi_{k,22} \end{bmatrix}$ where $\Pi_{k,11}$ is $n_v \times n_v$. Each frequency domain multiplier satisfies $\Pi_{k,11}(j\omega) \geq 0$ and $\Pi_{k,22}(j\omega) \leq 0$ $\forall \omega \in \mathbb{R} \cup \{\infty\}$.*

Assumption 4. *The perturbation has been normalized to satisfy $\|\Delta\| \leq 1$ and the first IQC is defined by the multiplier $\Pi_1 := \begin{bmatrix} I_{n_v} & 0 \\ 0 & -I_{n_w} \end{bmatrix}$.*

The first assumption ensures that there is a controller K_ρ from y to u that stabilizes the (nominal) open loop interconnection of G_ρ and K_ρ. This open loop interconnected system is a lower LFT, denoted $\mathcal{F}_l(G_\rho, K_\rho)$. The IQCs in the second assumption are used to bound the input-output behavior of the perturbation Δ. This formulation can handle systems where Δ has block diagonal structure including static nonlinearities (e.g. saturations) and
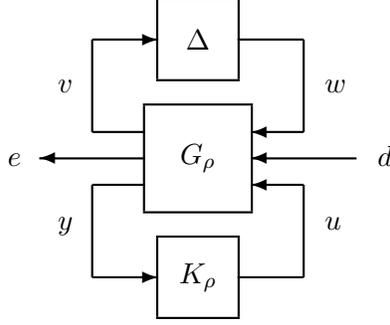
Figure 4.2: Interconnection for LPV robust synthesis.

infinite dimensional operators (e.g. time delays) in addition to true system uncertainties. The term "uncertainty" is used for simplicity when referring to the perturbation Δ. The notation $\boldsymbol{\Delta}(\Pi_1, \ldots, \Pi_N)$ will be used to denote the set of bounded, causal operators Δ that satisfy all frequency domain IQCs defined by $\{\Pi_k\}_{k=1}^N$.

The third and fourth assumptions are used to simplify the algorithm. Assumption 3 only requires the multipliers satisfy non-strict definiteness conditions $\Pi_{k,11} \geq 0$ and $\Pi_{k,22} \leq 0$. This is sufficiently general to cover most typical frequency domain multipliers used in IQC analysis. In fact, all frequency domain multipliers listed in [19] satisfy $\Pi_{k,11} \geq 0$ and $\Pi_{k,22} \leq 0$ except those for certain sector bounded nonlinearities and polytopic uncertainties which fail to contain the zero operator $\Delta = 0$. Finally, note that the individual multiplier Π_k need not satisfy the strict definiteness conditions $\Pi_{k,11} > 0$ and $\Pi_{k,22} < 0$ given in Lemma 5 for the existence of a J-spectral factorization. However, Assumptions 3 and 4 are sufficient to ensure that a "combined" multiplier that appears in the proposed robust synthesis algorithm satisfies the strict definiteness conditions and thus has a J-spectral factorization. Specifically, the "combined" multiplier described below will be formed (with an additional scaling neglected here) as: $\Pi_\lambda := \sum_{k=1}^N \lambda_k \Pi_k$. The coefficients will be constrained to satisfy $\lambda_1 > 0$ and $\lambda_k \geq 0$ for $k = 2, \ldots N$. Assumptions 3 and 4 along with these constraints on λ are sufficient to ensure that the combined multiplier satisfies the strict definiteness conditions $\Pi_{\lambda,11} > 0$ and $\Pi_{\lambda,22} < 0$ given in Lemma 5. Other parameterizations of the IQC multiplier are possible. For example, [90] uses the form $\Pi_\lambda := \Psi^* M(\lambda) \Psi$ where Ψ is a given stable filter (not necessarily square) and M is an affine matrix function of λ. The robust synthesis algorithm proposed here can be generalized to handle alternative parameterizations as long as the conditions $\Pi_{\lambda,11} > 0$ and $\Pi_{\lambda,22} < 0$ can be enforced as a convex constraint on λ.

To simplify notation, define $H_\rho := \mathcal{F}_l(G_\rho, K_\rho)$. The uncertain LPV system in Figure 4.2 can therefore be expressed as an upper LFT, denoted $\mathcal{F}_u(H_\rho, \Delta)$. A natural performance

metric for the uncertain LPV system is the worst-case gain:

$$\sup_{\Delta \in \mathbf{\Delta}(\Pi_1,\ldots,\Pi_N)} \|\mathcal{F}_u(H_\rho, \Delta)\| \tag{4.4}$$

This is the largest induced L_2 gain of the uncertain LPV system over all uncertainties consistent with the specified IQCs. This metric has been widely used for robustness analysis [77,78,91]. Note that the worst-case gain is only finite ($< +\infty$) if the controller K_ρ robustly stabilizes the plant over any uncertainty $\Delta \in \mathbf{\Delta}$. Hence the use of worst-case gain for robust synthesis requires an initial controller that achieves robust stability. It is possible to construct a two-stage synthesis algorithm that in the first stage attempts to design a robustly stabilizing controller and in the second stage attempts to minimize the worst-case gain. The first stage requires a way to measure the "robustness" of the controller relative to the uncertainty set $\mathbf{\Delta}$. A scaled uncertainty set can serve this purpose. Specifically, define S_b as the scaling matrix $\begin{bmatrix} bI_{n_v} & 0 \\ 0 & I_{n_w} \end{bmatrix}$. Let $\mathbf{\Delta}_b(\Pi_1, \ldots, \Pi_N)$ denote the set of bounded, causal operators Δ that satisfy the frequency domain IQCs defined by $S_b\Pi_k S_b$ for $k = 1, \ldots, N$. For the scaled set, if $b_2 \geq b_1$ then $\mathbf{\Delta}_{b_2} \supseteq \mathbf{\Delta}_{b_1}$.[2] This inclusion means that stability with respect to a scaled uncertainty set $\mathbf{\Delta}_b$ defined by b provides a useful robustness metric, i.e. larger values of b indicate more robustness. For many multipliers this scaling has a simple interpretation. For example, $\Pi := \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$ defines norm bounded uncertainty $\|\Delta\| \leq 1$. For this multiplier the scaled set $\mathbf{\Delta}_b(\Pi)$ corresponds to $S_b\Pi S_b = \begin{bmatrix} b^2 & 0 \\ 0 & -1 \end{bmatrix}$ and defines uncertainty $\|\Delta\| \leq b$. In some cases the interpretation is not as intuitive. For example, Zames-Falb multipliers can be used to describe an uncertainty set $\mathbf{\Delta}$ that contains monotonic nonlinearities such as the saturation. The scaled uncertainty set does not have an easy interpretation for such IQC multipliers. In these cases we can simply state that if a controller achieves robust stability with $b \geq 1$ then it achieves robust stability with respect to the unscaled set $\mathbf{\Delta}$.

The two-stage algorithm proposed above would use both a robust stability and worst-case gain metric. It is more convenient to combine these stages into a single algorithm. Specifically, it is standard, e.g. in DK synthesis, to instead use a robust performance metric that simultaneously scales both the uncertainty level and the system gain. This metric, formally defined below, is used for the robust synthesis algorithm in this chapter. The definition of robust performance uses the notion of a scaled uncertainty set discussed above.

Definition 3. *The system H_ρ achieves robust performance of level γ with respect to the*

---

[2]A sketch of the proof is given. Assume $\Pi_{11}(j\omega) \geq 0$ and $\Pi_{22}(j\omega) \leq 0$ $\forall \omega \in \mathbb{R} \cup \{\infty\}$. Let $\hat{V}$ and $\hat{W}$ be Fourier transforms of two signals in L_2. Define a function $f : \mathbb{R}_{>0} \to \mathbb{R}$ by $f(b) := \int_{-\infty}^{\infty} \begin{bmatrix} b\hat{V}(j\omega) \\ \hat{W}(j\omega) \end{bmatrix}^* \Pi(j\omega) \begin{bmatrix} b\hat{V}(j\omega) \\ \hat{W}(j\omega) \end{bmatrix} d\omega$. It can be shown that if $f(b_1) \geq 0$, then $f'(b) \geq 0$ $\forall b \geq b_1$. This can be used to show that if $(\hat{V}, \hat{W})$ satisfy the scaled IQC defined by $S_{b_1}\Pi S_{b_1}$ then $(\hat{V}, \hat{W})$ satisfy the scaled IQC for any $b_2 \geq b_1$.

uncertainty described by $\{\Pi_k\}_{k=1}^{N}$ *if*

$$\sup_{\Delta \in \mathbf{\Delta}_{1/\gamma}(\Pi_1,\dots,\Pi_N)} \|\mathcal{F}_u(H_\rho, \Delta)\| \leq \gamma \tag{4.5}$$

Let $r_{\mathbf{\Delta}(\Pi_1,\dots,\Pi_N)}[H_\rho]$ *denote the smallest level of robust performance achievable by* H_ρ.

H_ρ achieves robust performance of level γ if the worst-case induced L_2 gain from d to e is $\leq \gamma$ over all uncertainties in the scaled set $\mathbf{\Delta}_{1/\gamma}(\Pi_1,\dots,\Pi_N)$. For decreasing levels of robust performance, the gain decreases and the bound of the tolerable uncertainty increases. The robust performance level for the uncertain system with known nominal plant H_ρ can be analyzed using convex optimization as described in Section 4.4.1.

The objective of the robust synthesis problem is to synthesize an LPV controller K_ρ with the form in Equation 3.6 that stabilizes the open-loop model G_ρ and minimizes the closed-loop robust performance. Thus the synthesis problem is:

$$\inf_{K_\rho \text{ stabilizing}} r_{\mathbf{\Delta}(\mathbf{\Pi_1},\dots,\mathbf{\Pi_N})} [\mathcal{F}_l(G_\rho, K_\rho)] \tag{4.6}$$

It is typical to scale the performance weights to achieve a robust performance metric near 1. This ensures that the synthesized controller robustly stabilizes the modeled (unscaled) uncertainty.

4.3.2 DK Synthesis

This section briefly reviews the standard DK synthesis algorithm [20, 76]. The objective is to clarify the notation presented thus far and to provide a basis for comparison with the proposed algorithm. In DK synthesis the nominal plant G is LTI and the uncertainty Δ is LTI and unit norm bounded. The robust synthesis problem involves the search for an LTI controller K and robustness analysis scalings D (called D-scales). The problem is non-convex, in general, and DK synthesis employs a coordinate-wise iteration. Specifically, the algorithm iterates between a controller synthesis step (K-step) and a robustness analysis step (D-step). The synthesis step involves the design of an H_∞ controller K on a nominal (not-uncertain) scaled system. The analysis step involves the search for a frequency domain scaling D to assess the robust performance of the closed-loop $H := \mathcal{F}_l(G, K)$. The coordinate-wise iteration for DK synthesis does not, in general, converge to a local (nor global) optima. However it has the advantage that each of the decoupled synthesis and analysis steps is a convex optimization.

The main technical result for DK synthesis is that the iteration is well posed at each

step and the robust performance is (in theory) non-increasing. This result is based on the construction of a scaled system that links the analysis and synthesis steps. The scaled system used in the K-step is DGD^{-1} where D is the scaling from the analysis step. The main loop theorem [20] establishes the equivalence between robust performance of the (uncertain) closed-loop $\mathcal{F}_u(H, \Delta)$ and the induced L_2 performance of the (not-uncertain) scaled system $\mathcal{F}_l(DGD^{-1}, K)$. If the synthesis problem includes mixed (real and complex) uncertainty then the construction of an appropriate scaled system is more subtle. For example, the DGK synthesis algorithm [92, 93] uses a specific factorization of the D/G scalings to prove that the robust performance monotonically decreases. One of the major technical results given below leads to the construction of an appropriate scaled system for the robust LPV synthesis with general IQCs.

Finally, we briefly connect the notation used in standard DK synthesis with that introduced here for the robust LPV synthesis problem. The uncertainty in DK synthesis is, in general, block structured but for simplicity this discussion assumes Δ is SISO (no structure). If $\|\Delta\|_\infty \leq 1$ then Δ satisfies the frequency domain IQC defined by $\Pi = \begin{bmatrix} \alpha & 0 \\ 0 & -\alpha \end{bmatrix}$ for any SISO, LTI system α such that $\alpha(j\omega) = \alpha(j\omega)^* > 0 \ \forall \omega \in \mathbb{R} \cup \{\infty\}$. Moreover, if $\|\Delta\|_\infty \leq \frac{1}{\gamma}$ then Δ satisfies the frequency domain IQC defined by the scaled multiplier $S_{1/\gamma} \Pi S_{1/\gamma}$. The condition $\alpha > 0$ ensures that α has a spectral factorization $\alpha = d^\sim d$ where d is the scaling/multiplier that appears in DK synthesis. In this case, $\Psi = \begin{bmatrix} d & 0 \\ 0 & d \end{bmatrix}$ and $M = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$ defines a J-spectral factorization of Π. The D-step in DK synthesis is typically implemented by solving for D-scales on a frequency grid and then fitting the result with a rational transfer function. Here, the scalings will be restricted to a finite, linear combination of user selected basis functions. In particular, the definition of robust performance (Definition 3) requires a finite number N of (fixed) multipliers $\{\Pi_k\}$ to be specified. In the context of this DK synthesis example, this corresponds to the selection of N scalings $\{\alpha_k\}$. The proposed algorithm given below will search for the best linear combination of these scalings.

4.3.3 Algorithm Description

This section gives a high-level overview of the proposed LPV robust synthesis algorithm. Technical details regarding the algorithm are then given in Section 4.4. As in DK synthesis, the robust LPV synthesis is, in general, non-convex. In particular, Theorem 3 in Section 4.4.1 provides a linear matrix inequality (LMI) formulation for robust performance. Applying this result for synthesis leads to a matrix inequality condition that is bilinear in the state matrices for the controller K_ρ and the analysis variables consisting of a storage matrix $P \geq 0$ and IQC coefficients $\{\lambda_k\}_{k=1}^N$. A standard coordinate-wise approach is used to decouple the design into a nominal controller synthesis step (for K_ρ) and a robust performance analysis step (for P and λ). The technical results in Section 4.4 are used to link

these steps.

The detailed steps of the algorithm including the initialization and termination conditions are described in Algorithm 1. This algorithm is briefly described to provide a roadmap for the technical results in the following section. The algorithm initialization (Step 2) computes a factorization for each IQC multiplier. Any stable factorization of the Π_k may be used in and the construction in Lemma 4 of Appendix A is just one possibility. As noted above, a J-spectral factorization need not exist for the individual multipliers Π_k and hence the factorization need not be "hard". The main steps of the algorithm involve a synthesis step (Step 8), analysis step (Step 9), and the construction of a scaled-system G_ρ^{scl} that links these steps (Steps 5-7). The synthesis ste is a standard (nominal) LPV synthesis on the scaled system. It uses the algorithm in [57,58] and summarized by Theorem 2 in Section 3.2. The analysis step is a parameterized matrix inequality condition (Theorem 3 in Section 4.4.1) that involves a storage function matrix P, analysis vector λ, and robust performance bound γ. The bound γ enters bilinearly in the matrix inequality and hence this step requires bisection to find the minimum feasible value of γ. This step can be interpreted as a search over linear combinations of the scaled IQCs to form a single combined IQC multiplier (Step 6). The technical condition $\lambda_1(i) > 0$ in Step 9 is used to ensure that the combined IQC multiplier in Step 6 has a J-spectral factorization. In particular, Assumption 3 along with $\lambda \in \mathbb{R}_{\geq 0}$ implies that Π_λ satisfies the non-strict conditions $\Pi_{\lambda,11} \geq 0$ and $\Pi_{\lambda,22} \leq 0$. Assumption 4 gives $\Pi_1 := \begin{bmatrix} I & 0 \\ 0 & -I \end{bmatrix}$ so that $\lambda_1(i) > 0$ ensures that the strict conditions $\Pi_{\lambda,11} > 0$ and $\Pi_{\lambda,22} < 0$ are satisfied. Hence the combined multiplier has a J-spectral factorization by Lemma 5. Finally, the analysis and synthesis steps are linked by the construction of a particular scaled system (Step 7). The construction of the scaled system is described further in Section 4.4.2. The algorithm can be easily modified to incorporate other termination criteria in Step 10, e.g. maximum number of iterations and/or relative stopping tolerances.

As noted above, the LPV robust synthesis problem inherits the non-convexity of DK synthesis. The proposed coordinate-wise iteration will not, in general, converge to a local (nor global) optima. However, it is a pragmatic approach that decouples the synthesis and analysis steps into convex optimizations. The main technical result (Theorem 4 in Section 4.4.3) is that the algorithm iteration is well posed at each step and the robust performance is non-increasing. This is similar to the convergent property of the DK synthesis.
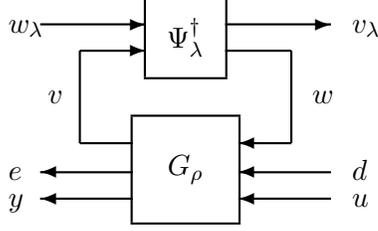
Figure 4.3: LFT interconnection of Scaled System, G_ρ^{scl}.

4.4 Technical Details

4.4.1 Robust Performance Condition

This section derives a matrix inequality condition to bound the robust performance for an uncertain LPV system. The uncertain LPV system is specified by the interconnection $\mathcal{F}_u(H_\rho, \Delta)$. The main technical issue is that the uncertainty Δ is described by IQCs $\{\Pi\}_{k=1}^N$ in the frequency domain but the nominal system H_ρ is LPV and does not have a valid frequency domain interpretation. The approach given here combines the frequency domain IQCs and converts it to a single, equivalent time domain IQC. The steps in this section alternate between various conditions involving the frequency domain IQCs and the single time domain IQC. This leads to the main technical result (Theorem 3) which involves a dissipation inequality characterization for robust performance.

The nominal LPV system H_ρ has the following state-space realization:

$$
\begin{bmatrix} \dot{x}_H \\ v \\ e \end{bmatrix} = \begin{bmatrix} A(\rho) & B_w(\rho) & B_d(\rho) \\ C_v(\rho) & D_{vw}(\rho) & D_{vd}(\rho) \\ C_e(\rho) & D_{ew}(\rho) & D_{ed}(\rho) \end{bmatrix} \begin{bmatrix} x_H \\ w \\ d \end{bmatrix}
\tag{4.7}
$$

where $x_H \in \mathbb{R}^{n_H}$, $w \in \mathbb{R}^{n_w}$, $d \in \mathbb{R}^{n_d}$, $v \in \mathbb{R}^{n_v}$ and $e \in \mathbb{R}^{n_e}$. The uncertainty Δ is assumed to satisfy multiple frequency domain IQCs defined by $\{\Pi_k\}_{k=1}^N$ under Assumptions 2, 3 and 4 in Section 4.3.1. Construct a factorization for each Π_k as (Ψ_k, M_k) where Ψ_k is stable, e.g. using the basic method described by Lemma 4 in Appendix A. The special J-spectral factorization is not required at this point. In fact, the individual Π_k need not satisfy the special sign-definiteness conditions specified in Lemma 5 for constructing a J-spectral factorization. Moreover, it should be emphasized that the factorization (Ψ_k, M_k) need not specify a valid time domain IQC as given by Definition 2.

The factorizations $\{(\Psi_k, M_k)\}_{k=1}^N$ are constructed for the multipliers $\{\Pi_k\}_{k=1}^N$ that define the normalized uncertainty set $\mathbf{\Delta}(\Pi_1, \ldots, \Pi_N)$. Recall that the definition of robust performance involves the scaled uncertainty set $\mathbf{\Delta}_{1/\gamma}(\Pi_1, \ldots, \Pi_N)$. This corresponds to the use of the scaled (frequency domain) multipliers $S_{1/\gamma} \Pi_k S_{1/\gamma}$ $(k = 1, \ldots, N)$. Thus a factorization

for each scaled multiplier is given by $(\Psi_k S_{1/\gamma}, M_k)$. Let z_k denote the output of the scaled system $\Psi_k S_{1/\gamma}$ driven by the input/output signals (v, w) of Δ assuming zero initial conditions. Then all $\{\Psi_k S_{1/\gamma}\}_{k=1}^N$ can be aggregated into a single system denoted $\Psi_{1/\gamma}$ with the following (minimal) state-space realization:

$$\begin{bmatrix} \dot{x}_\psi(t) \\ z_k(t) \end{bmatrix} = \begin{bmatrix} \tilde{A} & \gamma^{-1}\tilde{B}_v & \tilde{B}_w \\ \tilde{C}_{z_k} & \gamma^{-1}\tilde{D}_{z_k v} & \tilde{D}_{z_k w} \end{bmatrix} \begin{bmatrix} x_\psi(t) \\ v(t) \\ w(t) \end{bmatrix} \quad (k = 1, \ldots, N) \tag{4.8}$$

Equation 4.8 uses an abbreviated notation to denote that the outputs of $\Psi_{1/\gamma}$ are $[z_1^T, \ldots z_N^T]^T$. Note that the scaling matrix $S_{1/\gamma} := \begin{bmatrix} \gamma^{-1} I_{nv} & 0 \\ 0 & I_{nw} \end{bmatrix}$ only modifies the state matrices of $\Psi_{1/\gamma}$ associated with the v input, i.e. it only scales the $\tilde{B}_v$ and $\tilde{D}_{z_k v}$ matrices.

The robust performance analysis is based on the interconnection shown in Figure 4.4 with $\Delta \in \boldsymbol{\Delta}_{1/\gamma}(\Pi_1, \ldots, \Pi_N)$. The dynamics of this analysis interconnection are described by $w = \Delta(v)$ and the extended system of H_ρ and $\Psi_{1/\gamma}$:

$$\begin{bmatrix} \dot{x} \\ z_k \\ e \end{bmatrix} = \begin{bmatrix} \mathcal{A}(\rho) & \mathcal{B}_w(\rho) & \mathcal{B}_d(\rho) \\ \mathcal{C}_{z_k}(\rho) & \mathcal{D}_{z_k w}(\rho) & \mathcal{D}_{z_k d}(\rho) \\ \mathcal{C}_e(\rho) & \mathcal{D}_{ew}(\rho) & \mathcal{D}_{ed}(\rho) \end{bmatrix} \begin{bmatrix} x \\ w \\ d \end{bmatrix} \quad (k = 1, \ldots, N) \tag{4.9}$$

where the state vector is $x = [x_H; x_\psi] \in \mathbb{R}^{n_H + n_\psi}$ with x_H and x_ψ being the state vectors of the LPV system H_ρ and the filter $\Psi_{1/\gamma}$, respectively. The state matrices for the extended system can be expressed in terms of the state matrices for H_ρ (Equation 4.7) and $\Psi_{1/\gamma}$ (Equation 4.8). Appendix B provides one realization. Note that the state matrices of the extended system depend on the robust performance level γ. However this dependence on γ is not explicitly denoted. The uncertainty Δ is shown in the dashed box of Figure 4.4 to signify that the analysis condition given below is specified only in terms of the extended system of H_ρ and $\Psi_{1/\gamma}$. This effectively overbounds the precise relation $w = \Delta(v)$ with the IQCs satisfied by Δ.
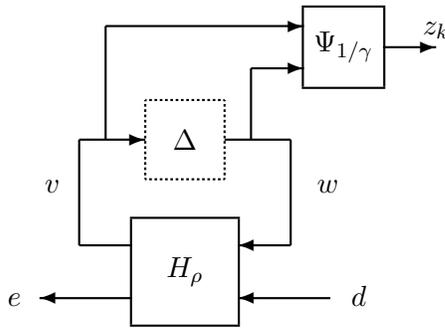

Figure 4.4: Uncertain LPV system extended to include filter $\Psi_{1/\gamma}$.

The robust performance analysis condition (given below) relies on a connection between $\Psi_{1/\gamma}$ and a combined multiplier $\Pi_\lambda := \sum_{k=1}^N \lambda_k S_{1/\gamma} \Pi_k S_{1/\gamma}$ defined by scalars $\{\lambda_k\}_{k=1}^N$ with conditions that $\lambda_1 \in \mathbb{R}_{>0}$ and $\lambda_k \in \mathbb{R}_{\geq 0}$ $(k = 2, \ldots, N)$. This combined multiplier can be expressed in terms of the state-space realization of $\Psi_{1/\gamma}$ (Equation 4.8) as:

$$\Pi_\lambda = \begin{bmatrix} (-sI - \tilde{A})^{-1}\tilde{B} \\ I \end{bmatrix}^T \begin{bmatrix} \tilde{Q}_\lambda & \tilde{S}_\lambda \\ \tilde{S}_\lambda^T & \tilde{R}_\lambda \end{bmatrix} \begin{bmatrix} (sI - \tilde{A})^{-1}\tilde{B} \\ I \end{bmatrix} \tag{4.10}$$

where

$$\tilde{B} := \begin{bmatrix} \gamma^{-1}\tilde{B}_v & \tilde{B}_w \end{bmatrix} \tag{4.11}$$

$$\begin{bmatrix} \tilde{Q}_\lambda & \tilde{S}_\lambda \\ \tilde{S}_\lambda^T & \tilde{R}_\lambda \end{bmatrix} := \sum_{k=1}^N \lambda_k \begin{bmatrix} \tilde{C}_{z_k}^T \\ \gamma^{-1}\tilde{D}_{z_k v}^T \\ \tilde{D}_{z_k w}^T \end{bmatrix} M_k \begin{bmatrix} \tilde{C}_{z_k} & \gamma^{-1}\tilde{D}_{z_k v} & \tilde{D}_{z_k w} \end{bmatrix} \tag{4.12}$$

Note that conditions on $\{\lambda_k\}_{k=1}^N$ along with Assumptions 3 and 4 imply that $(\Pi_\lambda)_{11}(j\omega) > 0$ and $(\Pi_\lambda)_{22}(j\omega) < 0 \ \forall \omega \in \mathbb{R} \cup \{+\infty\}$. Therefore Π_λ has a J-spectral factorization (Lemma 5). This factorization is constructed from the stabilizing solution X to the ARE in Equation A.2 with $(\tilde{A}, \tilde{B}, \tilde{Q}_\lambda, \tilde{S}_\lambda, \tilde{R}_\lambda)$. Without loss of generality, the J-spectral factorization can be rescaled as $(\Psi_\lambda, M_\lambda)$ where the constant matrix is $M_\lambda := \begin{bmatrix} \gamma^{-2}I & 0 \\ 0 & -I \end{bmatrix}$. Specifically, Let (Ψ, M) be a J-spectral factorization of Π_λ with $M := \begin{bmatrix} I & 0 \\ 0 & -I \end{bmatrix}$. Then $(\Psi_\lambda, M_\lambda) := (S_\gamma\Psi, S_{1/\gamma} M S_{1/\gamma})$ is another factorization of Π_λ with the constant matrix given by $S_{1/\gamma} M S_{1/\gamma} = \begin{bmatrix} \gamma^{-2}I & 0 \\ 0 & -I \end{bmatrix}$. In addition, the properties of a J-spectral factorization given in Lemma 6 carry over for this rescaled factorization. This rescaling will be important for the construction of the scaled plant in the synthesis step of our proposed algorithm (described in Section 4.4.2). The rescaled filter Ψ_λ only has one output and has a state-space realization of the form:

$$\begin{bmatrix} \dot{x}_\psi(t) \\ z_\lambda(t) \end{bmatrix} = \begin{bmatrix} \tilde{A} & \gamma^{-1}\tilde{B}_v & \tilde{B}_w \\ \tilde{C}_{z_\lambda} & \tilde{D}_{z_\lambda v} & \tilde{D}_{z_\lambda w} \end{bmatrix} \begin{bmatrix} x_\psi(t) \\ v(t) \\ w(t) \end{bmatrix} \tag{4.13}$$

This rescaled system Ψ_λ has the same state matrix $\tilde{A}$ and input matrix $[\gamma^{-1}\tilde{B}_v, \tilde{B}_w]$ as the original filter $\Psi_{1/\gamma}$. Only the output and feedthrough matrices of Ψ_λ are different from those in $\Psi_{1/\gamma}$. Finally, an extended system of H_ρ and Ψ_λ can be formed yielding:

$$\begin{bmatrix} \dot{x} \\ z_\lambda \\ e \end{bmatrix} = \begin{bmatrix} \mathcal{A}(\rho) & \mathcal{B}_w(\rho) & \mathcal{B}_d(\rho) \\ \mathcal{C}_{z_\lambda}(\rho) & \mathcal{D}_{z_\lambda w}(\rho) & \mathcal{D}_{z_\lambda d}(\rho) \\ \mathcal{C}_e(\rho) & \mathcal{D}_{ew}(\rho) & \mathcal{D}_{ed}(\rho) \end{bmatrix} \begin{bmatrix} x \\ w \\ d \end{bmatrix} \tag{4.14}$$

The state matrices for this extended system can be expressed in terms of the state matrices for H_ρ (Equation 4.7) and Ψ_λ (Equation 4.13). Only the output and feedthrough

matrices associated with z_λ in this alternative extended system differ from those given in Equation 4.9. Appendix B provides explicit formulae for $\mathcal{C}_{z_\lambda}$, $\mathcal{D}_{z_\lambda w}$, and $\mathcal{D}_{z_\lambda d}$. Again, the dependence on γ is not made explicit in this notation for the alternative extended system.

The robust performance condition relies on a technical lemma regarding matrix inequalities associated with the two extended systems presented thus far. Specifically, the extended system of H_ρ and $\Psi_{1/\gamma}$ (Equation 4.9) can be used to define the following parameterized matrix inequality involving multiple IQCs:

$$\begin{bmatrix} P\mathcal{A}+\mathcal{A}^T P & P\mathcal{B}_w & P\mathcal{B}_d \\ \mathcal{B}_w^T P & 0 & 0 \\ \mathcal{B}_d^T P & 0 & -I \end{bmatrix} + \sum_{k=1}^{N} \lambda_k \begin{bmatrix} \mathcal{C}_{z_k}^T \\ \mathcal{D}_{z_k w}^T \\ \mathcal{D}_{z_k d}^T \end{bmatrix} M_k \begin{bmatrix} \mathcal{C}_{z_k} & \mathcal{D}_{z_k w} & \mathcal{D}_{z_k d} \end{bmatrix} + \frac{1}{\gamma^2} \begin{bmatrix} \mathcal{C}_e^T \\ \mathcal{D}_{ew}^T \\ \mathcal{D}_{ed}^T \end{bmatrix} \begin{bmatrix} \mathcal{C}_e & \mathcal{D}_{ew} & \mathcal{D}_{ed} \end{bmatrix} < 0$$

$$(4.15)$$

This matrix inequality is parameterized by $\rho \in \mathcal{P}$ through the dependence of the extended system state matrices on the parameter. Similarly, the extended system of H_ρ and Ψ_λ (Equation 4.14) can be used to define the following parameterized matrix inequality involving the single, rescaled J-spectral factorization:

$$\begin{bmatrix} \tilde{P}\mathcal{A}+\mathcal{A}^T \tilde{P} & \tilde{P}\mathcal{B}_w & \tilde{P}\mathcal{B}_d \\ \mathcal{B}_w^T \tilde{P} & 0 & 0 \\ \mathcal{B}_d^T \tilde{P} & 0 & -I \end{bmatrix} + \begin{bmatrix} \mathcal{C}_{z_\lambda}^T \\ \mathcal{D}_{z_\lambda w}^T \\ \mathcal{D}_{z_\lambda d}^T \end{bmatrix} M_\lambda \begin{bmatrix} \mathcal{C}_{z_\lambda} & \mathcal{D}_{z_\lambda w} & \mathcal{D}_{z_\lambda d} \end{bmatrix} + \frac{1}{\gamma^2} \begin{bmatrix} \mathcal{C}_e^T \\ \mathcal{D}_{ew}^T \\ \mathcal{D}_{ed}^T \end{bmatrix} \begin{bmatrix} \mathcal{C}_e & \mathcal{D}_{ew} & \mathcal{D}_{ed} \end{bmatrix} < 0 \quad (4.16)$$

The technical result regarding these two matrix inequalities is formally stated in the next Lemma.

Lemma 1. *Let $\{\Pi_k\}_{k=1}^{N} \subset \mathbb{RL}_\infty^{(n_v+n_w)\times(n_v+n_w)}$, $\gamma > 0$, and $\{\lambda_k\}_{k=1}^{N}$ be givenwhere $\{\Pi_k\}_{k=1}^{N}$ satisfies Assumptions 3 and 4, $\lambda_1 \in \mathbb{R}_{>0}$ and $\lambda_k \in \mathbb{R}_{\geq 0}$ ($k = 2,\ldots,N$). Let each Π_k have a factorization (Ψ_k, M_k) where Ψ_k is stable. Define $\Pi_\lambda := \sum_{k=1}^{N} \lambda_k S_{1/\gamma}\Pi_k S_{1/\gamma}$. Thus $(\Pi_\lambda)_{11}(j\omega) > 0$ and $(\Pi_\lambda)_{22}(j\omega) < 0$ $\forall \omega \in \mathbb{R} \cup \{+\infty\}$ and hence Π_λ has a rescaled J-spectral factorization $(\Psi_\lambda, M_\lambda)$ as defined above. Let X denote the corresponding stabilizing solution to the ARE (Equation A.2) with $(\tilde{A}, \tilde{B}, \tilde{Q}_\lambda, \tilde{S}_\lambda, \tilde{R}_\lambda)$. Finally, assume the nominal system H_ρ is stable.*

Then, using the extended system notation defined above, the symmetric matrix $P = P^T$ satisfies Equation 4.15 for all $\rho \in \mathcal{P}$ if and only if $\tilde{P} := P + \begin{bmatrix} 0 & 0 \\ 0 & X \end{bmatrix} \geq 0$ satisfies Equation 4.16 for all $\rho \in \mathcal{P}$.

Proof. See Appendix C. □

This technical lemma is used to prove the following main result.

Theorem 3. *Assume $\mathcal{F}_u(H_\rho, \Delta)$ is well posed for all $\Delta \in \mathbf{\Delta}_{1/\gamma}(\Pi_1, \ldots, \Pi_N)$. Then H_ρ achieves robust performance of level γ if there exists a matrix $P = P^T \in \mathbb{R}^{(n_H + n_\psi) \times (n_H + n_\psi)}$ and scalars $\{\lambda_k\}_{k=1}^N$ such that:*

(i) (P, λ, γ) satisfy the parameterized matrix inequality in Equation 4.15 for all $\rho \in \mathcal{P}$

(ii) $\lambda_1 \in \mathbb{R}_{>0}$ and $\lambda_k \in \mathbb{R}_{\geq 0}$ $(k = 2, \ldots, N)$.

Proof. As described above, condition (ii) along with Assumptions 3 and 4 are sufficient to ensure the combined multiplier $\Pi_\lambda := \sum_{k=1}^N \lambda_k S_{1/\gamma} \Pi_k S_{1/\gamma}$ satisfies $(\Pi_\lambda)_{11}(j\omega) > 0$ and $(\Pi_\lambda)_{22}(j\omega) < 0$ $\forall \omega \in \mathbb{R} \cup \{+\infty\}$. Hence Π_λ has a rescaled J-spectral factorization $(\Psi_\lambda, M_\lambda)$. Define $\tilde{P} := P + \begin{bmatrix} 0 & 0 \\ 0 & X \end{bmatrix} \geq 0$ where X is the stabilizing ARE solution used to construct this factorization. By Lemma 1, $\tilde{P}$ satisfies the parameterized matrix in equality in Equation 4.16. The remainder of the proof is based on dissipation theory using the storage function $V : R^{n_H + n_\psi} \to \mathbb{R}_+$ defined as $V(x) := x^T \tilde{P} x$. Left and right multiply Equation 4.16 by $[x^T, w^T, d^T]$ and $[x^T, w^T, d^T]^T$ to show that V satisfies the dissipation inequality:

$$\dot{V}(t) + z_\lambda^T(t) M_\lambda z_\lambda(t) \leq d(t)^T d(t) - \gamma^{-2} e(t)^T e(t) \tag{4.17}$$

Append Ψ_λ to the (v, w) channels of the uncertain system $\mathcal{F}_u(H_\rho, \Delta)$. This corresponds to the interconnection shown in Figure 4.4 except with Ψ_λ replacing $\Psi_{1/\gamma}$. Let (x, w, d, z, e) the solution of this interconnection for some $\Delta \in \mathbf{\Delta}_{1/\gamma}(\Pi_1, \ldots, \Pi_N)$, disturbance $d \in L_2^{n_d}$, admissible trajectory $\rho \in \mathcal{T}$, and zero initial conditions. Integrating the dissipation inequality (4.17) along this solution from $t = 0$ to $t = T$ yields:

$$V(x(T)) + \int_0^T z_\lambda(t)^T M_\lambda z_\lambda(t)\, dt + \frac{1}{\gamma^2} \int_0^T e(t)^T e(t)\, dt \leq \int_0^T d(t)^T d(t)\, dt \tag{4.18}$$

It follows from $\lambda_k \geq 0$ that $\Delta \in \mathbf{\Delta}(\Pi_\lambda)$. In addition, the rescaled J-spectral factorization $(\Psi_\lambda, M_\lambda)$ is a hard factorization of Π_λ by Lemma 6. Therefore, $(\Psi_\lambda, M_\lambda)$ is a valid time domain IQC for Δ. Apply this time domain IQC along with $\tilde{P} \geq 0$ to Equation 4.18 to conclude that $\|e\| \leq \gamma \|d\|$. Hence H_ρ achieves RP of level γ. $\square$

The parameterized matrix inequality (Equation 4.15) in condition (i) of Theorem 3 involves N IQCs. Note that left/right multiplying Equation 4.15 by $[x^T, w^T, d^T]$ and $[x^T, w^T, d^T]^T$ does not yield a true dissipation inequality for two reasons. First, (Ψ_k, M_k) does not need to be a hard factorization and hence not a valid time domain IQC by Definition 2. Second, the matrix P need not be positive definite and thus does not necessarily define a valid storage function. The technical result in Lemma 1 addresses both issues. It converts the original problem in an alternative form (equation 4.16) involving only a single, valid time

domain IQC. The alternative form involves $\tilde{P} = P + \begin{bmatrix} 0 & 0 \\ 0 & X \end{bmatrix} \geq 0$ which defines a valid storage function. The term X, which is the stabilizing solution of the ARE used to construct a hard factorization, can be interpreted as additional energy. Finally, it is important to recall that soft IQCs only hold, in general, over infinite time horizons and they require the signals (v, w) to be in L_2. Hence soft IQCs cannot be used in the dissipation inequality proof since we do not know, a priori, that (v, w) are in L_2. On the other hand, hard IQCs hold over all finite time horizons and for all signals (v, w) in the extended space L_{2e}. Hence a hard factorization of Π_λ is needed.

The conditions in Theorem 3 are sufficient to prove that the uncertain system satisfies a certain level of robust performance (Definition 3). This ensures that $\|\mathcal{F}_u(H_\rho, \Delta)\| \leq \gamma$ for all uncertainty $\Delta \in \mathbf{\Delta}_{1/\gamma}$. This is a (finite-gain) input-output stability result, i.e. the gain from input to output is less than $\gamma < \infty$, that appears frequently in literature, e.g. [94]. In fact, feasibility of the parameterized LMI in Equation 4.16 is sufficient to prove a type of internal stability. In particular, the arguments in [95] (Proposition 1.2 and its proof) demonstrate that the extended state x converges asymptotically to zero from any initial condition and for any disturbance input $d \in L_2^{n_d}$. In general, exponential convergence of the extended state x does not hold and no conclusions regarding the "internal" state of Δ can be made.

4.4.2 Scaled System

This section constructs a specific scaled system that will be used to link the analysis and synthesis steps in our robust (IQC) synthesis algorithm. Consider the uncertain system $\mathcal{F}_u(H_\rho, \Delta)$. Theorem 3 provides a sufficient condition to ensure that H_ρ achieves robust performance of level γ. The proof involves a rescaled J-spectral factorization $(\Psi_\lambda, M_\lambda)$. In particular, robust performance is shown via a dissipation inequality (Equation 4.17) defined for the extended system of H_ρ and Ψ_λ. Recall that the J-spectral factorization was rescaled so that $M_\lambda := \begin{bmatrix} \gamma^{-2}I & 0 \\ 0 & -I \end{bmatrix}$. Thus partitioning $z_\lambda := \begin{bmatrix} v_\lambda \\ w_\lambda \end{bmatrix}$ simplifies the dissipation inequality to

$$\dot{V}(t) \leq \left(d(t)^T d(t) - \gamma^{-2} e(t)^T e(t) \right) + \left(w_\lambda(t)^T w_\lambda(t) - \gamma^{-2} v_\lambda(t)^T v_\lambda(t) \right) \quad (4.19)$$

The form of this dissipation inequality implies a connection to nominal induced L_2 gain performance. Note that Theorem 1 in Section 3.2 provides a sufficient condition to upper bound the induced L_2 gain of an LPV system. The proof for this nominal performance condition uses a dissipation inequality (Equation 3.4) that is similar to Equation 4.19. In particular, Equation 4.19 has the form of a dissipation inequality used to prove a (not-uncertain) LPV system with inputs (w_λ, d) and outputs (v_λ, e) has induced gain $\leq \gamma$. Based

67

on this insight, a scaled system will be constructed with these inputs and outputs. First, rewrite the extended system of H_ρ and Ψ_λ (Equation 4.14) by partitioning $z_\lambda := \begin{bmatrix} v_\lambda \\ w_\lambda \end{bmatrix}$:

$$
\begin{bmatrix} \dot{x} \\ v_\lambda \\ w_\lambda \\ e \end{bmatrix} = \begin{bmatrix} \mathcal{A}(\rho) & \mathcal{B}_w(\rho) & \mathcal{B}_d(\rho) \\ \mathcal{C}_{v_\lambda}(\rho) & \mathcal{D}_{v_\lambda w}(\rho) & \mathcal{D}_{v_\lambda d}(\rho) \\ \mathcal{C}_{w_\lambda}(\rho) & \mathcal{D}_{w_\lambda w}(\rho) & \mathcal{D}_{w_\lambda d}(\rho) \\ \mathcal{C}_e(\rho) & \mathcal{D}_{ew}(\rho) & \mathcal{D}_{ed}(\rho) \end{bmatrix} \begin{bmatrix} x \\ w \\ d \end{bmatrix} \tag{4.20}
$$

Assume that $\mathcal{D}_{w_\lambda w}(\rho)$ is nonsingular $\forall \rho \in \mathcal{P}$. Then the output equation for w_λ can be rewritten as:

$$
w = \mathcal{D}_{w_\lambda w}(\rho)^{-1} \left(w_\lambda - \mathcal{C}_{w_\lambda}(\rho)x - \mathcal{D}_{w_\lambda d}(\rho)d \right) \tag{4.21}
$$

Use this relation to substitute for w in the extended system (Equation 4.20). This gives the following "scaled" system with inputs (w_λ, d) and outputs (v_λ, e) (neglecting dependence on ρ):

$$
\begin{bmatrix} \dot{x} \\ v_\lambda \\ e \end{bmatrix} = \left(\begin{bmatrix} \mathcal{A} & \mathcal{B}_w & \mathcal{B}_d \\ \mathcal{C}_{v_\lambda} & \mathcal{D}_{v_\lambda w} & \mathcal{D}_{v_\lambda d} \\ \mathcal{C}_e & \mathcal{D}_{ew} & \mathcal{D}_{ed} \end{bmatrix} \begin{bmatrix} I & 0 & 0 \\ -\mathcal{D}_{w_\lambda w}^{-1}\mathcal{C}_{w_\lambda} & \mathcal{D}_{w_\lambda w}^{-1} & -\mathcal{D}_{w_\lambda w}^{-1}\mathcal{D}_{w_\lambda d} \\ 0 & 0 & I \end{bmatrix} \right) \begin{bmatrix} x \\ w_\lambda \\ d \end{bmatrix} \tag{4.22}
$$

The use of the term "scaled" system will be further clarified below. The next lemma gives a formal statement connecting robust performance of the extended system to nominal performance of this scaled system.

Lemma 2. *Let $\tilde{P} \geq 0$ and $\gamma > 0$ be given. The following statements are equivalent:*

1. *$(\tilde{P}, \gamma)$ satisfy the robust performance LMI associated with the extended system of H_ρ and Ψ_λ for all $\rho \in \mathcal{P}$:*

$$
\begin{bmatrix} \tilde{P}\mathcal{A}+\mathcal{A}^T\tilde{P} & \tilde{P}\mathcal{B}_w & \tilde{P}\mathcal{B}_d \\ \mathcal{B}_w^T\tilde{P} & 0 & 0 \\ \mathcal{B}_d^T\tilde{P} & 0 & -I \end{bmatrix} + \begin{bmatrix} \mathcal{C}_{v_\lambda}^T & \mathcal{C}_{w_\lambda}^T \\ \mathcal{D}_{v_\lambda w}^T & \mathcal{D}_{w_\lambda w}^T \\ \mathcal{D}_{v_\lambda d}^T & \mathcal{D}_{w_\lambda d}^T \end{bmatrix} M_\lambda \begin{bmatrix} \mathcal{C}_{v_\lambda} & \mathcal{D}_{v_\lambda w} & \mathcal{D}_{v_\lambda d} \\ \mathcal{C}_{w_\lambda} & \mathcal{D}_{w_\lambda w} & \mathcal{D}_{w_\lambda d} \end{bmatrix} + \frac{1}{\gamma^2} \begin{bmatrix} \mathcal{C}_e^T \\ \mathcal{D}_{ew}^T \\ \mathcal{D}_{ed}^T \end{bmatrix} \begin{bmatrix} \mathcal{C}_e & \mathcal{D}_{ew} & \mathcal{D}_{ed} \end{bmatrix} < 0
$$

$$
\tag{4.23}
$$

   where the dependence on ρ has been omitted.

2. *$\mathcal{D}_{w_\lambda w}(\rho)$ is nonsingular $\forall \rho \in \mathcal{P}$. Let $(\mathcal{A}_{scl}, \mathcal{B}_{scl}, \mathcal{C}_{scl}, \mathcal{D}_{scl})$ denote the state-space representation of the scaled system formed from H_ρ and Ψ_λ (Equation 4.22). $(\tilde{P}, \gamma)$ satisfy the induced L_2 gain LMI (Equation 3.3) associated with the scaled system for*

all $\rho \in \mathcal{P}$:

$$\begin{bmatrix} \tilde{P}\mathcal{A}_{scl} + \mathcal{A}_{scl}^T \tilde{P} & \tilde{P}\mathcal{B}_{scl} \\ \mathcal{B}_{scl}^T \tilde{P} & -I \end{bmatrix} + \frac{1}{\gamma^2} \begin{bmatrix} \mathcal{C}_{scl}^T \\ \mathcal{D}_{scl}^T \end{bmatrix} \begin{bmatrix} \mathcal{C}_{scl} & \mathcal{D}_{scl} \end{bmatrix} < 0 \tag{4.24}$$

where the dependence on ρ has been omitted.

Proof. $(1 \Rightarrow 2)$ Assume statement 1 holds. The $(2,2)$ block of Equation 4.23 implies:

$$\gamma^{-2}\mathcal{D}_{v_\lambda w}^T \mathcal{D}_{v_\lambda w} - \mathcal{D}_{w_\lambda w}^T \mathcal{D}_{w_\lambda w} + \gamma^{-2}\mathcal{D}_{ew}^T \mathcal{D}_{ew} < 0 \tag{4.25}$$

This inequality implies $\mathcal{D}_{w_\lambda w}^T \mathcal{D}_{w_\lambda w} > \gamma^{-2}(\mathcal{D}_{v_\lambda w}^T \mathcal{D}_{v_\lambda w} + \mathcal{D}_{ew}^T \mathcal{D}_{ew}) \geq 0$ and hence $\mathcal{D}_{w_\lambda w}$ is nonsingular. Next, define the parameter-dependent congruence transformation:

$$T(\rho) := \begin{bmatrix} I & 0 & 0 \\ -\mathcal{D}_{w_\lambda w}^{-1}(\rho)\mathcal{C}_{w_\lambda}(\rho) & \mathcal{D}_{w_\lambda w}^{-1}(\rho) & -\mathcal{D}_{w_\lambda w}^{-1}(\rho)\mathcal{D}_{w_\lambda d}(\rho) \\ 0 & 0 & I \end{bmatrix} \tag{4.26}$$

T is nonsingular for all $\rho \in \mathcal{P}$. Multiplying Equation 4.23 on the left/right by T^T/T demonstrates that Equation 4.24 holds. The reverse implication $(2 \Rightarrow 1)$ follows by the inverse transformation. Specifically, multiply Equation 4.24 on the left/right by T^{-T}/T^{-1} to show that Equation 4.23 holds. □

Multiplying the robust performance LMI in Equation 4.23 on the left/right by $[x^T, \ w^T, \ d^T]$ and its transpose yields the dissipation inequality in Equation 4.19. The congruence transformation T effectively changes to a dissipation inequality in variables (x, w_λ, d). The lemma states that the robust performance condition for H_ρ is satisfied if and only if the nominal (induced L_2 gain) performance condition is satisfied for the scaled system. The main issue at this point is that the extended system depends on H_ρ and Ψ_λ. Thus the scaled system in Equation 4.22 appears to be a complicated function of the state matrices of H_ρ and Ψ_λ. This is an issue because the robust synthesis algorithm will require the use of this result with the closed-loop, $H_\rho := \mathcal{F}_l(G_\rho, K_\rho)$.

In fact, the scaled system has a particularly simple construction. The extended system is formed by H_ρ and Ψ_λ. The scaled system is essentially formed by inverting the input/output channel associated with w to w_λ. The channel from w to w_λ only involves the filter Ψ_λ. The filter Ψ_λ (given in Equation 4.13) can be expressed in terms of the partitioned output $z_\lambda := \begin{bmatrix} v_\lambda \\ w_\lambda \end{bmatrix}$ as:

$$\begin{bmatrix} \dot{x}_\psi \\ v_\lambda \\ w_\lambda \end{bmatrix} = \begin{bmatrix} \tilde{A} & \gamma^{-1}\tilde{B}_v & \tilde{B}_w \\ \tilde{C}_{v_\lambda} & \tilde{D}_{v_\lambda v} & \tilde{D}_{v_\lambda w} \\ \tilde{C}_{w_\lambda} & \tilde{D}_{w_\lambda v} & \tilde{D}_{w_\lambda w} \end{bmatrix} \begin{bmatrix} x_\psi \\ v \\ w \end{bmatrix} \tag{4.27}$$

The conditions on $\{\lambda_k\}_{k=1}^N$ along with Assumptions 3 and 4 imply $(\Pi_\lambda)_{22}(j\omega) < 0 \ \forall \omega \in \mathbb{R} \cup \{+\infty\}$. This condition at $\omega = \infty$ is sufficient to ensure that $\tilde{D}_{w_\lambda w}$ is nonsingular. If $\tilde{D}_{w_\lambda w}$ is nonsingular then w can be solved in terms of (x_ψ, w_λ, v):

$$w = \tilde{D}_{w_\lambda w}^{-1} \left(w_\lambda - \tilde{C}_{w_\lambda} x_\psi - \tilde{D}_{w_\lambda v} v \right) \tag{4.28}$$

In this case, let $\Psi_\lambda^\dagger$ denote the filter from (v, w_λ) to (v_λ, w) obtained by inverting the w to w_λ channel of Ψ_λ. $\Psi_\lambda^\dagger$ has the following state-space realization:

$$\begin{bmatrix} \dot{x}_\psi(t) \\ v_\lambda(t) \\ w(t) \end{bmatrix} = \left(\begin{bmatrix} \tilde{A} & \gamma^{-1}\tilde{B}_v & \tilde{B}_w \\ \tilde{C}_{v_\lambda} & \tilde{D}_{v_\lambda v} & \tilde{D}_{v_\lambda w} \\ 0 & 0 & I \end{bmatrix} \begin{bmatrix} I & 0 & 0 \\ 0 & 0 & I \\ -\tilde{D}_{w_\lambda w}^{-1}\tilde{C}_{w_\lambda} & \tilde{D}_{w_\lambda w}^{-1} & -\tilde{D}_{w_\lambda w}^{-1}\tilde{D}_{w_\lambda v} \end{bmatrix} \right) \begin{bmatrix} x_\psi(t) \\ w_\lambda(t) \\ v(t) \end{bmatrix}$$
$$\tag{4.29}$$

The next lemma provides an alternative, but equivalent, construction for the scaled system as a simple linear fractional transformation.

Lemma 3. *Assume $\tilde{D}_{w_\lambda w}$ is nonsingular so that $\Psi_\lambda^\dagger$ as defined in Equation 4.29 is well-defined. Moreover, assume $\mathcal{D}_{w_\lambda w}(\rho)$ is nonsingular $\forall \rho \in \mathcal{P}$ so that the scaled system formed from H_ρ and Ψ_λ (Equation 4.22) is well-posed. Then the scaled system is equivalent to the LFT interconnection of H_ρ and $\Psi_\lambda^\dagger$ as shown in Figure 4.5.*
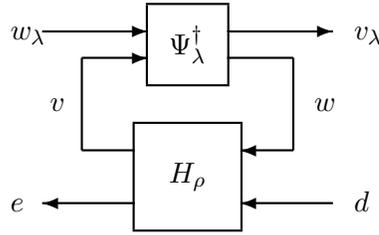
Figure 4.5: LFT interconnection of H_ρ and $\Psi_\lambda^\dagger$.

Proof. The state-space realization for the scaled system (Equation 4.22) is constructed from the state matrices of H_ρ (Equation 4.7) and Ψ_λ (Equation 4.27). The state-space realization for the interconnected system in Figure 4.5 is constructed from the state matrices of H_ρ (Equation 4.7) and $\Psi_\lambda^\dagger$ (Equation 4.29). The proof only involves algebra to verify the equivalence of the two state-space realization. This is straight-forward and hence details are omitted. $\square$

For the special case of LTI uncertainty the scaled system shown in Figure 4.5 reverts to that used in DK synthesis. As noted above, the use of D-scales in DK synthesis (for SISO

LTI uncertainty) corresponds to the frequency domain IQC defined by $\Pi = \begin{bmatrix} \alpha & 0 \\ 0 & -\alpha \end{bmatrix}$ for any SISO, LTI system α such that $\alpha(j\omega) = \alpha(j\omega)^* > 0 \ \forall \omega \in \mathbb{R} \cup \{\infty\}$. Moreover, the rescaled J-spectral factorization in Step 6 is given by $M_\lambda := \begin{bmatrix} \gamma^{-2}I & 0 \\ 0 & -I \end{bmatrix}$ and $\Psi_\lambda = \begin{bmatrix} d & 0 \\ 0 & d \end{bmatrix}$ where d is a spectral factor of α. In this case, inverting the channels w and w_λ yields $\Psi_\lambda^\dagger := \begin{bmatrix} 0 & d \\ d^{-1} & 0 \end{bmatrix}$. The scaled system created by the LFT interconnection of $\mathcal{F}_l(G_\rho^{scl}, K_\rho)$ and $\Psi_\lambda^\dagger$ is thus given by $\begin{bmatrix} d & 0 \\ 0 & 1 \end{bmatrix} \mathcal{F}_l(G_\rho^{scl}, K_\rho) \begin{bmatrix} d^{-1} & 0 \\ 0 & 1 \end{bmatrix}$. This is precisely the standard scaled system that appears in DK synthesis.

4.4.3 Main Theorem

The main technical result for the proposed algorithm is that the iteration is well posed at each step and the robust performance is non-increasing at each iteration. Thus the closed-loop robust performance metric will eventually converge and the iteration in Algorithm 1 will terminate. As with DK synthesis, there are no guarantees that the coordinate-wise iteration will lead to a local optima let alone a global optima. However, the iteration is a useful heuristic that enables robust synthesis to extended naturally from LTI to LPV systems. This main convergence result is now stated.

Theorem 4. *The iteration is well-posed at each step and the iteration is non-increasing, i.e. $\gamma(i) \leq \gamma(i-1)$ for $i = 1, 2, \ldots$.*

Proof. Note that the initial iteration $i = 1$ slightly differs from the consecutive ones. Specifically, the choice of $\lambda(0) = [1, 0, \ldots, 0]$ yields $\Pi_{\lambda(0)} = \Pi_1$ in Step 6 of the first iteration. The definition of Π_1 (Assumption 4) implies that it has a simple J-factorization with $\Psi_1 := I_{n_v + n_w}$ and $M_1 := \begin{bmatrix} I_{n_v} & 0 \\ 0 & -I_{n_w} \end{bmatrix}$ in Step 7. No rescaling is used on the first iteration. The static filter Ψ_1 is equivalent to $z_\lambda := \begin{bmatrix} v_\lambda \\ w_\lambda \end{bmatrix}$ satisfying $v_\lambda = v$ and $w_\lambda = w$. In this case, the scaled system in Step 7 is simply $G_\rho^{scl} = G_\rho$. The synthesis step 8 is then performed with no special modifications for the initial step. As a result, the synthesis step 8 yields a controller $K_\rho(1)$ that stabilizes the system G_ρ and achieves a finite closed-loop gain $\nu(1) < \infty$. This follows because the nominal system G_ρ is quadratically stabilizable and detectable (Assumption 1). The analysis step of the first iteration then achieves a finite robust performance $\gamma(1) < \infty$ because the closed-loop H_ρ is stable. Thus the first iteration is well-posed and achieves $\gamma(1) < \gamma(0) = +\infty$.

Subsequent iterations $(i > 1)$ begin with the iteration count update (Step 4) and performance scaling definition (Step 5). Next the combined multiplier Π_λ is constructed. The coefficients from the previous analysis step satisfy $\lambda_k(i-1) \geq 0$ and $\lambda_1(i-1) > 0$. This fact along with Assumptions 3 and 4 imply that the combined multiplier satisfies $(\Pi_\lambda)_{11}(j\omega) > 0$ and $(\Pi_\lambda)_{22}(j\omega) < 0 \ \forall \omega \in \mathbb{R} \cup \{\infty\}$. Thus the combined multiplier satisfies

the sufficient conditions in Lemma 5 for the existence of a J-spectral factorization. In addition, $(\Pi_\lambda)_{22}(+\infty) < 0$ implies that the feedthrough matrix of Ψ_λ from w to w_λ must be non-singular. In the notation of Section 4.4.2, this corresponds to nonsingularity of $\tilde{D}_{w_\lambda w}$. Hence by Lemma 3, the construction of $\Psi_\lambda^\dagger$ in Step 7 is well-defined.

The analysis step from the previous iteration involves the robust performance parameterized matrix inequality (Equation 4.15) with the factorized IQC multipliers $\{(\Psi_k, M_k)\}_{k=1}^N$. Hence there exists $(P(i-1), \lambda(i-1), \gamma(i-1))$ satisfying Equation 4.15. By Lemma 1, this implies the existence of $\tilde{P}(i-1) \geq 0$ that, along with $(\lambda(i-1), \gamma(i-1))$, satisfies the matrix inequality (Equation 4.15) with the rescaled J-spectral factorization.

Next, Lemma 2 states that feasibility of Equation 4.15 (which is simply Equation 4.23 written in different notation) implies that the scaled closed-loop of $H_\rho := \mathcal{F}_l(G_\rho, K_\rho(i-1))$ and Ψ_λ is well-posed and has induced gain $\leq \gamma(i-1)$. By Lemma 3, this scaled system can be represented by the feedback interconnection of H_ρ and $\Psi_\lambda^\dagger$ as shown in Figure 4.5. Removing the controller, i.e. opening up the u/y channels, yields the scaled open-loop plant. Thus the construction of the scaled system in Step 7 is well-defined.

Finally, the synthesis in Step 8 optimizes over all stabilizing controllers. Hence the new controller $K_\rho(i)$ must yield a cost no greater than that achieved by the previous controller $K_\rho(i-1)$ on the scaled plant. Hence $\nu(i) \leq \gamma(i-1)$. Thus the new controller must satisfy the nominal performance LMI in Equation 4.24 with the slightly larger cost of $\gamma := \gamma(i-1)$. Lemmas 2 and 1 can be used to work backward to the analysis condition in Step 9. Specifically, the closed-loop with new controller $K_\rho(i)$ satisfies the analysis condition in Step 9 with the previous performance level $\gamma(i-1)$, scalings $\lambda(i-1)$ and matrix $P(i-1)$. Step 9 involves optimizing over all feasible coefficients λ and matrix P. This must yield a robust performance cost no greater than the previous step $\gamma(i) \leq \gamma(i-1)$. $\qquad\square$

4.5 Numerical Example

A simple example is used to demonstrate the applicability of the proposed robust synthesis algorithm. The example is based on an example that appears in [81] to test an alternative IQC synthesis algorithm for LTI systems. Here the example is extended to include plant dynamics described by an LPV system. The objective of the example is to design a robust controller for the feedback system shown in Figure 4.6. The nominal plant dynamics are given by the following 2-input, 2-output LPV system F_ρ:

$$\dot{x}(t) = \left(-\frac{1}{71 + 2\rho} I_2\right) x(t) + \left(\frac{1}{71 + 2\rho} I_2\right) u(t) \tag{4.30}$$

$$y(t) = \begin{bmatrix} 87 + 0.2\rho^2 & -87.2 + 0.2\rho^2 \\ 107.4 + 0.2\rho^2 & -110.4 + 0.2\rho^2 \end{bmatrix} x(t) \tag{4.31}$$

The plant dynamics depend on a single scheduling parameter ρ that is restricted to the interval $[1, 3]$. This nominal LPV plant F_ρ was constructed by modifying an LTI model for the idealized distillation process in [96]. The objective is to synthesize a robust controller K_{rob} that offers good tracking performance at low frequencies while penalizing control input at high frequencies. These objectives are specified via the following weights W_e and W_u on the error e and control input u, respectively:

$$W_e(s) = \frac{0.3(s + 0.1)}{2s + 10^{-5}} I_2 \tag{4.32}$$

$$W_u(s) = \frac{s + 10}{s + 100} I_2 \tag{4.33}$$

The controller should also be robust to the uncertainty Δ. The specific assumptions regarding Δ will differ in the various comparisons given below. However, in each case the uncertainty weight is defined as $W_d := \left[\begin{smallmatrix} 0.6 & 0 \\ 0 & 0.3 \end{smallmatrix}\right]$.
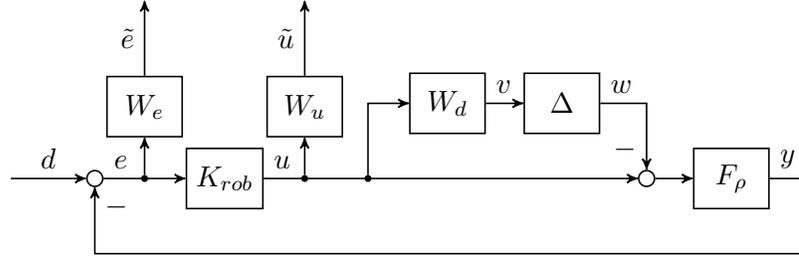

Figure 4.6: Synthesis interconnection for the numerical example.

4.5.1 Comparison to Standard DK Synthesis

First, the proposed algorithm is compared with the DK iteration algorithm. DK synthesis solves the robust synthesis problem for LTI systems. Hence for this comparison the parameter ρ is fixed at 2 to get an LTI model $F_{\rho=2}$. The uncertainty Δ in Figure 4.6 is assumed to be block diagonal, i.e. $\Delta := \left[\begin{smallmatrix} \Delta_1 & 0 \\ 0 & \Delta_2 \end{smallmatrix}\right]$. In addition, each block is assumed to be an LTI uncertainty with norm bound of 1. To apply the proposed algorithm for synthesis, 5 IQCs are selected to model each Δ_i. As stated in Algorithm 1, the first IQC is $\Pi_a := \left[\begin{smallmatrix} 1 & 0 \\ 0 & -1 \end{smallmatrix}\right]$. The remaining 4 IQCs $\{\Pi_{p_k}\}_{k=1}^4$ are given by $\Pi_{p_k} = \Psi_{p_k}^\sim M \Psi_{p_k}$ where $\Psi_{p_k} = \frac{p_k}{s + p_k} I_2$ and $M = \left[\begin{smallmatrix} 1 & 0 \\ 0 & -1 \end{smallmatrix}\right]$. The four poles $\{p_k\}_{k=1}^4$ are spaced logarithmically on $[0.01, 1]$. These multipliers Π_a and $\{\Pi_{p_k}\}_{k=1}^4$ are defined for the blocks Δ_i. They can be combined in the following way to obtain an IQC multiplier for the block diagonal structured uncertainty

$$\Delta := \begin{bmatrix} \Delta_1 & 0 \\ 0 & \Delta_2 \end{bmatrix}:$$

$$(\Pi_i, \Pi_j) := \begin{bmatrix} (\Pi_i)_{11} & 0 & (\Pi_i)_{12} & 0 \\ 0 & (\Pi_j)_{11} & 0 & (\Pi_j)_{12} \\ (\Pi_i)_{21} & 0 & (\Pi_i)_{22} & 0 \\ 0 & (\Pi_j)_{21} & 0 & (\Pi_j)_{22} \end{bmatrix} \tag{4.34}$$

The individual multipliers Π_a and $\{\Pi_{p_k}\}_{k=1}^4$ are combined to construct 9 extended IQCs: $\Pi_1 := (\Pi_a, \Pi_a)$, $\Pi_2 := (\Pi_a, \Pi_{p_1})$, $\Pi_3 := (\Pi_{p_1}, \Pi_a)$, ..., $\Pi_8 := (\Pi_a, \Pi_{p_4})$ and $\Pi_9 := (\Pi_{p_4}, \Pi_a)$. It is easy to check that $\{\Pi_n\}_{n=1}^9$ satisfy Assumptions 2, 3 and 4 in Section 4.3.1.

The synthesis results are listed in Table 4.1. From Case 1 to Case 5, the number of IQCs involved in the synthesis is gradually increased to provide less conservative modeling of Δ. The robust performance has accordingly decreased from 1.06 to 0.88 with the price of increased computation time. As a comparison, the DK synthesis in Case 6 provides a controller with robust performance of 0.75 after $11.84\,s$. For this example, DK synthesis provides a better result with less computation time. This is due, in part, by the use of frequency-gridding and other LTI-specific numerical techniques in standard DK synthesis. The advantage of the proposed algorithm is that it extends, albeit with more computation, to cases where the nominal system is LPV and/or more general classes of uncertainty described by IQCs. This example also illustrates that further research into the parameterization of IQC multipliers is needed. This could potentially reduce the conservatism in the result and also reduce the computational burden.

Table 4.1: LTI robust synthesis results.

Case #	Method	IQCs	Robust Performance	Computation Time (s)
1	Algorithm 1	Π_1	1.06	1.87
2	Algorithm 1	$\{\Pi_n\}_{n=1}^3$	1.00	7.68
3	Algorithm 1	$\{\Pi_n\}_{n=1}^5$	0.96	32.41
4	Algorithm 1	$\{\Pi_n\}_{n=1}^7$	0.89	127.74
5	Algorithm 1	$\{\Pi_n\}_{n=1}^9$	0.88	467.87
6	DK	N/A	0.75	11.84

4.5.2 Robust LPV Synthesis

Next, the proposed algorithm is used to synthesize a robust controller K_{rob} when F_ρ is LPV and Δ is a nonlinear perturbation. Each block of Δ is assumed to be a dead zone operator $w_i = \Delta_i(v_i)$ defined by:

$$w_i = \Delta_i(v_i) := \begin{cases} v_i - b_i, & v_i > b_i \\ 0, & v_i \in [-b_i, b_i] \\ v_i + b_i, & v_i < -b_i \end{cases} \tag{4.35}$$

74

where $b_i = 0.05$ ($i = 1, 2$). The weights W_d, W_e and W_u are the same as above. Standard DK synthesis is unable to synthesize a controller for this case. Three different IQCs are chosen to describe each dead zone Δ_i. The first one is still $\Pi_a = \left[\begin{smallmatrix} 1 & 0 \\ 0 & -1 \end{smallmatrix}\right]$. The second one $\Pi_b = \left[\begin{smallmatrix} 0 & 1 \\ 1 & -2 \end{smallmatrix}\right]$ is used to model the $[0, 1]$ sector bound [19] of the dead zone. The last IQC $\Pi_c = \left[\begin{smallmatrix} 0 & 1+H(s) \\ 1+H^\sim(s) & -(2+H(s)+H^\sim(s)) \end{smallmatrix}\right]$ with $H(s) = \frac{1}{s+1}$ corresponds to a Zames-Falb multiplier. This is used to model the monotonic odd nonlinearity [19]. Five extended IQCs (Equation 4.34) are used to model the structured uncertainty $\Delta := \left[\begin{smallmatrix} \Delta_1 & 0 \\ 0 & \Delta_2 \end{smallmatrix}\right]$: $\bar{\Pi}_1 := (\Pi_a, \Pi_a)$, $\bar{\Pi}_2 := (\Pi_a, \Pi_b)$, $\bar{\Pi}_3 := (\Pi_b, \Pi_a)$, $\bar{\Pi}_4 := (\Pi_a, \Pi_c)$ and $\bar{\Pi}_5 := (\Pi_c, \Pi_a)$.

Several cases are considered to explore properties of the proposed robust synthesis algorithm. A stopping criteria $\epsilon_{tol} = 0.05$ is used in all iterations. Cases (i) and (ii) approximate F_ρ with 5 and 11 gridding points, respectively, spaced equally in the parameter range $[1, 3]$. In both cases, $\{\bar{\Pi}_k\}_{k=1}^5$ are used to describe Δ. The synthesis results are shown in Table 4.2. The denser grid in Case (ii) provides a more accurate approximation to the true, infinite-dimensional synthesis problem. This yields a large (worse) robust performance measure because the LMI constraints are enforced on more points and this also requires more computation time. As mentioned in Section 3.2, a common approach in practice is to use a sparse gridding set for synthesis and verify the performance afterwards using a denser gridding set. Here the two results are sufficiently close that we will focus on Case (i) for further study.

Next, Cases (iii) and (iv) also use 5 grid points to approximate F_ρ but the IQCs used to describe Δ are $\{\bar{\Pi}_k\}_{k=1}^3$ and $\bar{\Pi}_1$, respectively. Comparison of Cases (i), (iii) and (iv) illustrates a basic trade-off with the multipliers. Specifically, a more conservative result (larger robust performance) comes with less constraints on Δ. Additional multipliers reduce this conservatism by further constraining the input/output behavior of Δ. The additional multipliers come with the expense of computation time: Case (i) takes 46.89s compared to 4.75s for Case (iii) and 3.55s for Case (iv).

Finally, Case (v) uses an alternative synthesis algorithm for comparison. Algorithm 1 decouples the design into two steps which are each individually convex. Case (v) is instead solved using a simple line search for a single IQC scaling λ_1. Specifically, a value of λ_1 is selected and used to construct a combined multiplier $\Pi_\lambda := \lambda_1 \bar{\Pi}_1$. Then a robust controller is synthesized with this Π_λ using Steps 7 and 8 in Algorithm 1. This one dimensional search is performed with 100 values of λ_1 selected equidistantly in the interval $[0.02, 2]$. This yields a robust performance of 0.87 in 233.15s. The alternative approach used for Case (v) can, in theory, yield the global optimum with an exact search in the IQC scaling space. This idea can be extended to cases with more than one IQC multiplier/scaling. The gap between Cases (iv) and (v) is another indicator that the proposed algorithm yields a suboptimal solution. However, the alternative search used in Case (v) requires significantly

more computation and becomes infeasible for more than a few IQC multipliers and scalings.

Table 4.2: LPV robust synthesis results.

Case #	Method	Gridding Points	IQCs	Robust Performance	Time (s)
(i)	Algorithm 1	5	$\{\bar{\Pi}_n\}_{n=1}^{5}$	0.96	46.89
(ii)	Algorithm 1	11	$\{\bar{\Pi}_n\}_{n=1}^{5}$	0.99	103.43
(iii)	Algorithm 1	5	$\{\bar{\Pi}_n\}_{n=1}^{3}$	0.98	4.75
(iv)	Algorithm 1	5	$\bar{\Pi}_1$	1.00	3.55
(v)	Line Search	5	$\bar{\Pi}_1$	0.87	233.15

Finally, the LPV robust controller K_ρ synthesized in Case (i) is compared with a nominal LPV controller K_{nom} designed for the system without uncertainty ($\Delta = 0$). K_{nom} is designed using the standard LPV synthesis method described in Section 3.2 again using 5 grid points to approximate F_ρ. The induced L_2 norm of the nominal system using K_{nom} and K_{rob} is given by 0.42 and 0.56, respectively. As expected, K_{nom} achieves better nominal performance as measured with the L_2 norm bound. Next, the robust performance of the closed-loop was assessed using the matrix inequality condition in Section 4.4.1. This yields 3.03 and 0.96 for K_{nom} and K_{rob}, respectively. As expected, the robust design K_{rob} achieves better robust performance. The gap in robust performance between the two controllers is also illustrated by a time domain step response simulation (Figure 4.7). In the simulation, unit step signals are injected into both channels of d simultaneously at $t = 10\,s$ and the parameter trajectory is given by $\rho(t) = \sin(0.05\,t) + 2$. The responses of y_1 and y_2 are shown in Figure 4.7. It is seen that K_{nom} performs well (solid blue curve) when there is no uncertainty in the system. However, it degrades dramatically (dash-dot red curve) when the uncertainty is added. In contrast, K_{rob} maintains good tracking and steady state error (dash green curve) with existence of the uncertainty.
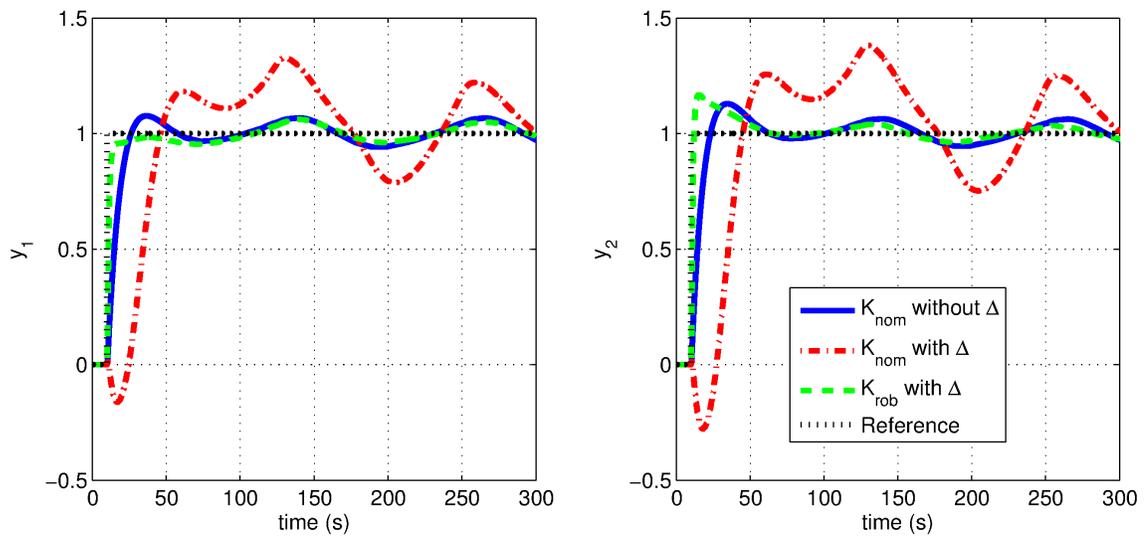
Figure 4.7: Step responses for K_{nom} and K_{rob}.

Algorithm 1 Robust Synthesis for LPV Systems

1: **Given:** LPV system G_ρ and multipliers $\{\Pi_k\}_{k=1}^N$ satisfying Assumptions 1-4; Stopping tolerance parameters $i_{max} \in \mathbb{N}$ and $\epsilon_{tol} > 0$.

2: **Initialization:** Initialize the iteration count to $i = 0$. Set $\lambda(0) = [1, 0, \ldots, 0] \in \mathbb{R}_{\geq 0}^N$ and $\gamma(0) = +\infty$. Factorize each Π_k as (Ψ_k, M_k) with $\Psi_k \in \mathbb{RH}_\infty^{n_z \times (n_v + n_w)}$ according to Lemma 4 in Appendix A.

3: **if** $i < i_{max}$ **then**

4:     **Iteration Count:** Increment count $i := i + 1$.

5:     **Performance Scaling:** If $i > 1$ then define the scaling matrix $S(i-1) := \begin{bmatrix} \gamma^{-1}(i-1)I_{n_v} & 0 \\ 0 & I_{n_w} \end{bmatrix}$, otherwise $S(0) := I_{n_v + n_w}$.

6:     **Combined Multiplier:** Construct $\Pi_\lambda := \sum_{k=1}^N \lambda_k(i-1)S(i-1)\Pi_k S(i-1)$. Compute a J-spectral factorization $(\Psi_\lambda, M_\lambda)$ of Π_λ according to Lemma 5 in Appendix A.

7:     **Scaled System Construction:** Assume Ψ_λ has the state-space realization as in Equation 4.27. Invert the w/w_λ channels to construct $\Psi_\lambda^\dagger$ with state-space realization in Equation 4.29. Form the (open-loop) scaled system G_ρ^{scl} as shown in Figure 4.3 by connecting: the first n_v outputs of G_ρ to the last n_v inputs of $\Psi_\lambda^\dagger$ and the last n_w outputs of $\Psi_\lambda^\dagger$ to the first n_w inputs of G_ρ. The scaled system has inputs (w_λ, d, u) and outputs (v_λ, e, y).

8:     **Synthesis Step:** Use Theorem 2 in Section 3.2 to solve the synthesis problem with the scaled plant: $\min_{K_\rho} \|\mathcal{F}_l(G_\rho^{scl}, K_\rho)\|$. This minimizes the (upper bound) on the closed-loop induced gain from (w_λ, d) to (v_λ, e). The result is the bound on closed-loop induced gain, denoted $\nu(i)$, and controller $K_\rho(i)$.

9:     **Analysis Step:** Use Theorem 3 in Section 4.4.1 to compute the best upper bound on the robust performance of the closed-loop $H_\rho := \mathcal{F}_l(G_\rho, K_\rho(i))$ with respect to $\mathbf{\Delta}(\Pi_1, \ldots, \Pi_N)$. Enforce $\lambda \in \mathbb{R}_{\geq 0}^N$ and $\lambda_1(i) > 0$ in this calculation. The result is the robust performance bound $\gamma(i)$, scalars $\{\lambda_k(i)\}_{k=1}^N$, and storage function matrix $P(i) = P(i)^T$.

10:     **Termination Condition:** If $\gamma(i) - \gamma(i-1) \leq \epsilon_{tol}$ then stop the iteration.

11: **end if**

12: **Return:** Final controller $K_\rho(i)$ and robust performance upper bound $\gamma(i)$.

Chapter 5

Robust LPV Design for Active Power Control

5.1 Motivation

As discussed in Chapter 1 and Section 2.1, the power output of wind turbines operated in the traditional mode is variable due to time-varying wind speeds and this may cause unreliable operation of the power grid. This is not a significant issue when wind power is only a small portion of the total electricity generated on the grid. However, to integrate higher levels of variable wind power into the grid it is important for wind turbines to provide active power control (APC) [4]. APC can be used for the turbine to respond to fluctuations in grid frequency, termed primary response, and to the power curtailment command from transmission system operator, termed secondary response or automatic generation control (AGC) [97].

However, traditional wind turbine control systems as introduced in Section 2.5 and Chapter 3 do not provide active power control. The power electronics used in variable speed wind turbines decouple the mechanical/inertial turbine dynamics from the power grid. Thus a wind turbine with a traditional control law does not have the inertial response to a grid frequency event like a conventional coal power generator [98]. As a result the wind turbine does not participate in the primary response. Moreover, the power output from the turbine fluctuates with variations in wind speed. As a result, new control strategies are being considered to enable wind turbines to track power commands and possible provide ancillary services [99–104]. Some of these designs provide primary response by using inertia response emulation [99, 100]. Another approach is to operate the wind turbine above the optimal tip speed ratio thus reserving kinetic energy [101, 102]. This approach enables the

wind turbine to track the power commands and hence this can be used to realize AGC. The use of blade pitch control with or without combined generator torque control has also been explored [103,104].

The robust synthesis algorithm proposed in Chapter 4 will be used in this chapter to design an LPV controller to provide APC. The architecture is a 2-input, 2-output controller where collective blade pitch and generator torque are coordinated in order to track power and rotor speed reference commands. Similar to the LPV controller proposed in Chapter 3 for traditional operations, this active power controller has parameter dependence on the wind speed. Actually, this control system architecture can be considered as a natural extension of the LPV controller in Chapter 3, as there is only one extra feedback loop for power reference tracking added to the existing design for APC purposes. The design procedure is therefore significantly simplified as some of the tuning results in Chapter 3 can be directly inherited here. However, different from the design in Chapter 3, the LPV model of wind turbine is slightly modified in this chapter to satisfy performance requirements of APC. In addition, a multiplicative uncertainty is considered in the blade pitch input channel of the turbine model. The synthesized robust LPV controller shows similar performance on APC as a nominal LPV controller designed without considerations of uncertainty. However, the robust controller has much better performance when the worst case uncertainty is added to the system dynamics.

The remainder of this chapter is organized as follows. Section 5.2 briefly describes the proposed control strategy for APC. Section 5.3 gives the detailed design process for the robust LPV controller. Simulation results are presented and discussed in Section 5.4.

5.2   Control Strategy Development

Traditional turbine control systems as reviewed in Section 2.5 and proposed in Chapter 3 do not provide active power control. This section describes the proposed approach to provide the capability to track power reference commands. It is important to note that the wind conditions limit the power that can be generated (in steady-state) by the turbine. Specifically, the turbine must operate within the power vs. wind speed envelope below the blue curve for traditional operations shown in Figure 2.4. Thus active power control is constrained to power reference commands that are within this envelope. Methods to reserve power and operate within this envelope include de-rating, relative spinning reserve, and absolute spinning reserve [102,104,105]. Each of these methods corresponds to operation along a specific power v.s. wind speed curve that lies within the available power envelope.

The proposed approach here is to operate anywhere within the power envelope. This would enable de-rating, relative spinning reserve, and absolute spinning reserve as special cases.

The basic operational concept is shown in Figure 5.1. To operate at one of the (v, P) trim conditions within the envelope of Figure 2.4, the turbine must reduce the power coefficient to a new value $C_p < C_{p*}$ by changing the blade pitch angle and/or the tip speed ratio. As shown in Figure 5.1, there is a contour of possible values of (λ, β) that achieve any value of $C_p < C_{p*}$. For a given (v, P) trim condition, the controller can be designed to operate at any point on the new C_p contour. For example, in low wind speeds the controller proposed in [102] shifts from (λ_*, β_*) to a larger tip speed ratio $\lambda > \lambda_*$ while holding blade pitch fixed at β_*. The benefit of this approach is that the turbine operates at a higher rotor speed and hence retains kinetic energy that can be extracted at a later point in time. To summarize, each (v, P) trim condition corresponds to a desired power coefficient. The selection of (λ, β) along the contour of this desired C_p enables a secondary performance objective to be achieved, e.g. stored kinetic energy, reduced structural loads, etc.
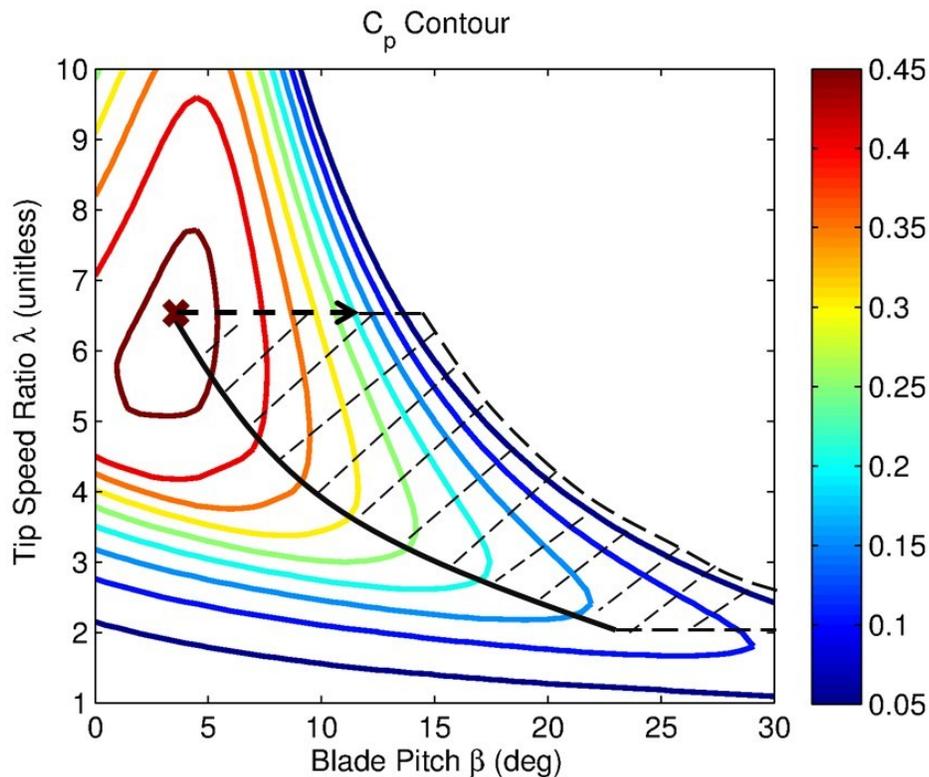

Figure 5.1: Operation envelope for APC.

The controller proposed in this chapter tracks the desired power as follows. In low wind speeds, the controller shifts from (λ_*, β_*) to the desired C_p by increasing to a larger blade pitch $\beta > \beta_*$ while holding tip speed ratio fixed at λ_*. The dash black arrow in Figure 5.1 indicates the proposed shift to the desired C_p in low wind speeds. In constant wind conditions, this approach holds desired rotor speed constant (to maintain λ_*) while blade pitch

angle is increased to shed extra power according to the desired power command. The benefit is that the constant loads on the blade, tower, and gearbox should be reduced by this method of shedding power. However, this approach has the drawback that it will increase the pitch actuator usage. Another drawback of this approach is that less kinetic energy is retained in the rotor than if the turbine were to shift to a larger tip speed ratio.

The APC strategy proposed here can be implemented as the control system structure shown in Figure 5.2. A 2-input, 2-output control system is used to coordinate the blade pitch and generator torque. The main objective is to track the power reference command P_{cmd}. The generator speed command ω_{g_cmd} specifies the desired point on the power coefficient contour. In particular the generator speed command is defined as follows:

$$\omega_{g\_cmd} = \min\left\{N\frac{\lambda_*}{R}v_{trim}, w_{g\_rated}\right\} \tag{5.1}$$

where w_{g_rated} is the rated generator speed and v_{trim} is an estimate of the effective wind speed. As described above, this generator speed command attempts to keep the λ at the optimal value λ_* at lower wind speeds. This will cause an increasing rotor speed demand as wind speed increases. At higher wind speeds, the generator speed command saturates and attempts to maintain the rated value. The solid black curve in Figure 5.1 shows the operation curve for traditional operations in above rated wind speeds. Therefore, the shaded region as shown on the right side of the solid black curve represents the envelope for APC in Region 3.

It is also assumed here that an accurate and real time measurement of the wind speed is available. As shown in Figure 5.2, an estimate of the wind speed could be obtained from a LIDAR [71]. Alternatively, an estimate of the effective wind speed could be constructed [72]. In either case, the actual wind speed fluctuates and hence low-pass filtering, denoted LPF in the figure, is used to smooth out these fluctuations.

5.3 Robust LPV Design

This section provides details on a robust control design using the strategy proposed in Section 5.2. As shown in Figure 5.2, the proposed active power controller has a 2-input 2-output MIMO structure. It forms a closed loop system that is similar to the one proposed in Section 3.3 for traditional operations, except for the extra feedback loop for power reference tracking. This consistency of the system structure provides convenience in the design of a new active power controller. Specifically, some design procedures, such as modeling and controller tuning, can be inherited from the design in Chapter 3 without significant modifications. As a further step of consideration, the design in this Chapter takes the possible
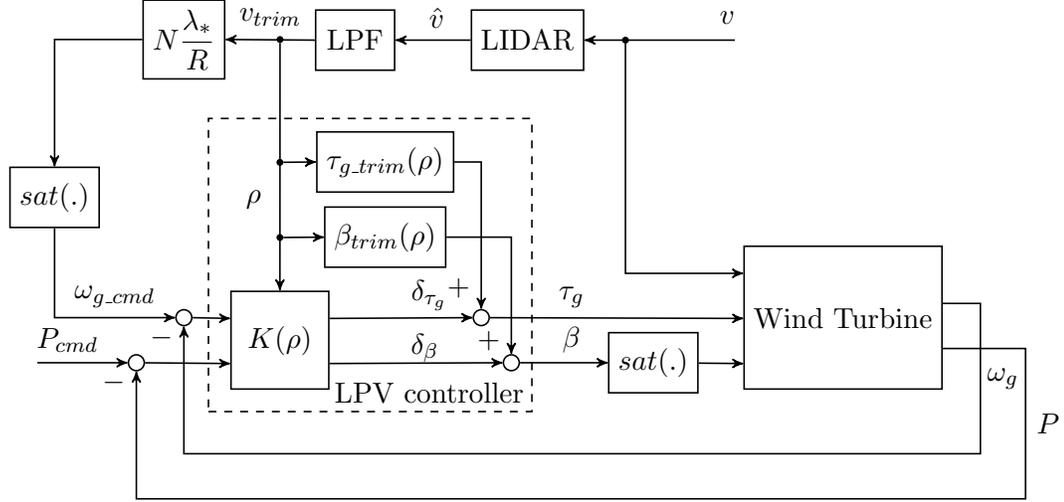
Figure 5.2: The proposed LPV controller for APC.

model uncertainty into account. The robust synthesis algorithm proposed in Chapter 4 is therefore used to find out an LPV controller that ensures robust performance for the uncertain system.

5.3.1 Uncertain LPV Model Construction

The nominal LPV model constructed in Section 3.4.1 covers the dynamics variation with the parameter of wind speed in Regions 2 and 3. A uniform LPV control design as proposed in Section 3.3 is therefore capable of achieving multiple objectives in traditional operations. However, this LPV model is not suitable for APC design purposes. As shown in Section 5.2, APC requires the turbine to operate in the power v.s. wind speed envelope below the traditional operation curve. In the mode of APC, the system dynamics is affected by not only the wind speed, but also the power generation. The LPV model constructed on the traditional operation curve is therefore not accurate enough when the turbine operates in the status of low power generation. In an extreme condition which has been discussed in Section 3.3, the model linearized at any trim point on the maximum power generation curve of Region 2 is theoretically not affected by the input of blade pitch angle. The control strategy proposed in Section 5.2 can never be realized in this case.

To find an LPV model for the APC design, trim points for linearization have been modified as follows. In Chapter 3, 7 trim points were taken uniformly on the maximum power generation curve, as shown by the red circles in Figure 5.3. In the case of APC, these trim points are shifted downwards such that the power generation at each point is 80% of the original value. Other trim values at each point can be found out according to the control strategy proposed in Section 5.2. For instance, the trim generator speed will be the same

after the modification, while the trim generator torque will be 80 % of the original value. The trim blade pitch angle can be found out after the simulation for linearization in FAST. These values are shown by the blue squares in Figure 5.3 with comparison to original values for traditional operations in Chapter 3.

It will be shown in following sections that this LPV model is accurate enough to ensure that the synthesized controller achieves objectives of the APC design. The choice on how much these trim points should be shifted from original ones is based on practice. In a more general setting, the percentage of maximum power generation can be another scheduling parameter for constructing the LPV model. For instance, 2 extra groups of trim points can be chosen such that the power generations are 50 % and 20 % of the maximum value, for each group respectively. Therefore, a more accurate LPV model can be constructed on a 7-by-3 gridding set of the 2 scheduling parameters. This is similar to the approach used in [106] for constructing an LPV model with 2 parameters. However, the use of absolute power generation as one of the parameters in [106] leads to a non-rectangular set of trim points, which is less convenient for applying the LPV toolbox [18] in Matlab. It should also be noted that adding one extra parameter in the LPV model construction will significantly increase the computational time in the following control synthesis. This issue is more critical for the iteration algorithm proposed in Chapter 4 for robust synthesis. Therefore, concerns on the accuracy and complexity should be well balanced in the model construction stage.

As the robust synthesis algorithm to be applied in this chapter is expected to take much longer time for computation than the nominal LPV synthesis algorithm, the linearization in FAST for constructing the LPV model has been simplified to contain only 2 DOFs. These 2 DOFs are the rotor position and tower first fore-aft bending mode. As no DOFs with the blade motion are involved in the model, the MBC transformation is not required in the post analysis. The resulting LTI model therefore has 3 states after removing the state of rotor azimuth angle for avoiding numerical issues. This simplified model captures the essential rotor dynamics and part of the structural dynamics. It will simplify the design and it is useful for verifying the effectiveness of the proposed APC strategy.

Similar to the model used for traditional operations in Chapter 3, the disturbance input is the hub-height wind speed v and 2 control inputs are still the generator torque τ_g and collective pitch angle β. Outputs of the model have been modified to contain the generator speed ω_g and the power generation P [kW] for feedback control purposes. Therefore, the LPV model for APC design can be constructed as

$$\begin{bmatrix} \dot{x} \\ y \end{bmatrix} = \begin{bmatrix} A(\rho) & B_d(\rho) & B_u(\rho) \\ C(\rho) & D_d(\rho) & D_u(\rho) \end{bmatrix} \begin{bmatrix} x \\ d \\ u \end{bmatrix} \tag{5.2}$$
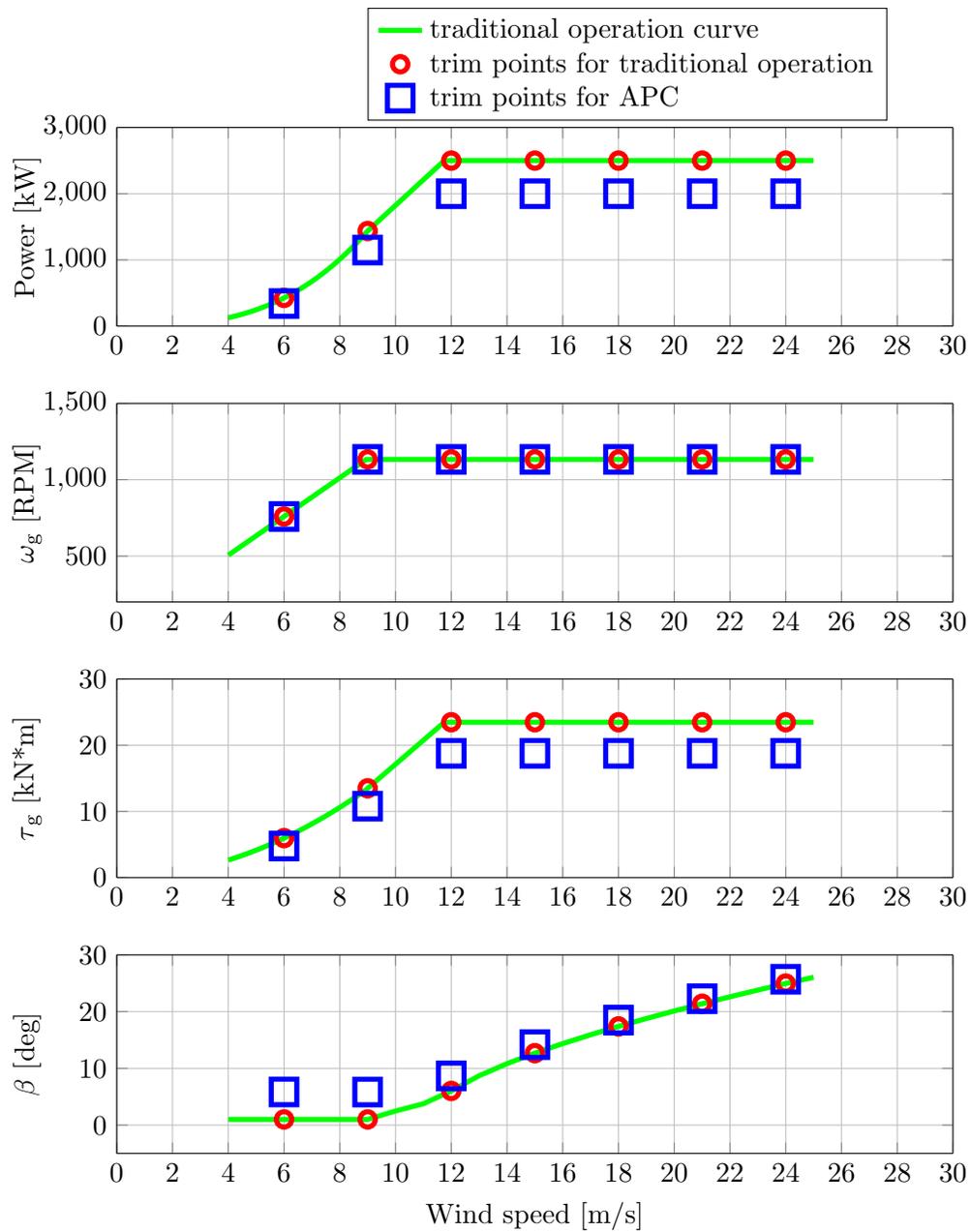
84

Figure 5.3: Trim points for APC design.

where $x \in \mathbb{R}^3$ is the state, $d := \delta v \in \mathbb{R}$ is the disturbance, $u := [\delta\tau_g \ \delta\beta]^T \in \mathbb{R}^2$ is the vector of control inputs and $y := [\delta\omega_g \ \delta P]^T \in \mathbb{R}^2$ is the vector of outputs. This nominal (without uncertainties) LPV model is shown as $G(\rho)$ in Figure 5.4.
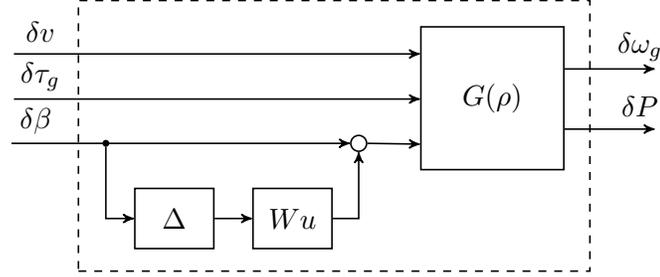

Figure 5.4: Uncertain LPV model of wind turbine.

The model uncertainty is considered in the APC design to ensure enough robustness of the controller. In current stage, there is only a multiplicative uncertainty Δ added to the blade pitch input channel as shown in Figure 5.4. The concern of robustness in this channel is raised as more blade pitch actuations are required for APC. Δ here is assumed to be an LTI uncertainty with norm bound of 1. The weight W_u will be selected later to shape frequency properties of Δ.

5.3.2 Weights Tuning

Frequency loop shaping techniques are still used here for the APC design. Figure 5.5 shows the augmented system for synthesis of the proposed LPV controller. Here, 7 weights (W_e, W_τ, W_β, W_v, W_{Pi}, W_{Pe} and W_u) need to be selected for the loop shaping. Comparing to the LPV design for traditional operations in Chapter 3, there are more weights required here for achieving APC. However, the design process can be significantly simplified by inheriting some of the weights used in Chapter 3. These weights include W_e, W_τ, W_β and W_v, which are the same as listed in Table 3.1.

W_{Pi} and W_{Pe} are used for penalizing the power reference input and tracking error. Here W_{Pi} equals to 20, which corresponds to a power reference command of $20 \, \text{kW}$. Though this value is relatively small, considering the range of power variation from 0 to $2500 \, \text{kW}$, it is considered as a normalization such that the weighted power reference command matches existing weights on control actuations. $W_{Pe}(s) = \frac{0.025s+0.003307}{s+0.006614}$ is the performance weight on the power reference tracking error. Similar to W_e for the generator speed tracking, $W_{Pe}(s)$ emphasizes more on the low frequency error and less on the high frequency part. The low frequency gain of $W_{Pe}(s)$ is 0.5, which corresponds to a steady-state error of 2 RPM. $W_{Pe}(s)$ is tuned to have a bandwidth of $0.05 \, \text{rad/s}$. Thought this value is relatively small comparing to the bandwidth for the generator speed tracking, it is enough to ensure
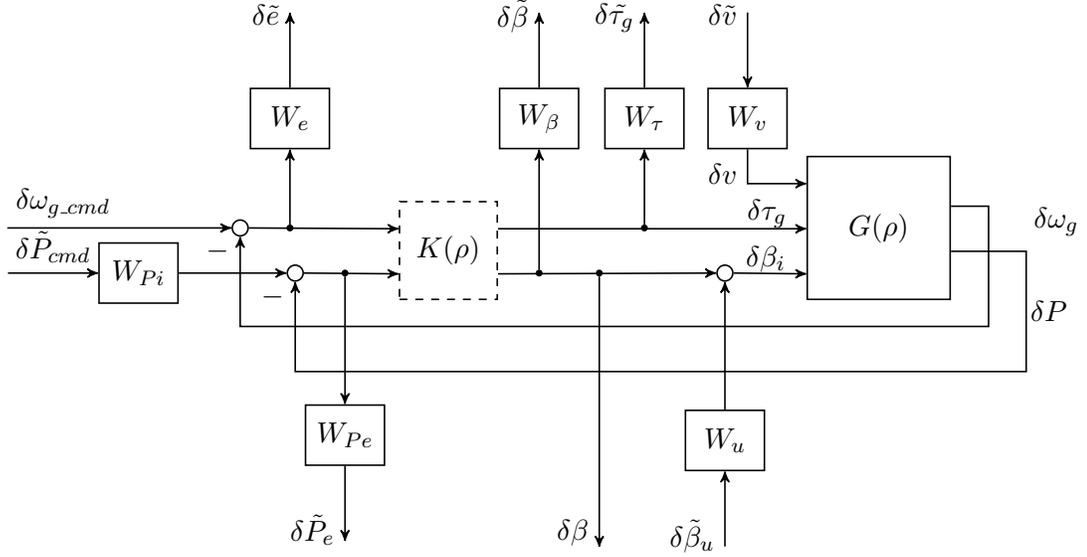
Figure 5.5: Augmented system for APC design.

the performance of APC, as the response in power grids usually takes several minutes or even longer [4]. It should be noted that these 2 weights are not time varying with the parameter ρ.

As described in Section 5.3.1, $W_u = \frac{1.25s+0.3062}{s+1.531}$ is the weight used to shape frequency properties of the normalized uncertainty Δ. As a reasonable assumption, there should be more uncertainty in the high frequency than in the low frequency. Therefore, W_u is selected to have a low frequency gain of 0.2 and a high frequency gain of 1.25, which correspond to uncertain levels of 20 % and 125 %, respectively.

5.3.3   Synthesis Results

The augmented system described in Section 5.3.2 is used to synthesize a robust LPV controller for APC purposes. Similar to the set up for nominal LPV control synthesis in Chapter 3, the parameter varying rate for ρ in the robust synthesis is chosen as $0.1\,\mathrm{m/s^2}$. The corresponding Lyapunov matrices are also set to have an affine dependence on ρ. To start Algorithm 1, 2 IQCs are selected for the normalized uncertainty Δ. As required by the algorithm, the first IQC is $\Pi_1 = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$. The second IQC is defined by

$$\Pi_2 = \begin{bmatrix} \frac{1}{(s+1)^2} & 0 \\ 0 & -\frac{1}{(s+1)^2} \end{bmatrix} \tag{5.3}$$

to simplify the computation. The stopping tolerance ϵ_{tol} is set as 0.04 to terminate the algorithm at a proper time. The synthesized robust LPV controller will be denoted as K_{rob}.

87

At the same time, a nominal LPV controller K_{nom} will be synthesized for comparisons without considering the uncertainty Δ.

Details on the iteration process for the robust LPV controller K_{rob} is shown in Table 5.1. According to Theorem 4, the robust performance should be decreasing non-strictly each iteration. However, it is seen in Table 5.1 that the algorithm converges after 5 iterations and leads to a robust performance of 1.5891, which is slightly larger than the value in the previous iteration. A possible explanation for this phenomenon should be attributed to the numerical error.

Table 5.1: Iteration process for the robust LPV controller.

Iteration #	Time consumption [s]		Robust performance [unitless]
	Synthesis	Analysis	
1	7.4038	387.7608	13.7120
2	14.1918	380.0770	2.9085
3	13.8396	268.1088	1.8678
4	13.6005	316.5674	1.5409
5	13.4095	362.2312	1.5891

The computation time is another concern for the robust synthesis algorithm. It is noted that the computation time for each analysis step is usually around 260 to 400 s, which is much longer than the time consumption of synthesis step. As described in Section 4.3.3, the robust performance bound γ at the analysis step enters bilinearly in the matrix inequality and hence bisection is required to find the minimum feasible value of γ. This bisection takes multiple times and leads to the low efficiency of the algorithm. For instance, it is found out that the bisection in the 1-st analysis step ends after 12 times. This is a critical bottleneck for applications of the robust synthesis algorithm, as the time consumption will be unacceptable for more complicated designs. As the bound γ enters into the matrix inequality bilinearly, it leads to a quasi-convex optimization problem, which can be solved more efficiently using existing method than the bisection search. Therefore, an update of Algorithm 1 should be considered in the future to improve the efficiency.

It is also noted that the synthesis step in the first iteration takes around 7 s while all other synthesis steps need about 14 s. This is due to the size difference of the synthesized controller at each iteration. Specifically, there is only 1 constant IQC Π_1 used in the first iteration of Algorithm 1. Considering the size of the augmented system in Figure 5.5, the synthesized controller at this iteration contains 9 states. However, all other iterations lead to a controller with 11 states, as the second IQC Π_2 is involved. It can be concluded that the computation time for the nominal LPV synthesis is very sensitive to the model size and a careful choice of model complexity is required in applications.

To evaluate performance of the robust LPV controller K_{rob} before simulations, it is first compared with the nominal LPV controller K_{nom} in the case with no uncertainty in the model. The induced L_2 norm of the nominal system using K_{nom} and K_{rob}, is given by 0.81 and 0.99, respectively. As expected, K_{nom} achieves slightly better nominal performance than K_{rob}. Detailed checks are also performed to evaluate frequency responses of these 2 controllers at each trim points. As shown in Section 3.4.3, these local frequency responses provide some complementary information of the system. For instance, Figure 5.6 shows Bode magnitude plots of the sensitivity function from δP_{cmd} to δP_e at trim points of $\rho = 9\,\mathrm{m/s}$ and $18\,\mathrm{m/s}$, which represent operations in below and above rated wind speeds, respectively. It is seen that K_{nom} achieves slightly better power reference tracking performance than K_{rob} in both 2 wind conditions, which fits the numerical analysis results mentioned above.
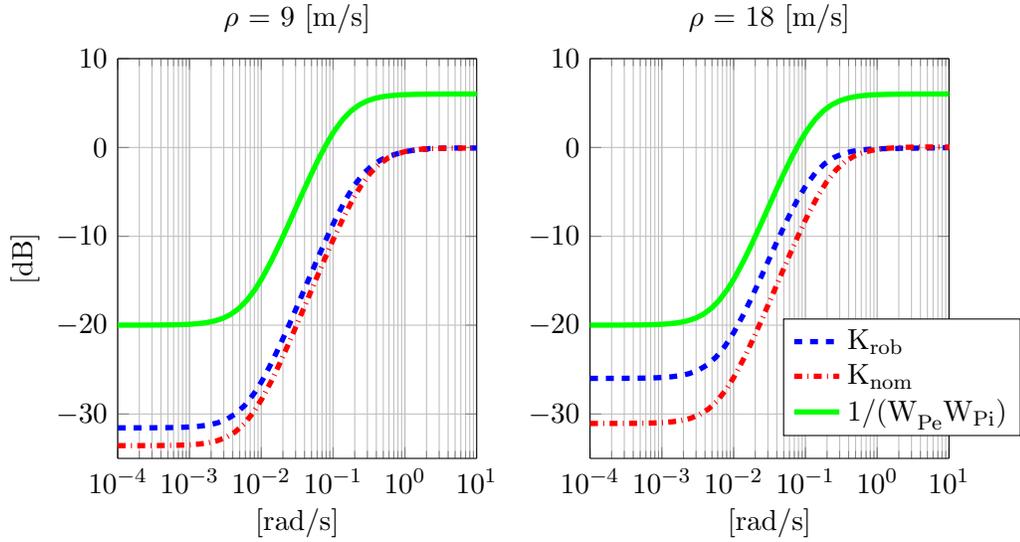

Figure 5.6: Bode magnitude plots of the sensitivity function from δP_{cmd} to δP_e.

In the next step, the controller K_{rob} is compared with K_{nom} when the uncertainty Δ is considered in the turbine model. As shown in Table 5.1, the robust performance of K_{rob} is about 1.6. It indicates that the induced L_2 norm of the uncertain system with K_{rob} has an upper bound of 1.6, when $\|\Delta\| \leq \frac{1}{1.6}$. The matrix inequality condition in Section 4.4.1 is used here to analyze the robust performance of K_{nom} and it yields a value of 2.41. Apparently, K_{rob} achieves better performance than K_{nom} with existence of the uncertainty in the model.

The robustness of K_{rob} and K_{nom} is further analyzed using the metric of worst case gain, as discussed in Section 4.3.1. Here, a specific value of b is chosen as the norm bound of Δ. The worst case gain condition in [107] is used to analyze the induced L_2 norm of the uncertain system when $\|\Delta\| \leq b$. This condition has a small variation to the robust performance

condition in Section 4.4.1 by fixing the norm bound of Δ and it provides an upper bound of the worst case gain γ_{wc}. At the same time, the maximum value of the worst case LTI gain at each trim point provides a lower bound of γ_{wc}. This worst case LTI gain can be computed by using the function `wcgain()` in Matlab [108]. Figure 5.7 shows how the upper and lower bounds of γ_{wc} change with b for K_{rob} and K_{nom} respectively. It is seen that the upper bound for K_{nom} is less than the value for K_{rob} when $b \leq 0.3$. However, the performance of K_{nom} degrades significantly when b gets close to 0.53. This value provides a robust stability margin for K_{nom}. In contrast, K_{rob} robustly stabilizes the uncertain system until $b = 0.86$.
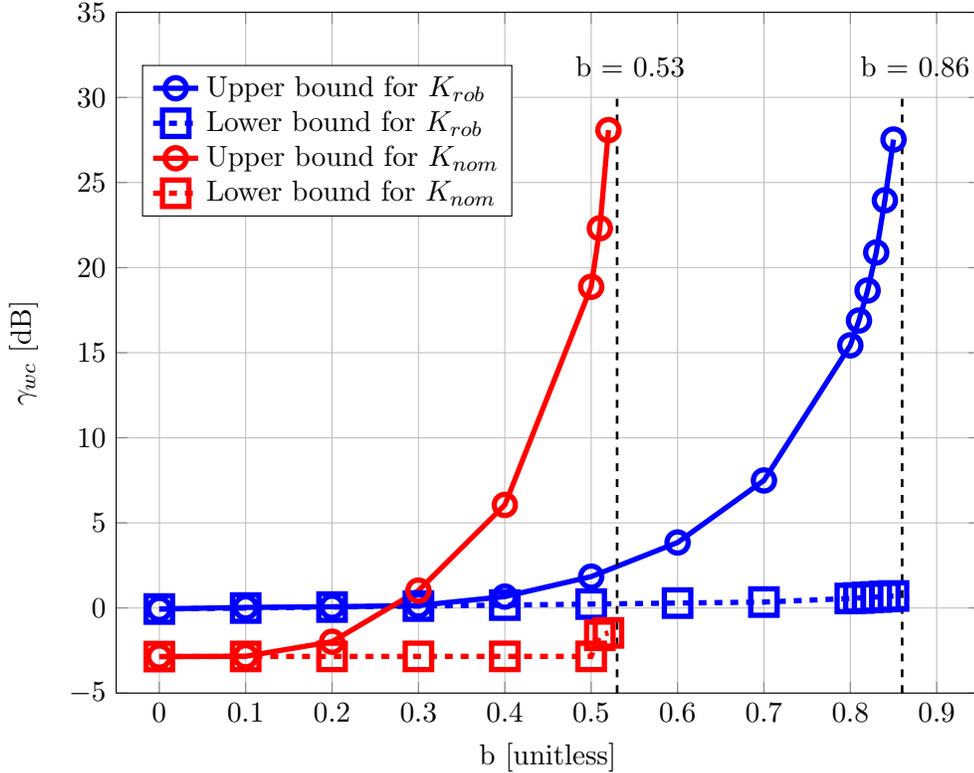

Figure 5.7: Worst case gain analysis for the robust controller K_{rob} and the nominal controller K_{nom}.

It should be noted that the lower bounds for both K_{rob} and K_{nom} have much smaller values, comparing to the upper bounds shown in Figure 5.7. These gaps explain conservativeness of the computation methods used above. Specifically, the LMI condition for worst case gain of uncertain LPV systems [107] is naturally conservative. In addition, the worst case gain for LTI systems at forzen trim points could be much smaller than the real value of γ_{wc}, as the worst case parameter trajectory might contain large variations with time. Instead, [109] provides a less conservative method to compute the lower bound of γ_{wc} by searching over classes of parameter varying trajectories. This method will be considered in the future work to minimize the gap between upper and lower bounds of γ_{wc}.

It should also be noted that the lower bounds in Figure 5.7 only provides limited information on the local robustness as the value of b is not larger enough to destabilize the LTI systems. Instead, Figure 5.8 plots disk margins in the collective pitch input channel for these 2 controllers at each trim points. The disk margins can be calculated using the Matlab function `dmplot()`. It provides a disk gain margin and a disk phase margin of a specific input/output channel, which form an ellipse to guarantee stability of the closed loop system for all combined gain/phase variations. Here, the channel selected is the collective pitch input to the wind turbine, which is consistent with the assumption of uncertainty in Section 5.3.1. As shown in Figure 5.8, K_{rob} has better disk margins than K_{nom} at all trim conditions. For instance, the disk gain margin for K_{rob} is 7.5 at $\rho = 6$ m/s while K_{nom} only has a margin of 3.4. The disk phase margins at this trim condition are 76 deg and 57 deg for K_{rob} and K_{nom}, respectively. However, both these 2 controllers show smaller margins at $\rho = 12$ m/s, which indicate less robustness at the transition between Regions 2 and 3. A possible reason for this phenomenon might come from the variation of control actuations shaped by W_β.
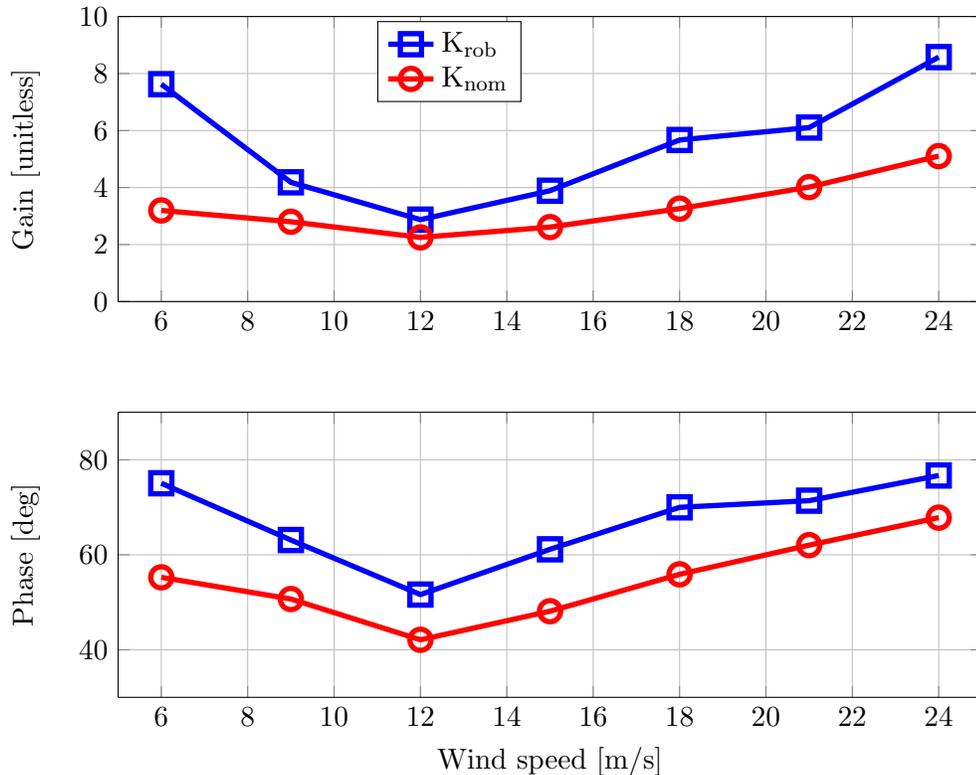

Figure 5.8: Disk margins analysis for the robust controller K_{rob} and the nominal controller K_{nom} at different trim wind speeds.

To conclude here, K_{nom} achieves slightly better nominal performance than K_{rob}. However,

K_{rob} has much better robust performance when the effect of uncertainty can not be ignored in the model. The difference between these 2 controllers will be verified by simulations in the following section.

5.4   Simulations and Analysis

5.4.1   Simulations for APC

Both the robust controller K_{rob} and the nominal controller K_{nom} will be tested in the FAST simulation environment. As these two designs were based on a simplified model of the C96 turbine, structural modes in the FAST simulation include the rotor position, drive train torsion and first tower fore-aft mode. Therefore, the model used for simulations contains 3 DOFs. The objective of simulation is to verify the APC performance without uncertainty in the turbine model. Similar to the set up in Section 3.5, a turbulence level of 5 % is used to generate all wind profiles in TurbSim. In addition, simulation results in the first 60 s will be ignored due to the effect of initial response.

For simplicity, simulation results in 2 different wind speeds are shown in Figures 5.9 and 5.10, which cover Regions 2 and 3 wind conditions. In the first simulation, the mean wind speed is $9\,m/s$. K_{rob} and K_{nom} are simulated to track a power reference as shown by the solid green curve in the second subplot of Figure 5.9. This command steps from 700 kW to 1300 kW at $t = 210\,s$, then drops to 100 kW at $t = 360\,s$ and finally steps back to 700 kW at $t = 560\,s$. It is seen that K_{rob} and K_{nom} have similar responses to the reference signal. The 90 % settling time to the step command is around 20 s, which is reasonable for the objective of APC. It should be noted that the maximum available power at $9\,m/s$ is 1426 kW, according to Equation 2.1. Therefore, the wind turbine might not be able to track the power reference of 1300 kW accurately from $t = 210\,s$ to 360 s, if the wind speed drops too much due to the turbulence. To prevent possible instability of the system in this situation, the power reference signal is saturated by the maximum available power, which can be calculated based on measurements of the trim wind speed. It is seen that the power output drops at $t = 300\,s$ when the wind speed is close to 8 m/s and the system maintains stable due to the saturation. The generator speed is also affected by the wind turbulence. As shown in the third subplot of Figure 5.9, both K_{rob} and K_{nom} try to track the rated generator speed when the wind speed is above 9 m/s. In contrast, the generator speed tracks a time varying command that is proportional to the wind speed when it goes below 9 m/s.

The second simulation is performed at an average wind speed of $18\,m/s$ which is above the rated value. In traditional operations as discussed in Chapter 3, the C96 wind turbine tries to track the rated generator speed and maintain the power output of 2500 kW at this wind speed. Therefore, in the mode of APC, K_{rob} and K_{nom} should be able to track a
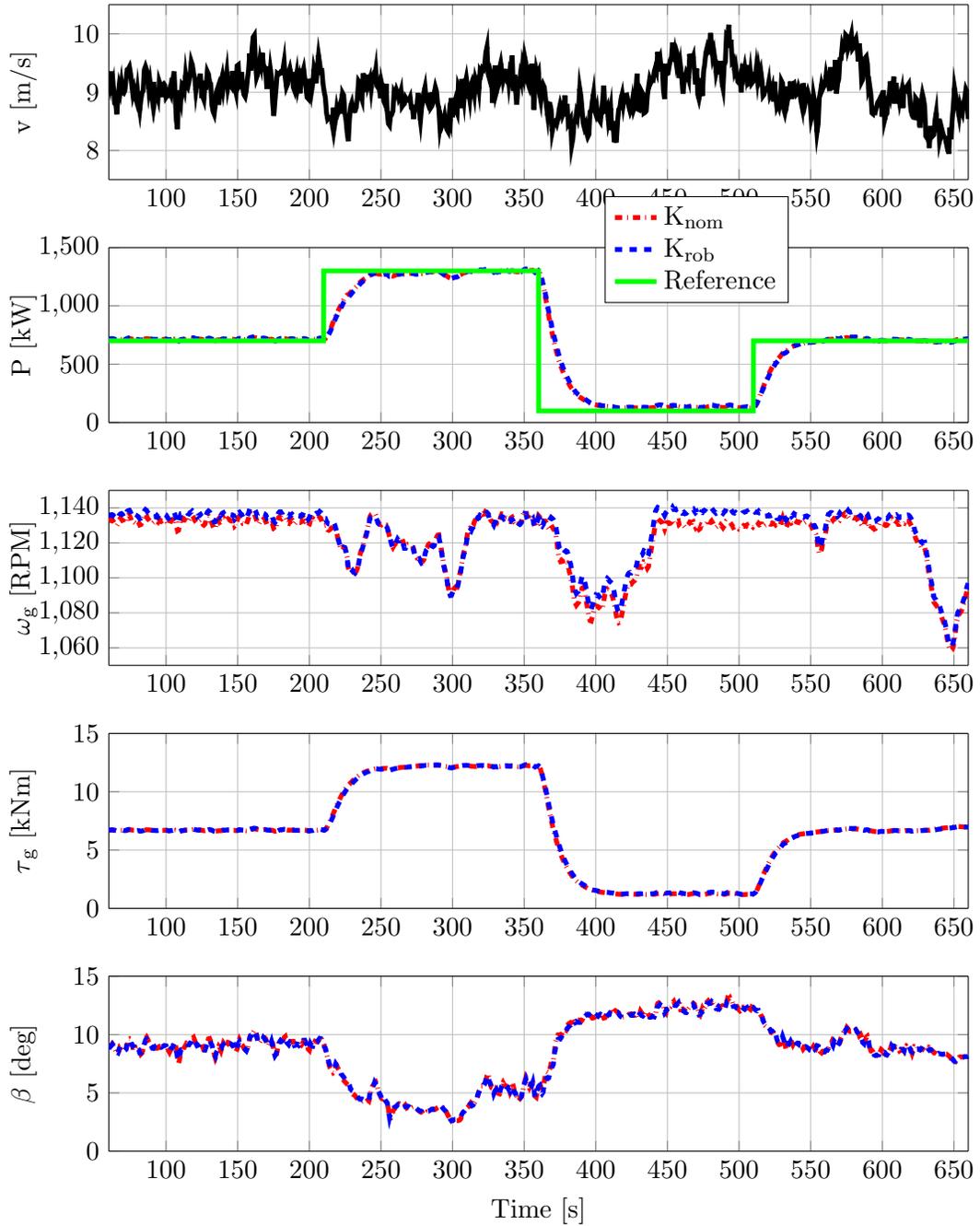
Figure 5.9: Simulations for APC using the robust controller K_{rob} and the nominal controller K_{nom} at $\rho = 9\,\mathrm{m/s}$.

power reference command varying from 0 to the rated value. Here, the power reference is set as shown by the solid green curve in the second subplot of Figure 5.10. It steps from $1500\,\mathrm{kW}$ to $2300\,\mathrm{kW}$ at $t = 210\,\mathrm{s}$, then drops to $700\,\mathrm{kW}$ at $t = 360\,\mathrm{s}$ and finally steps back to $1500\,\mathrm{kW}$ at $t = 560\,\mathrm{s}$. Results of the power reference tracking for K_{rob} and K_{nom} are also shown in this subplot of Figure 5.10. Comparing to the nominal controller K_{nom}, K_{rob} shows a slower response to the reference. This difference can be explained by the sensitivity function of power reference tracking, as shown in the right subplot of Figure 5.6. Specifically, K_{rob} has a smaller bandwidth for tracking the reference signal at this trim condition. A further comparison on the fourth subplot of Figure 5.10 shows that the limited bandwidth of actuation on the generator torque leads to the slow response of K_{rob}. As the wind speed is kept above rated in the simulation, the generator speed command is held fixed at the rated value. Both K_{rob} and K_{nom} show good tracking performance of this constant command, as shown in the third subplot of Figure 5.10. The actuations on the blade pitch are also similar between these 2 controllers.

5.4.2 Worst Case Performance Simulations

Simulations in Section 5.4.1 show that the robust controller K_{rob} has similar performance as the nominal controller K_{nom} for the objective of APC. These results were based on the assumption that there is no uncertainty in the turbine model. Therefore, it is meaningful to explore robust performance of K_{rob} and K_{nom}. The basic idea here is to find out a worst case uncertainty model Δ_{wc} with a fixed norm bound for each controller and then compare the performance degradation after Δ_{wc} is considered in simulations. However, the robust performance condition in Section 4.4.1 does not provide any information on the worst case uncertainty model for LPV systems. As a compromise, the Robust Control toolbox [108] is used again to find out Δ_{wc} for the LTI system at a specific trim point. Here, the trim point is chosen at $\rho = 15\,\mathrm{m/s}$ to get the closed loop LTI model. As described in Section 5.3.1, the uncertainty in the blade pitch input channel is assumed to be LTI and normalized as $\|\Delta\| \leq 1$. It can be modeled using the command `ultidyn()` and interconnected with the nominal system. This interconnected uncertain system is analyzed using the function `wcgain()` which provides detailed information on the worst case gain and the corresponding model of Δ_{wc}. However, it should be noted that the worst case uncertainty Δ_{wc} might be different for K_{rob} and K_{nom}.

These 2 controllers are first simulated in FAST without Δ_{wc} interconnected to the system and the average wind speed is $15\,m/s$ with $5\,\%$ of turbulence. In the next step, Δ_{wc} is included in the system with K_{rob} and K_{nom} for simulations. The performance degradation is analyzed based on the error of output $y_u = y_0 - y_{wc}$. Here y_0 and y_{wc} are outputs of simulation without and without Δ_{wc}. Figure 5.11 shows the error of power output P_u for
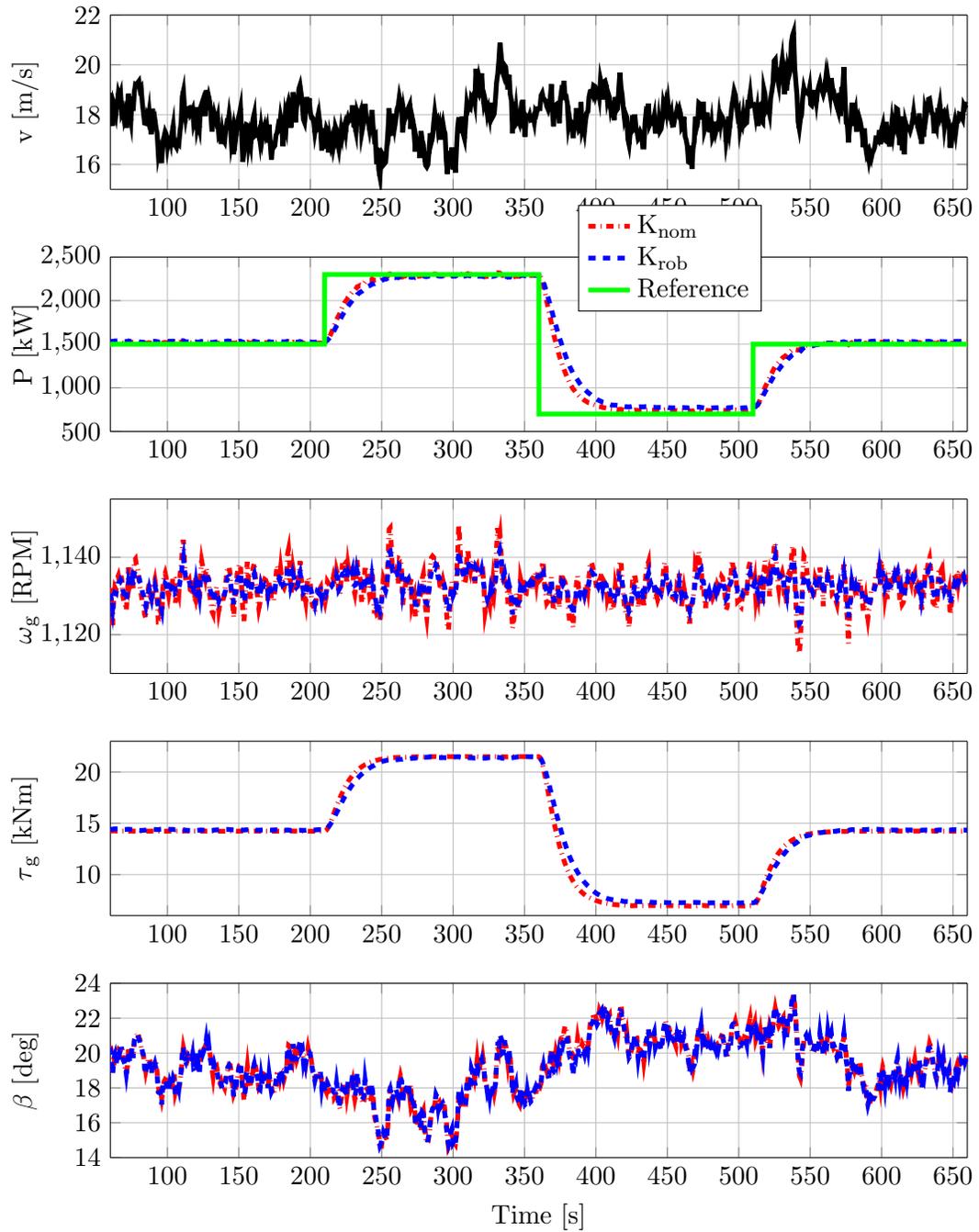
Figure 5.10: Simulation for APC using the robust controller K_{rob} and the nominal controller K_{nom} at $\rho = 18\,\mathrm{m/s}$.

K_{rob} and K_{nom}. It is clear that Δ_{wc} leads to large variation of P_u for K_{nom}. Analysis results for other outputs signals show the similar conclusion and are therefore not plotted here.
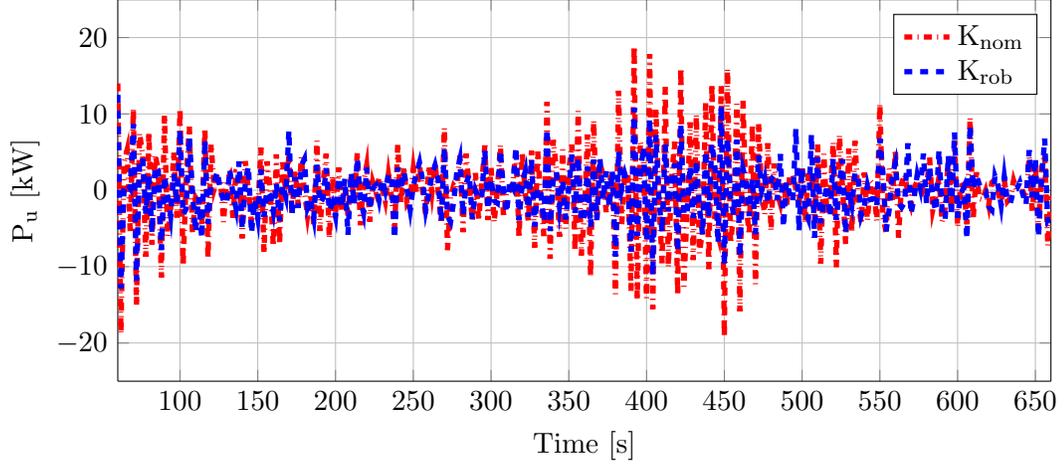

Figure 5.11: Error on the power output with Δ_{wc} using the robust controller K_{rob} and the nominal controller K_{nom}.

To further evaluate these errors, define the root mean square (RMS) of the error y_u as:

$$y_{RMS} = \left(\frac{1}{T} \int_0^T |y_u|^2 dt \right)^{\frac{1}{2}} \tag{5.4}$$

where T is the simulation time. This metric is applied to the generator speed, power output and blade pitch angle. The results are shown in Table 5.2, which provides a quantitative comparison between K_{rob} and K_{nom}. It is seen that K_{rob} has significantly prevented the performance degradation due to the worst case uncertainty. In addition, the damage equivalent load (DEL) is also calculated for the low speed shaft torque. The values in the last row of Table 5.2 show increments of DEL after the Δ_{wc} was included in the model. It is noted that the DEL for K_{nom} has increased by 74.12 %, while this value for K_{rob} is only 34.39 %. Apparently, K_{rob} shows better robust performance than K_{nom} on load reduction.

Table 5.2: Performance degradation with Δ_{wc}.

Description	K_{nom}	K_{rob}
RMS Power Error [kW]	6.2196	4.0241
RMS Speed Error [RPM]	3.8464	1.9653
RMS Pitch Angle [deg]	0.8642	0.4466
Low Speed Shaft DEL Increase	74.12 %	34.39 %

96

Chapter 6

Conclusions

This thesis proposes a uniform MIMO control framework for wind turbines using the robust linear parameter varying (LPV) design method. This framework takes multiple objectives in different wind conditions into a systematic consideration, such that existing results based on SISO linear control design can be integrated together with stability and performance guarantee. As a first step, Chapter 3 proposes an LPV controller that is able to maximize the power generation in Region 2 and track the rated generator speed in Region 3. Considerations on load reduction are also included in the MIMO control structure. The use of recently developed LPV toolbox in Matlab [18] ensures that the designed controller meets objectives in different wind conditions and provides better load reduction performance than the baseline controller.

A further step for the proposed LPV design is to ensure enough robustness of the system. However, robust synthesis algorithms like μ-synthesis [20] is not directly available for LPV systems and existing theories for robust LPV control are still incomplete. Therefore, Chapter 4 proposes a robust synthesis algorithm for a class of uncertain LPV system, using the theory of integral quadratic constraints (IQCs). The robust synthesis problem leads to a non-convex optimization and the proposed algorithm is a coordinate-wise descent similar to the well-known DK iteration for μ synthesis. Specifically, the proposed algorithm alternates between an LPV synthesis step and an IQC analysis step. Both steps can be efficiently solved as semidefinite programs. It is shown that the proposed algorithm ensures that the robust performance is non-increasing at each iteration step.

This robust synthesis algorithm is used in Chapter 5 to design an LPV controller to provide APC. The control architecture is proposed by slightly modifying the LPV controller for traditional operations in Chapter 3, with only one extra feedback loop for power reference tracking. The design procedure is therefore significantly simplified as some of the

tuning results in Chapter 3 can be directly inherited for APC. In addition, a multiplicative uncertainty is considered in the blade pitch input channel of the turbine model. The synthesized robust LPV controller shows similar performance on APC as a nominal LPV controller designed without considerations of uncertainty. However, the robust controller has much better performance when the worst case uncertainty is added to the system dynamics. Therefore, the robust LPV controller has guaranteed performance on APC, even with perturbed system dynamics.

Overall, the robust LPV design framework proposed in this thesis provides a uniform control architecture for wind turbines. This architecture, along with the unified design flow from modeling, weights tuning to simulation, makes possible improvements in the future a natural extension of the existing design. For example, it is possible to construct extra feedback loops in this architecture for further load reduction, as sensors for turbine loads monitoring are incorporated in the system. The popular individual pitch control can also be included to decrease the periodic load on turbine blades. At the same time, more uncertainties can be considered in the design to ensure robust performance of the controller, such as parametric uncertainty in the turbine model, saturation/rate-limitation for actuators, etc.

Another important direction for future works is to improve efficiency of the robust synthesis algorithm in Chapter 4. As shown in Chapter 5, this algorithm takes much longer time for computation than the nominal synthesis algorithm included in the LPV toolbox. Therefore, there were only 2 DOFs considered in the turbine model to simplify the APC design. This is a critical bottleneck for applications of the robust synthesis algorithm, as the time consumption will be unacceptable for more complicated designs. As the robust performance bound γ enters into the matrix inequality bilinearly, it leads to a quasi-convex optimization problem. It is therefore possible to solved the analysis step more efficiently using existing method than the current bisection search.

Bibliography

[1] "World Wind Energy Report 2012," *Proceedings of the 12 th World Energy Conference*, World Wind Energy Association, Havana, Cuba, 2013.

[2] "Eolos wind energy research consortium," http://www.eolos.umn.edu/.

[3] "Wind turbine components," http://www.alternative-energy-news.info/images/technical/wind-turbine.jpg.

[4] Aho, J., Buckspan, A., Laks, J., Fleming, P., Jeong, Y., Dunne, F., Churchfield, M., Pao, L., and Johnson, K., "A Tutorial of Wind Turbine Control for Supporting Grid Frequency through Active Power Control," *Proceedings of American Control Conference*, IEEE, 2012, pp. 3120–3131.

[5] Lindenberg, S., Smith, B., and O'Dell, K., "20% wind energy by 2030," *National Renewable Energy Laboratory (NREL), US Department of Energy, Renewable Energy consulting Services, Energetics Incorporated*, 2008.

[6] Crabtree, G., Misewich, J., Ambrosio, R., Clay, K., DeMartini, P., James, R., Lauby, M., Mohta, V., Moura, J., Sauer, P., et al., "Integrating Renewable Electricity on the Grid," *AIP Conference Proceedings-American Institute of Physics*, Vol. 1401, 2011, p. 387.

[7] Mai, T., Wiser, R., Sandor, D., Brinkman, G., Heath, G., Denholm, P., Hostick, D., Darghouth, N., Schlosser, A., and Strzepek, K., "Exploration of High-Penetration Renewable Electricity Futures. Vol. 1 of Renewable Electricity Futures Study," *National Renewable Energy Laboratory, Golden, CO, Tech. Rep. NREL/TP-6A20-52409-1*, 2012.

[8] Burton, T., Sharpe, D., Jenkins, N., and Bossanyi, E., *Wind Energy Handbook*, John Wiley & Sons, 1st ed., 2001.

[9] Bossanyi, E. A., "Individual Blade Pitch Control for Load Reduction," *Wind Energy*, Vol. 6, No. 2, 2003, pp. 119–128.

[10] Bossanyi, E., "Wind turbine control for load reduction," *Wind energy*, Vol. 6, No. 3, 2003, pp. 229–244.

[11] Darrow, J., Johnson, K., and Wright, A., "Design of a tower and drive train damping controller for the three-bladed controls advanced research turbine operating in design-driving load cases," *Wind Energy*, Vol. 14, No. 4, 2011, pp. 571–601.

[12] Díaz de Corcuera, A., Pujana-Arrese, A., Ezquerra, J., Milo, A., and Landaluze, J., "Linear models-based LPV modelling and control for wind turbines," *Wind Energy*, Vol. 18, No. 7, 2015, pp. 1151–1168.

[13] Licari, J., Ugalde-Loo, C. E., Ekanayake, J. B., and Jenkins, N., "Comparison of the performance and stability of two torsional vibration dampers for variable-speed wind turbines," *Wind Energy*, Vol. 18, No. 9, 2015, pp. 1545–1559.

[14] Boyd, S., El Ghaoui, L., Feron, E., and Balakrishnan, V., *Linear Matrix Inequalities in System and Control Theory*, Siam, 1994.

[15] Gahinet, P. and Apkarian, P., "A linear matrix inequality approach to H_∞ control," *International journal of robust and nonlinear control*, Vol. 4, No. 4, 1994, pp. 421–448.

[16] Ossmann, D., Theis, J., and Seiler, P., "Robust Control Design For Load Reduction on a Liberty Wind Turbine," *submitted to the ASME Dynamic Systems and Control Conference*, 2016.

[17] Jonkman, J. and Buhl, M., *FAST User's Guide*, National Renewable Energy Laboratory, Golden, Colorado, 2005.

[18] Hjartarson, A., Seiler, P., and Packard, A., "LPVTools: A toolbox for modeling, analysis, and synthesis of parameter varying control systems," *IFAC-PapersOnLine*, Vol. 48, No. 26, 2015, pp. 139–145.

[19] Megretski, A. and Rantzer, A., "System Analysis via Integral Quadratic Constraints," *IEEE Trans. on Automatic Control*, Vol. 42, 1997, pp. 819–830.

[20] Zhou, K., Doyle, J., and Glover, K., *Robust and Optimal Control*, Prentice-Hall, 1996.

[21] Hardy, C., "Renewable energy and role of Marykirks James Blyth," *The courier. DC Thomson & Co. Ltd, Dundee (06/07/2013)*, 2010.

[22] "Mr. Brush's Windmill Dynamo," *Scientific American*, Dec. 1890.

[23] Østergaard, K. Z., *Robust, Gain-Scheduled Control of Wind Turbines*, Ph.D. thesis, Automation and Control, Department of Electronic Systems, Aalborg University, 2008.

[24] Ozdemir, A. A., *Preview Control for Wind Turbines*, Ph.D. thesis, University of Minnesota, 2013.

[25] "The world's 10 biggest wind turbines," Jan. 2014, http://www.power-technology.com/features/featurethe-worlds-biggest-wind-turbines-4154395/.

[26] Manwell, J., McGowan, J., and Rogers, A., *Wind Energy Explained: Theory, Design, and Application*, Wiley, 2010.

[27] Bossanyi, E., "The design of closed loop controllers for wind turbines," *Wind Energy*, Vol. 3, 2000, pp. 149–163.

[28] Laks, J., Pao, L., and Wright, A., "Control of Wind Turbines: Past, Present, and Future," *Proceedings of American Control Conference*, 2009, pp. 2096–2103.

[29] Johnson, K., Pao, L., Balas, M., and Fingersh, L., "Control of variable-speed wind turbines: standard and adaptive techniques for maximizing energy capture," *IEEE Control System Magazine*, Vol. 26, No. 3, 2006, pp. 70–81.

[30] Buhl Jr, M. L. and Manjock, A., "A comparison of wind turbine aeroelastic codes used for certification," *Conference Paper National Renewable Energy Laboratory, Golden, CO, Report No. NREL/CP-500-39113*, 2006.

[31] Clough, R. W. and Penzien, J., "Dynamics of structures," Tech. rep., 1975.

[32] Meirovitch, L., *Dynamics and control of structures*, John Wiley & Sons, 1990.

[33] Jonkman, B. J., "TurbSim user's guide: Version 1.50," 2009.

[34] Commission, I. E. et al., "IEC 61400-1: Wind turbines part 1: Design requirements," *International Electrotechnical Commission*, 2005.

[35] Moriarty, P. J. and Hansen, A. C., *AeroDyn theory manual*, Citeseer, 2005.

[36] Sørensen, J. N., "Blade-Element/Momentum Theory," *General Momentum Theory for Horizontal Axis Wind Turbines*, Springer, 2016, pp. 99–121.

[37] Johnson, W., "Helicopter TheoryPrinceton University Press," *Princeton, New Jersey, USA*, 1980.

[38] Stol, K., Balas, M., and Bir, G., "Floquet modal analysis of a teetered-rotor wind turbine," *Journal of solar energy engineering*, Vol. 124, No. 4, 2002, pp. 364–371.

[39] Bir, G., "Multiblade coordinate transformation and its application to wind turbine analysis," *ASME Wind Energy Symposium*, 2008.

[40] Stol, K. A., Moll, H.-G., Bir, G., and Namik, H., "A comparison of multi-blade coordinate transformation and direct periodic techniques for wind turbine control design," *Proceedings of the 47th AIAA/ASME*, 2009.

[41] Bir, G. S., "User's guide to MBC3: Multi-blade coordinate transformation code for 3-bladed wind turbine," 2010.

[42] Selvam, K., Kanev, S., van Wingerden, J., van Engelen, T., and Verhaegen, M., "Feedback–feedforward individual pitch control for wind turbine load reduction," *International Journal of Robust and Nonlinear Control*, Vol. 19, No. 1, 2009, pp. 72–91.

[43] Pao, L. Y. and Johnson, K. E., "A tutorial on the dynamics and control of wind turbines and wind farms," *American Control Conference*, IEEE, 2009, pp. 2076–2089.

[44] Harris, M., Hand, M., and Wright, A., "Lidar for turbine control," *National Renewable Energy Laboratory, Golden, CO, Report No. NREL/TP-500-39154*, 2006.

[45] Simley, E., Pao, L. Y., Frehlich, R., Jonkman, B., and Kelley, N., "Analysis of wind speed measurements using continuous wave LIDAR for wind turbine control," *Proceedings of the 49th AIAA Aerospace Sciences Meeting*, 2011.

[46] Wang, N., Johnson, K. E., and Wright, A. D., "FX-RLS-based feedforward control for LIDAR-enabled wind turbine load mitigation," *Control Systems Technology, IEEE Transactions on*, Vol. 20, No. 5, 2012, pp. 1212–1222.

[47] Ozdemir, A. A., Seiler, P., and Balas, G. J., "Design tradeoffs of wind turbine preview control," *Control Systems Technology, IEEE Transactions on*, Vol. 21, No. 4, 2013, pp. 1143–1154.

[48] Schlipf, D., Schlipf, D. J., and Kühn, M., "Nonlinear model predictive control of wind turbines using LIDAR," *Wind Energy*, Vol. 16, No. 7, 2013, pp. 1107–1129.

[49] Rezaei, V., "LIDAR-based robust wind-scheduled control of wind turbines," *American Control Conference*, IEEE, 2014, pp. 4416–4421.

[50] Bossanyi, E., Kumar, A., and Hugues-Salas, O., "Wind turbine control applications of turbine-mounted LIDAR," *Journal of Physics: Conference Series*, Vol. 555, IOP Publishing, 2014, p. 012011.

[51] Lim, D.-W., Mantell, S. C., and Seiler, P. J., "Wireless structural health monitoring of wind turbine blades using an energy harvester as a sensor," *Proceedings of the 32nd ASME wind energy symposium. MD: National Harbor*, 2014.

[52] Ozdemir, A. A., Seiler, P. J., and Balas, G. J., "Performance of Disturbance Augmented Control Design in Turbulent Wind Conditions," *Mechatronics*, Vol. 21, No. 4, 2011, pp. 634 – 644.

[53] Fleming, P. A., Van Wingerden, J.-W., and Wright, A. D., *Comparing state-space multivariable controls to multi-siso controls for load reduction of drivetrain-coupled modes on wind turbines through field-testing*, American Institute of Aeronautics and Astronautics (AIAA), 2012.

[54] Bianchi, F. D., De Battista, H., and Mantz, R. J., "Robust Multivariable Gain-Scheduled Control of Wind Turbines for Variable Power Production." *International Journal of Systems Control*, Vol. 1, No. 3, 2010.

[55] Inthamoussou, F. A., Bianchi, F. D., De Battista, H., and Mantz, R. J., "LPV wind turbine control with anti-windup features covering the complete wind speed range," *Energy Conversion, IEEE Transactions on*, Vol. 29, No. 1, 2014, pp. 259–266.

[56] Apkarian, P. and Gahinet, P., "A Convex Characterization of Gain-Scheduled H_∞ Controllers," *IEEE Trans. on Automatic Control*, Vol. 40, 1995, pp. 853–864.

[57] Wu, F., *Control of Linear Parameter Varying Systems*, Ph.D. thesis, University of California, Berkeley, 1995.

[58] Wu, F., Yang, X. H., Packard, A., and Becker, G., "Induced $\mathcal{L}_2$ norm control for LPV systems with bounded parameter variation rates," *International Journal of Robust and Nonlinear Control*, Vol. 6, 1996, pp. 983–998.

[59] Hoffmann, C. and Werner, H., "A survey of linear parameter-varying control applications validated by experiments or high-fidelity simulations," *Control Systems Technology, IEEE Transactions on*, Vol. 23, No. 2, 2015, pp. 416–433.

[60] Bianchi, F., Mantz, R., and Christiansen, C., "Control of variable-speed wind turbines by LPV gain scheduling," *Wind Energy*, Vol. 7, No. 1, 2004, pp. 1–8.

[61] Østergaard, K. Z., Stoustrup, J., and Brath, P., "Linear parameter varying control of wind turbines covering both partial load and full load conditions," *International Journal of Robust and Nonlinear Control*, Vol. 19, No. 1, 2009, pp. 92–116.

[62] Bianchi, F. D., De Battista, H., and Mantz, R. J., "Robust Multivariable Gain-Scheduled Control of Wind Turbines for Variable Power Production." *International Journal of Systems Control*, Vol. 1, No. 3, 2010.

[63] Bobanac, V., Jelavić, M., and Perić, N., "Linear Parameter Varying Approach to Wind Turbine Control," *14th International Power Electronics and Motion Control Conference*, 2010, pp. T12–60–T12–67.

[64] Bakka, T., Karimi, H.-R., and Christiansen, S., "Linear parameter-varying modelling and control of an offshore wind turbine with constrained information," *Control Theory & Applications, IET*, Vol. 8, No. 1, 2014, pp. 22–29.

[65] Sloth, C., Esbensen, T., and Stoustrup, J., "Robust and fault-tolerant linear parameter-varying control of wind turbines," *Mechatronics*, Vol. 21, No. 4, 2011, pp. 645–659.

[66] Adegas, F. D., Sonderby, I. B., Hansen, M. H., and Stoustrup, J., "Reduced-order LPV model of flexible wind turbines from high fidelity aeroelastic codes," *Control Applications (CCA), 2013 IEEE International Conference on*, IEEE, 2013, pp. 424–429.

[67] Shirazi, F. A., Grigoriadis, K. M., and Viassolo, D., "An integrated approach towards structural and LPV controller design in wind turbines," *American Control Conference (ACC), 2012*, IEEE, 2012, pp. 5789–5794.

[68] Apkarian, P. and Adams, R., "Advanced gain-scheduling techniques for uncertain systems," *IEEE Trans. on Control Systems Technology*, Vol. 6, No. 1, 1998, pp. 21–32.

[69] Packard, A., "Gain scheduling via linear fractional transformations," *Systems & Control Letters*, Vol. 22, No. 2, 1994, pp. 79–92.

[70] Safonov, M. G., Limebeer, D. J. N., and Chiang, R. Y., "Simplifying the H_∞ theory via loop-shifting, matrix-pencil and descriptor concepts," *International Journal of Control*, Vol. 50-6, 1989, pp. 2467–2488.

[71] Mikkelsen, T., Hansen, K., Angelou, N., Sjöholm, M., Harris, M., Hadley, P., Scullion, R., Ellis, G., and Vives, G., "Lidar wind speed measurements from a rotating spinner," *Proc. European Wind Energy Conference, Warsaw, Poland*, 2010.

[72] Knudsen, T., Bak, T., and Soltani, M., "Prediction models for wind speed at turbine locations in a wind farm," *Wind Energy*, Vol. 14, No. 7, 2011, pp. 877–894.

[73] M. Buhl, J., "MCrunch User's Guide for Version 1.00," May 2008, http://www.nrel.gov/docs/fy08osti/43139.pdf.

[74] Moreno, C., Seiler, P., and Balas, G., "Linear Parameter Varying Model Reduction for Aeroservoelastic Systems," *AIAA Atmospheric Flight Mechanics Conference*, 2012, pp. Paper No. AIAA–2012–4859.

[75] Wang, S. and Seiler, P., "Gain Scheduled Active Power Control for Wind Turbines," *AIAA Atmospheric Flight Mechanics Conference*, 2014, pp. Paper No. AIAA–2014–1220.

[76] Balas, G., Chiang, R., Packard, A., and Safonov, M., "Robust Control Toolbox 3 User's Guide," Tech. rep., The Math Works, Inc., 2007.

[77] Pfifer, H. and Seiler, P., "Robustness Analysis of Linear Parameter Varying Systems Using Integral Quadratic Constraints," *accepted to the International Journal of Robust and Nonlinear Control*, 2014.

[78] Pfifer, H. and Seiler, P., "Robustness Analysis of Linear Parameter Varying Systems Using Integral Quadratic Constraints," *American Control Conference*, 2014, pp. 4476–4481.

[79] Cantoni, M., Jönsson, U., and Khong, S., "Robust stability analysis for feedback interconnections of time-varying linear systems," *SIAM J. of Control Optim.*, Vol. 51, No. 1, 2013, pp. 353–379.

[80] Veenman, J. and Scherer, C., "On robust LPV controller synthesis: A dynamic integral quadratic constraint based approach," *IEEE Conference on Decision and Control*, 2010, pp. 591–596.

[81] Veenman, J. and Scherer, C., "IQC-synthesis with general dynamic multipliers," *International Journal of Robust and Nonlinear Control*, Vol. 24, 2012, pp. 3027–3056.

[82] Scherer, C. and Kose, I., "Gain-scheduled control synthesis using dynamic D-scales," *IEEE Trans. on Automatic Control*, Vol. 57, 2012, pp. 2219–2234.

[83] Scherer, C., "Gain-scheduled synthesis with dynamic positive real multipliers," *IEEE Conference on Decision and Control*, 2012, pp. 6641–6646.

[84] Wang, S., Pfifer, H., and Seiler, P., "Robust Synthesis for Linear Parameter Varying Systems Using Integral Quadratic Constraints," *IEEE Conference on Decision and Control*, 2014, pp. 4789–4794.

[85] Yakubovich, V., "S-procedure in nonlinear control theory," *Vestnik Leningrad Univ.*, 1971, pp. 62–77.

[86] Scherer, C. and Wieland, S., "Linear Matrix Inequalities in Control," Lecture notes for a course of the dutch institute of systems and control, Delft University of Technology, 2004.

[87] Fu, M., Dasgupta, S., and Soh, Y., "Integral quadratic constraint approach vs. multiplier approach," *Automatica*, Vol. 41, 2005, pp. 281–287.

[88] Carrasco, J., Heath, W., and Lanzon, A., "Factorization of multipliers in passivity and IQC analysis," *Automatica*, Vol. 48, No. 5, 2012, pp. 909–916.

[89] Megretski, A., "KYP Lemma for Non-Strict Inequalities and the associated Minimax Theorem," Arxiv, 2010.

[90] Scherer, C. and Köse, I., "Robustness with dynamic IQCs: An exact state-space characterization of nominal stability with applications to robust estimation," *Automatica*, Vol. 44, 2008, pp. 1666–1675.

[91] Turner, M. and Kerr, M., "L_2 gain bounds for systems with sector bounded and slope-restricted nonlinearities," *International Journal of Robust and Nonlinear Control*, Vol. 22, No. 13, 2012, pp. 1505–1521.

[92] Young, P., *Robustness with Parametric and Dynamic Uncertainty*, Ph.D. thesis, California Institute of Technology, 1993.

[93] Young, P., "Controller Design with Mixed Uncertainties," *American Control Conference*, 1994, pp. 2333–2337.

[94] Schaft, A. and A.J.Schaft, *L_2-gain and passivity in nonlinear control*, Springer-Verlag New York, Inc., 1999.

[95] Jönsson, U., *Robustness analysis of uncertain and nonlinear systems*, Ph.D. thesis, Lund Institute of Technology, 1996.

[96] Skogestad, S. and Postlethwaite, I., *Multivariable Feedback Control*, John Wiley and Sons, Chichester, 2005.

[97] Rebours, Y., Kirschen, D., Trotignon, M., and Rossignol, S., "A Survey of Freqency and Voltage Control Ancillary Services-Part I: Technical Features," *IEEE Transactions on Power Systems*, Vol. 22, No. 1, 2007, pp. 350–357.

[98] Greedy, L., "Review of electrical drive-train topologies," *Project UpWind, Mekelweg, the Netherlands and Aalborg East, Denmark, Tech. Rep*, 2007.

[99] Nelson, R. J., "Frequency-Responsive Wind Turbine Output Control," 2011, U.S. Patent 0 001 318.

[100] Keung, P.-K., Li, P., Banakar, H., and Ooi, B. T., "Kinetic Energy of Wind Turbine Generators for System Frequency Support," *IEEE Transactions on Power Systems*, Vol. 24, No. 1, 2009, pp. 279–287.

[101] Juankorena, X., Esandi, I., López, J., and Marroyo, L., "Method to Enable Variable Speed Wind Turbine Primary Regulation," *International Conference on Power Engineering, Energy and Electrical Drives, 2009*, IEEE, 2009, pp. 495–500.

[102] Aho, J., Buckspan, A., Pao, L., and Fleming, P., "An Active Power Control System for Wind Turbines Capable of Primary and Secondary Frequency Control for Supporting Grid Reliability," *51st AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, 2013, pp. AIAA 2013–0456.

[103] Acedo Sanchez, J., Carcar, M., Lusarreta, M., Perez Barbachano, J., Simon Segura, S., Sole Lopez, D., Zabaleta Maeztu, M., Marroyo Palomo, L., Lopez Taberna, J., et al., "Method of Operation of A Wind Turbine to Guarantee Primary or Secondary Regulation in An Electric Grid," 2011, U.S. Patent 0 057 445.

[104] Jeong, Y., Johnson, K., and Fleming, P., "Comparison and testing of power reserve control strategies for grid-connected wind turbines," *Wind Energy*, 2013.

[105] Tarnowski, G., Kjær, P., Dalsgaard, S., and Nyborg, A., "Regulation and Frequency Response Service Capability of Modern Wind Power Plants," *Proceedings of IEEE Power and Energy Society General Meeting*, 2010, pp. 1–8.

[106] Wang, S. and Seiler, P., "LPV Active Power Control and Robust Analysis for Wind Turbines," *33rd Wind Energy Symposium*, 2015, p. 1210.

[107] Pfifer, H. and Seiler, P., "Robustness Analysis of Linear Parameter Varying Systems Using Integral Quadratic Constraints," *International Journal of Robust and Nonlinear Control*, Vol. 25, 2015, pp. 2843–2864.

[108] Balas, G., Chiang, R., Packard, A., and Safonov, M., "MATLAB robust control toolbox 3, users guide," 2010.

[109] Peni, T. and Seiler, P., "Computation of lower bounds for the induced L_2 norm of LPV systems," Vol. 26, No. 4, 2016, pp. 646–661.

[110] Francis, B., *A Course in H_∞ Control Theory*, Springer-Verlag, 1987.

[111] Seiler, P., "Stability Analysis with Dissipation Inequalities and Integral Quadratic Constraints," *accepted to the IEEE Trans. on Automatic Control*, 2013.

[112] Bart, H., Gohberg, I., and Kaashoek, M., *Minimal Factorization of Matrix and Operator Functions*, Birkhäuser, 1979.

[113] Meinsma, G., "J-spectral Factorization and Equalizing Vectors," *Systems and Control Letters*, Vol. 25, 1995, pp. 243–249.

[114] Engwerda, J., *LQ Dynamic Optimization and Differential Games*, Wiley, 1st ed., 2005.

[115] Engwerda, J., "Uniqueness conditions for the affine open-loop linear quadratic differential game," *Automatica*, Vol. 44, 2008, pp. 504–511.

[116] Willems, J., "Least squares stationary optimal control and the algebraic Riccati equation," *IEEE Trans. on Aut. Control*, Vol. 16, 1971, pp. 621–634.

Appendix A

IQC Factorizations

Two factorizations of the frequency domain IQC multiplier Π are provided in Lemma 4 and Lemma 5. The first factorization given by Lemma 4 only assumes Π to be rational and uniformly bounded. However, this factorization (Ψ, M) does not, in general, yield a valid time domain IQC. The second factorization given by Lemma 5 is called a J-spectral factorization and it requires additional assumptions on Π. Lemma 6 shows that this J-spectral factorization yields a valid time domain IQC.

Lemma 4. *If $\Pi = \Pi^{\sim} \in \mathbb{RL}_{\infty}^{m \times m}$ then there exists real matrices $\tilde{A}, \tilde{B}, \tilde{Q}, \tilde{S}, \tilde{R}$ of compatible dimensions with $\tilde{A}$ Hurwitz, $\tilde{Q} = \tilde{Q}^T$, and $\tilde{R} = \tilde{R}^T$ such that*

$$\Pi(s) = \left[\, \tilde{B}^T(-sI-\tilde{A}^T)^{-1} \ \ I \,\right] \begin{bmatrix} \tilde{Q} & \tilde{S} \\ \tilde{S}^T & \tilde{R} \end{bmatrix} \begin{bmatrix} (sI-\tilde{A})^{-1}\tilde{B} \\ I \end{bmatrix} \tag{A.1}$$

Proof. The proof follows from arguments in Section 7.3 of [110]. Let $(A_\pi, B_\pi, C_\pi, D_\pi)$ be a minimal state-space realization for Π. Separate Π into its stable and unstable parts $\Pi = G_S + G_U + D_\pi$. Let $(\tilde{A}, \tilde{B}, \tilde{C}, 0)$ denote a state-space realization for the stable part G_S so that $\tilde{A}$ is Hurwitz. The assumptions on Π imply that G_U has a state-space realization of the form $(-\tilde{A}^T, -\tilde{C}^T, \tilde{B}^T, 0)$ (Section 7.3 of [110]). Thus $\Pi = G_S + G_U + D_\pi$ can be written as in Equation A.1 with $\tilde{Q} = 0$, $\tilde{S} = \tilde{C}^T$ and $\tilde{R} = D_\pi$. $\qquad\square$

Lemma 4 provides a factorization of Π in the form $\Psi^{\sim} M \Psi$ where $\Psi(s) := \begin{bmatrix} (sI-\tilde{A})^{-1}\tilde{B} \\ I \end{bmatrix}$ and $M := \begin{bmatrix} \tilde{Q} & \tilde{S} \\ \tilde{S}^T & \tilde{R} \end{bmatrix}$. The main construction in the proof is to separate the stable and unstable parts of Π. This can be easily implemented in Matlab using the `stabsep` command. This provides a factorization $\Pi = \Psi^{\sim} M \Psi$ where $\Psi \in \mathbb{RH}_{\infty}^{n_z \times m}$ is stable but non-square. Moreover, this factorization (Ψ, M) does not, in general, yield a valid time domain IQC as

described in Definition 2. Lemma 5 below states another special factorization with some additional assumptions on Π.

Lemma 5. *Let* $\Pi = \Pi^\sim \in \mathbb{RL}_\infty^{(n_v + n_w) \times (n_v + n_w)}$ *be partitioned as* $\begin{bmatrix} \Pi_{11} & \Pi_{12} \\ \Pi_{12}^\sim & \Pi_{22} \end{bmatrix}$ *where* $\Pi_{11} \in \mathbb{RL}_\infty^{n_v \times n_v}$ *and* $\Pi_{22} \in \mathbb{RL}_\infty^{n_w \times n_w}$. *If* $\Pi_{11}(j\omega) > 0$ *and* $\Pi_{22}(j\omega) < 0$ $\forall \omega \in \mathbb{R} \cup \{\infty\}$, *then*

(i) *There exists real matrices* $\tilde{A}, \tilde{B}, \tilde{Q}, \tilde{S}, \tilde{R}$ *of compatible dimensions with* $\tilde{A}$ *Hurwitz,* $\tilde{Q} = \tilde{Q}^T$, *and* $\tilde{R} = \tilde{R}^T$ *such that* Π *can be expressed as in Equation A.1.*

(ii) $\tilde{R}$ *is nonsingular and there exists a unique real solution* $X = X^T$ *to the the following ARE*

$$\tilde{A}^T X + X\tilde{A} - (X\tilde{B} + \tilde{S})\tilde{R}^{-1}(X\tilde{B} + \tilde{S})^T + \tilde{Q} = 0 \qquad (\text{A.2})$$

such that $\tilde{A} - \tilde{B}\tilde{R}^{-1}\left(X\tilde{B} + \tilde{S}\right)^T$ *is Hurwitz.*

(iii) Π *has a factorization* (Ψ, M) *with* $M := \begin{bmatrix} I_{n_v} & 0 \\ 0 & -I_{n_w} \end{bmatrix}$ *and* $\Psi, \Psi^{-1} \in \mathbb{RH}_\infty^{(n_v + n_w) \times (n_v + n_w)}$. *A state-space realization of* Ψ *is given by* $\left(\tilde{A}, \tilde{B}, \tilde{C}, \tilde{D}\right)$ *where* $\tilde{D}$ *is a solution of* $\tilde{R} = \tilde{D}^T M \tilde{D}$ *and* $\tilde{C} := M\tilde{D}^{-T}\left(\tilde{B}^T X + \tilde{S}^T\right)$.

Proof. Conclusion (i) holds for any $\Pi = \Pi^\sim$ and follows from Lemma 4. Conclusions (ii) and (iii) follow from Lemma 4 in [111]. □

The factorization in Conclusion (iii) is called a J-spectral factorization of Π. For this factorization, Ψ is square, stable, and stably invertible. Existence conditions for a J-spectral factor of Π are provided by the canonical factorization theorem in [112]. Chapter 7 of [110] summarizes these results. Existence conditions for a J-spectral factor can also be specified using the notion of an equalizing vector as defined in [113]. Lemma 5 above provides an alternative existence condition for a J-spectral factorization in terms of definiteness properties on Π. Lemma 6 below states that the J-spectral factorization is a hard factorization of Π. Thus a frequency domain IQC multiplier can, under some additional assumptions on Π, be factorized to yield a valid time domain IQC. Lemma 6 also provides an additional technical result that will be used in Chapter 4.

Lemma 6. *Let* $\Pi = \Pi^\sim \in \mathbb{RL}_\infty^{(n_v + n_w) \times (n_v + n_w)}$ *be partitioned as* $\begin{bmatrix} \Pi_{11} & \Pi_{12} \\ \Pi_{12}^\sim & \Pi_{22} \end{bmatrix}$ *where* $\Pi_{11} \in \mathbb{RL}_\infty^{n_v \times n_v}$ *and* $\Pi_{22} \in \mathbb{RL}_\infty^{n_w \times n_w}$. *Assume* $\Pi_{11}(j\omega) > 0$ *and* $\Pi_{22}(j\omega) < 0$ $\forall \omega \in \mathbb{R} \cup \{\infty\}$. *Let* (Ψ, M) *be the J-spectral factorization given in Conclusion (iii) of Lemma 5. Then,*

(i) (Ψ, M) *is a hard factorization of* Π*: If* Δ *is a bounded, causal operator that satisfies the frequency domain IQC specified by* Π *(Definition 1) then* Δ *satisfies the time domain IQC specified by* (Ψ, M) *(Definition 2).*

(ii) *The cost of the max/min game defined in Equation A.3 based on* (Ψ, M) *satisfies* $\underline{J}(x_{\psi 0}) = 0.$

$$\underline{J}(x_{\psi 0}) := \sup_{w \in L_2^{n_w}[0,\infty)} \inf_{v \in L_2^{n_v}[0,\infty)} \int_0^\infty z(t)^T M z(t) \, dt \qquad \text{(A.3)}$$

subject to:

$$\dot{x}_\psi = \tilde{A} x_\psi + \tilde{B} \begin{bmatrix} v \\ w \end{bmatrix}, \quad x_\psi(0) = x_{\psi 0}$$
$$z = \tilde{C} x_\psi + \tilde{D} \begin{bmatrix} v \\ w \end{bmatrix}$$

Proof. Conclusion (i) follows from Theorem 2.4 in [89] or Theorem 4 in [111]. Conclusion (ii) follows from Lemma 5 and the proof of Theorem 4 in [111]. $\underline{J}$ is the lower-value of a two-player differential game and the proof of Conclusion (ii) given in [111] essentially relies on results for LQ differential games [114, 115]. □

Appendix B

Extended System State Matrices

A state-space realization for the extended system of H_ρ and $\Psi_{1/\gamma}$ is given in Equation 4.9. The state matrices for the extended system can be expressed in terms of the state matrices for H_ρ (Equation 4.7) and $\Psi_{1/\gamma}$ (Equation 4.8) as:

$$\mathcal{A}(\rho) := \begin{bmatrix} A(\rho) & 0 \\ \gamma^{-1}\tilde{B}_v C_v(\rho) & \tilde{A} \end{bmatrix} \tag{B.1}$$

$$\mathcal{B}_w(\rho) := \begin{bmatrix} B_w(\rho) \\ \gamma^{-1}\tilde{B}_v D_{vw}(\rho) + \tilde{B}_w \end{bmatrix}, \qquad \mathcal{B}_d(\rho) := \begin{bmatrix} B_d(\rho) \\ \gamma^{-1}\tilde{B}_v D_{vd}(\rho) \end{bmatrix} \tag{B.2}$$

$$\mathcal{C}_{z_k}(\rho) := \begin{bmatrix} \gamma^{-1}\tilde{D}_{z_k v} C_v(\rho), & \tilde{C}_{z_k} \end{bmatrix}, \qquad \mathcal{C}_e(\rho) := \begin{bmatrix} C_e(\rho), & 0 \end{bmatrix} \tag{B.3}$$

$$\mathcal{D}_{z_k w}(\rho) := \gamma^{-1}\tilde{D}_{z_k v} D_{vw}(\rho) + \tilde{D}_{z_k w}, \qquad \mathcal{D}_{ew}(\rho) := D_{ew}(\rho) \tag{B.4}$$

$$\mathcal{D}_{z_k d}(\rho) := \gamma^{-1}\tilde{D}_{z_k v} D_{vd}(\rho), \qquad \mathcal{D}_{ed}(\rho) := D_{ed}(\rho) \tag{B.5}$$

Similarly, the state-space realization for the extended system of H_ρ and Ψ_λ is given in Equation 4.14. These state matrices can be expressed in terms of the state matrices for H_ρ (Equation 4.7) and Ψ_λ (Equation 4.13). Only the output and feedthrough matrices associated with the output z_λ are changed. These are given by:

$$\mathcal{C}_{z_\lambda}(\rho) := \begin{bmatrix} \gamma^{-1}\tilde{D}_{z_\lambda v} C_v(\rho), & \tilde{C}_{z_\lambda} \end{bmatrix} \tag{B.6}$$

$$\mathcal{D}_{z_\lambda w}(\rho) := \gamma^{-1}\tilde{D}_{z_\lambda v} D_{vw}(\rho) + \tilde{D}_{z_\lambda w} \tag{B.7}$$

$$\mathcal{D}_{z_\lambda d}(\rho) := \gamma^{-1}\tilde{D}_{z_\lambda v} D_{vd}(\rho) \tag{B.8}$$

Appendix C

Proof of Lemma 1

Proof. ($\Rightarrow$) Assume $P = P^T$ satisfies Equation 4.15. The output z_k from $\Psi_{1/\gamma}$ is a linear function of (x_ψ, v, w) as defined in Equation 4.8:

$$z_k = \begin{bmatrix} \tilde{C}_{z_k} & \gamma^{-1}\tilde{D}_{z_k v} & \tilde{D}_{z_k w} \end{bmatrix} \begin{bmatrix} x_\psi \\ v \\ w \end{bmatrix} \tag{C.1}$$

These variables (x_ψ, v, w) can, in turn, be expressed in terms of the extended system state and inputs (x, w, d) as:

$$\begin{bmatrix} x_\psi \\ v \\ w \end{bmatrix} = \begin{bmatrix} [0, \ I] & 0 & 0 \\ [C_v(\rho), \ 0] & D_{vw}(\rho) & D_{vd}(\rho) \\ [0, \ 0] & I & 0 \end{bmatrix} \begin{bmatrix} \begin{bmatrix} x_H \\ x_\psi \end{bmatrix} \\ w \\ d \end{bmatrix} := L(\rho) \begin{bmatrix} \begin{bmatrix} x_H \\ x_\psi \end{bmatrix} \\ w \\ d \end{bmatrix} \tag{C.2}$$

Thus, using the extended system state matrices defined in Appendix B, the second term of the matrix inequality in Equation 4.15 can be rewritten as:

$$\sum_{k=1}^{N} \lambda_k \begin{bmatrix} \mathcal{C}_{z_k}^T \\ \mathcal{D}_{z_k w}^T \\ \mathcal{D}_{z_k d}^T \end{bmatrix} M_k \begin{bmatrix} \mathcal{C}_{z_k} & \mathcal{D}_{z_k w} & \mathcal{D}_{z_k d} \end{bmatrix} = L(\rho)^T \begin{bmatrix} \tilde{Q}_\lambda & \tilde{S}_\lambda \\ \tilde{S}_\lambda^T & \tilde{R}_\lambda \end{bmatrix} L(\rho) \tag{C.3}$$

$\tilde{Q}_\lambda$, $\tilde{S}_\lambda$, and $\tilde{R}_\lambda$ are defined in Equation 4.12. Substitute for $\tilde{Q}_\lambda$ using the ARE in Equation A.2. Rearrange terms in the matrix inequality to show that $\tilde{P} := P + \begin{bmatrix} 0 & 0 \\ 0 & X \end{bmatrix}$ satisfies Equation 4.16.

This direction of the proof is completed by showing that $\tilde{P} \geq 0$. Define the quadratic function $V(x_0) := x_0^T \tilde{P} x_0$. In addition, define the following quadratic cost functional $V^*(x_0)$

based on the extended system of H_ρ and the rescaled factorization $(\Psi_\lambda, M_\lambda)$:

$$V^*(x_0) := \sup_{w \in L_2^{n_w}[0,\infty)} \int_0^\infty z_\lambda(t)^T M_\lambda z_\lambda(t)\, dt \qquad \text{(C.4)}$$

subject to:

$$\dot{x} = \mathcal{A}(\rho)x + \mathcal{B}_w(\rho)w, \qquad x(0) = x_0$$

$$z_\lambda = \mathcal{C}_{z_\lambda}(\rho)x + \mathcal{D}_{z_\lambda w}(\rho)w$$

The disturbance input of the extended system is neglected ($d = 0$) in this linear quadratic optimization. Note that the extended system is stable since H_ρ is stable (by assumption), Ψ_λ is stable (by construction), and Ψ_λ is connected in an open loop fashion to H_ρ. First we show that $V(x_0) \geq V^*(x_0)$ for all $x_0 \in \mathbb{R}^{n_H + n_\psi}$. This follows along the lines of Theorems 2 and 3 in [116] and hence the proof is only sketched. Let $x(t)$, $z_\lambda(t)$ be the resulting solutions of the extended system of H_ρ and Ψ_λ for a given input $w \in L_2^{n_w}[0,\infty)$, admissible trajectory $\rho \in \mathcal{T}$, and initial condition $x_0 \in \mathbb{R}^{n_H + n_\psi}$ assuming $d = 0$. Multiply the matrix inequality in Equation 4.16 on the left/right by $\begin{bmatrix} x(t) \\ w(t) \\ 0 \end{bmatrix}^T$ and $\begin{bmatrix} x(t) \\ w(t) \\ 0 \end{bmatrix}$ to show $\dot{V}(x(t)) + z_\lambda(t)^T M_\lambda z_\lambda(t) \leq 0$. Integrate this inequality from $t = 0$ to $t = T$ to obtain

$$V(x(T)) + \int_0^T z_\lambda(t)^T M_\lambda z_\lambda(t)\, dt \leq V(x_0) \qquad \text{(C.5)}$$

$\lim_{T \to \infty} x(T) = 0$ for any $w \in L_2^{n_w}[0,\infty)$ because the extended system is stable. Maximizing the left side of Equation C.5 over $w \in L_2^{n_w}[0,\infty)$ for $T = \infty$ thus yields $V(x_0) \geq V^*(x_0)$.

Next, consider the max/min game defined for the rescaled J-spectral factorization $(\Psi_\lambda, M_\lambda)$:

$$\underline{J}(x_{\psi 0}) := \sup_{w \in L_2^{n_w}[0,\infty)} \inf_{v \in L_2^{n_v}[0,\infty)} \int_0^\infty z_\lambda(t)^T M_\lambda z_\lambda(t)\, dt \qquad \text{(C.6)}$$

subject to:

$$\dot{x}_\psi = \tilde{A}x_\psi + \tilde{B}\begin{bmatrix} v \\ w \end{bmatrix}, \quad x_\psi(0) = x_{\psi 0}$$

$$z = \tilde{C}_{z_\lambda}x_\psi + \tilde{D}_{z_\lambda}\begin{bmatrix} v \\ w \end{bmatrix}$$

where $\tilde{D}_{z_\lambda} := [\tilde{D}_{z_\lambda v}, \tilde{D}_{z_\lambda w}]$. This max/min game is connected to the quadratic optimization defined in Equation C.4. Specifically, restricting v in the max/min game to be the output of H_ρ generated by $w \in L_2$, $d = 0$, and $x_H(0) = x_{H0}$ yields the quadratic optimization in Equation C.4. This specific choice of v yields a value that is no lower than the infimum over all possible $v \in L_2$. Hence the max/min game yields the bound $\underline{J}(x_{\psi 0}) \leq V^*(x_0)$. By Lemma 6, the cost of this max/min game is $\underline{J}(x_{\psi 0}) = 0$. Putting these results together

114

yields the following inequality

$$0 = \underline{J}(x_{\psi 0}) \leq V^*(x_0) \leq V(x_0) := x_0^T \tilde{P} x_0 \qquad (\text{C.7})$$

This holds for any x_0 and thus $\tilde{P} \geq 0$.

($\Longleftarrow$) This direction of the proof essentially involves reversing the algebraic rearrangement to go from the matrix inequality in Equation 4.16 to the form in Equation 4.15. $\qquad \square$